A Multi-Objective Approach for VNE Problems using multiple ILP formulations

Enrique Dávalos[1], Cristian Aceval[2], Víctor Franco[3], Benjamín Barán[4]
Universidad Nacional de Asunción, [1,3,4]Facultad Politécnica, [2]Centro Nacional de Computación
San Lorenzo, Paraguay, 2111
{edavalos, bbaran}@pol.una.py, {cristian.aceval, victorfranco90}@gmail.com

Abstract

Network Virtualization is a key technology for the Future Internet, allowing the deployment of multiple independent virtual networks that use resources of the same basic infrastructure. An important challenge in the dynamic provision of virtual networks resides in the optimal allocation of physical resources (nodes and links) to requirements of virtual networks. This problem is known as Virtual Network Embedding (VNE). For the resolution of this problem, previous research has focused on designing algorithms based on the optimization of a single objective. On the contrary, in this work we present a multi-objective algorithm, called VNE-MO-ILP, for solving dynamic VNE problem, which calculates an approximation of the Pareto Front considering simultaneously resource utilization and load balancing. Experimental results show evidences that the proposed algorithm is better or at least comparable to a state-of-the-art algorithm. Two performance metrics were simultaneously evaluated: (i) Virtual Network Request Acceptance Ratio and (ii) Revenue/Cost Relation. The size of test networks used in the experiments shows that the proposed algorithm scales well in execution times, for networks of 84 nodes.

Keywords: ILP, VNE, Virtual network embedding, Network virtualization, Multi-objective optimization.

1 Introduction

Network Virtualization is an important component in the evolution of Internet since it enables functions that were not available in the original design, and it breaks with the process of Internet ossification [1, 2]. Besides, many testbeds of academic and industry organizations are based on this paradigm with the goal of providing an independent research environment to different working groups in the area [3-5].

A virtual network is formed by virtual nodes and virtual links, and shares resources from nodes and links of the physical or substrate network. Both virtual nodes and virtual links form a Virtual Topology [6]. Multiple virtual networks can coexist, isolated and independent from each other, mapped on the same physical hardware.

The problem of assigning physical network resources to Virtual Networks Requirements (VNR) is known as VNE (Virtual Network Embedding) and it is critical for the deployment of Network Virtualization [7]. As in most engineering optimization problems, the goal is to maximize benefits with an efficient allocation. These benefits can refer to maximizing profits, optimal utilization of physical resources or achieving a desired QoS (Quality of Service). VNE problem is intrinsically complex, not only in its mathematical formulation but also at the computational level [8], and currently it is considered as a NP-hard problem [6].

Due to its complexity, VNE is usually divided in 2 sub-problems: (1) Node Mapping, that assigns resources (as processing capacity) of physical nodes to virtual nodes, and (2) Virtual Link Mapping that assigns links or path formed by consecutive physical links with their corresponding bandwidth, to virtual links. Virtual nodes only can be assigned to physical nodes if there are sufficient resources for hosting this virtual node in the physical node. Analogously, virtual links have to be assigned to physical links with available resources (as bandwidth) to host them.

We can consider two versions of the VNE problem: the static or offline version, in which VNRs are known in advance, and the dynamic or online version, where the VNR are treated individually, one at a time, as they appear in a time basis. This work deals with the online version.

Recently, VNE problem has attracted a lot of attention from the scientific community, as we can see in published surveys [6, 9]. Proposed methods for the VNE resolution are mainly based on heuristics and ILP (Integer Lineal Programming) formulations.

Early works dealing with VNE have proposed algorithms in which the allocation of virtual nodes and links are made in an independent way, with no coordination between these two sub-problems, as in [7, 10]. On the other hand, Chowdhury et al. proposed coordinated allocation of virtual nodes and virtual links, but in two separated phases [11].

In [12], Cheng et al. performed both allocations in one coordinated phase, using topological attributes of the VNR and the Substrate Network, elaborating a ranking of nodes. This work was inspired in the PageRank algorithm used by Google for web page ranking [13]. In this way, they tried to take into account the incidence of the network topology. In the same line, the authors of [14] present a heuristic that maps virtual nodes and virtual links in a coordinated way. A new metric is proposed, named Global Resource Capacity (GRC), which takes into consideration the network topology. Based on GRC, the heuristic first maps all virtual nodes, and then performs virtual link mapping based on shortest paths. The objective is to minimize the revenue-cost relation.

The work of Lischka and Karl in [15] also makes the allocations of nodes and links in a coordinated way, in this case using Subgraph Isomorphism Detection (SID) algorithms. With this proposal, an isomorph subgraph that represents a VNR is searched in the physical topology, applying restrictions that limit the quantity of hops to map the virtual links.

In [16] the authors make the observation that in many real scenarios, resources requirement of most applications vary in time. They present a model that exploits this situation and try to utilize unused resources of virtual networks to share in a more efficient way physical network resources. However, this strategy breaks with the isolation that must exist among virtual entities sharing the same substrate. As a final example of heuristics applied to VNE, [17] analyzes physical topology in order to plan physical network expansion, identifying partitions and cut links, looking for a better interconnection of the network while avoiding blocking of requirements.

All these cited works solved the VNE problem with heuristics that optimize a single specific objective. In [8], Melo et al. used an ILP formulation getting good results considering performance metrics as well as running time. They have evaluated three different objective-functions for the “node-link formulation”, concluding that the best option was the Weighted Shorted Distance Path (WSPD), which combines weighted objectives of load balance and shortest paths for the virtual links. A later work using exact methods is [18] which uses column generation strategies in order to minimize computational load and facilitate the scalability of the solution to a dynamic VNE problem.

While all these approaches are applied to generic packet-switched physical networks, some other works focus on a single technology in the substrate network for the purpose of establishing specific restrictions and proposing a solution over a real scenario. For optical networks, late works are centered on Elastic Optical Networks (EON), in which the optical fiber spectrum is divided in Frequency Slots (FS). One or more contiguous FS can be used by a transducer, depending on its data bit rate and modulation technique. The work presented in [19] presents an exact solution for Virtual Optical Network Embedding (VONE) problem, based on the concept called Maximum contiguous slot-block (MCSB). Besides, they proposed two heuristics for both transparent (with no electrical-optical-electrical conversion in physical nodes) networks and opaque networks. Other work in the same area is [20] which treats multicast traffic in virtual networks, over optical EON physical substrates. For wireless networks, an example is [21] where the authors propose the use of partially available spectrum bands as aggregated capacities in 3G or LTE technologies.

The present paper is an extension of our previous work [22] in which we presented a multi-objective approach based on the idea that a trade-off exists between the cost of the allocation in terms of the utilization of physical resources, and the balance in a uniform distribution of resources in the substrate network, two of the most studied objective functions and therefore, both objectives should be optimized simultaneously. To the best of our knowledge, this work was the first in proposing a multi-objective approach that allows the calculation of a Pareto Front approximation [23] for the VNE, considering simultaneously the two above cited objective functions. In our previous work, we presented limited experiments using two different medium size network topologies. In what follows, the experiments were extended here to network topologies up to 84 nodes and 102 links, which can be already considered as a large network. Experimental results will be presented to demonstrate that the proposed algorithm scales well enough even to these large networks, confirming previous results published in [22].

The proposed algorithm, named in this work VNE-MO-ILP (Virtual Network Embedding - Multi-objective - Integer Lineal Programming) performs multiple executions of an ILP formulation, each one with variations on a given restriction. In this way, the Pareto Front is generated one point at a time, and the network operator has multiple trade-off options for allocating resources when allocating resources to a VNR.

Since the network operator has multiple options for the allocation of a single VNR, it is also interesting to investigate the effect of the criterion to be followed in the selection of a specific Pareto solution at each time instant, over the global performance metrics usually used for evaluating the efficiency of the algorithm, as: (1) the VNR Acceptance Ratio, and (2) the Revenue/Cost Relation [12].

The remainder of this work is organized as follows. In Section II, we formulate a formal description of the VNE problem. In Section III, the proposed algorithm VNE-MO-ILP is described. The simulation results, using a network simulator, is presented in Section IV. Finally, Sections V and VI conclude this paper.

2 Formal Description of the VNE Problem

This section describes the modeling of virtual and physical networks, defining also main restrictions in VNE problem formulation. This formulation is based on the proposal of Melo *et al.* [8]. We also present the performance metrics that will be used to evaluate the efficiency of the proposed algorithm, when compared to a state of the art alternative [8].

2.1 Modeling of the physical network and the VNR

The physical (or substrate) network is modeled as a weighted undirected graph $G^p = \{ N^p, L^p, C^p, B^p, D^p \}$ composed by a set of physical nodes N^p and a set of physical links L^p. Each physical node $i \in N^p$ is characterized by a processing capacity C_i^p (for instance, CPU units). We consider that each physical link $l_{ij} \in L^p$ has a bandwidth B_{ij}^p and a specific propagation delay D_{ij}^p.

In turn, a single VNR can be described as a weighted undirected graph $G^v = \{ N^v, L^v, C^v, B^v, D^v \}$ composed by a set of virtual nodes N^v and a set of virtual links L^v. Each virtual node $m \in N^v$ is characterized by the requirement of some processing capacity C_m^v. With respect to virtual links, each $l_{mn} \in L^v$, $m < n$, requires a bandwidth B_{mn}^v with a maximum propagation delay of D_{mn}^v. This work deals with the dynamic problem, so each VNR is known in a specific time t^I and has a life time t^v after which, it must be retired from the physical network and the used physical resources can be released for latter reuse by another VNR.

Besides, L_i^p represents the sub-set of physical links l_{ij} that are directly connected to physical node *i* in the physical network. Similarly, L_m^v represents the subset of virtual links l_{mn} that are directly connected to virtual node m in a given VNR.

2.2 Mapping Variables

The binary variable x_i^m represents the mapping of virtual nodes to physical nodes and it is defined by the expression:

$$x_i^m = \begin{cases} 1, & if\ virtual\ node\ m\ is\ mapped\ to\ physical\ node\ i \\ 0, & otherwise \end{cases} \tag{1}$$

The binary variable y_{ij}^{mn} represents the allocation of a virtual link to a path formed by consecutive physical links:

$$y_{ij}^{mn} = \begin{cases} 1, if\ virtual\ link\ l_{mn}\ uses\ physical\ link\ l_{ij} \\ 0, \quad otherwise \end{cases} \tag{2}$$

After a successful allocation and before a new incoming VNR is considered, the physical network must be updated, subtracting the resources in use by the new virtual network from the potentially available physical links and nodes.

2.3 VNE problem Restrictions

In order to ensure a correct mapping of virtual nodes and links, and obeying the conservation of physical network resources law, a set of restrictions are defined following the guideline proposed in [8] by Melo *et al*.

2.3.1 Allocation of virtual nodes to physical nodes

Relations (3) to (5) ensure that each virtual node is assigned to only one physical node, and that each physical node can be used by at most one virtual node per each virtual network (in the same VNR). Besides, the capacity of each physical node cannot be exceeded:

$$\forall m: \quad \sum_i x_i^m = 1 \tag{3}$$

$$\forall i: \quad \sum_m x_i^m \leq 1 \tag{4}$$

$$\forall i: \quad \sum_m x_m^i \,.\, C_m^v \leq C_i^p \tag{5}$$

2.3.2 Allocation of virtual links to physical links

In order to allow simultaneous optimization in the allocation of virtual links and virtual nodes, the *Multi-Commodity Flow* [24] restriction is applied as a whole, considering the *Node-Link Formulation* [8]. Besides, the notion of directional flows is used over the virtual links that represent expression (6), following the notation proposed in [8].

$$\forall\, l_{mn} \in L_m^v, \qquad m < n, \qquad \forall\, i:$$

$$\sum_{j \epsilon L_i^p} \left(y_{ij}^{mn} - y_{ji}^{mn} \right) = x_i^m - x_i^n \tag{6}$$

2.3.3 Bandwidth Limitation

Expression (7) assures that the available bandwidth capacity at each physical link will not be exceeded.

$$\forall\, l_{ij} \in L_i^p, \qquad i < j:$$

$$\sum_{m,n\, \epsilon\, L_m^v, m<n} B_{mn}^v \left(y_{ij}^{mn} + y_{ji}^{mn} \right) \leq B_{ij}^p \tag{7}$$

2.3.4 Limit in the Propagation Delay of virtual links

The maximum delay D^v_{mn} of a virtual link l_{mn} is a parameter of the problem that narrows the value of the additional physical link's delay that composes a virtual link. Melo *et al*. apply (8) to assure that this delay restriction is fulfilled:

$$\forall\, l_{mn} \in L^v_m\,, \qquad m < n, \qquad \forall\, i:$$
$$\sum_{j \in L^p_i, i<j} D^p_{ij}(\, y^{mn}_{ij} + y^{mn}_{ji}\,) \le D^v_{mn} \tag{8}$$

2.4 Performance metrics for VNE problem

It is necessary to define some metrics to evaluate the efficiency of the studied mapping algorithm. In the case of the dynamic version of the VNE problem, *i.e.* when VNRs arrive at different time slots without *a priori* knowledge, these metrics are calculated at the end of the simulation process, after making all allocations of virtual networks in a time basis.

The two performances metrics that we will use in this work are the most accepted in the specialized literature [10, 11] and are the following:

2.4.1 VNR Acceptance Ratio

This metric is denoted as A^{VNR}, and it is given by equation (9). It measures the global performance of the allocation method considering its capacity of mapping (and not blocking or rejecting) as many VNRs as possible.

$$A^{VNR} = \frac{k^{VNRa}}{k^{VNR}} \tag{9}$$

In this equation, k^{VNR} represents the total number of arriving VNRs, while k^{VNRa} is the number of VNRs successfully satisfied by the studied algorithm.

2.4.2 Revenue / Cost Relation

The cost of a VNR mapping is considered as the cost of the resources in the physical or substrate network that were used in the mapping process. The revenue can be considered as proportional to the costs of resources requested by a single VNR (CPU of virtual nodes and Bandwidth of virtual links). This indicator E^{VNR} gives an idea of the efficiency in the allocation process for the use of network resources and therefore, it represents more profit for the owner of the network infrastructure [7]. Coefficients α and β relate real costs of capacity units in the nodes to link bandwidth costs.

$$E^{VNR} = \frac{\alpha \sum_m C^v_m + \beta\, \sum_{m,n} B^v_{mn}}{\alpha \sum_i C^p_i + \beta \sum_{i,j} B^p_{ij}} \tag{10}$$

3 VNE-MO-ILP Algorithm

When an algorithm assigns resources to a given VNR, the objective of this assignation should point to get good results with the considered global performance metrics, in this case, Acceptance Ratio A^{VNR} and Revenue/Cost Relation E^{VNR}.

It is necessary to clearly understand the difference between the *Global performance metrics* and the *Objectives for a single VNR*. Global performance metrics evaluate the entire VNE process, *i.e.* at the end of all VNR assignment attempts, and measures the efficiency of the VNE algorithm as a whole. For an online process, they are calculated at the end of a large period of time, after many VNR allocations were made. For an offline process, the global performance metrics are calculated when all VNRs were attempted to allocate. These global metrics can be seen as *a posteriori* metrics. On the other hand, the objectives for a single VNR are optimization objectives applied to the less complex problem of the allocation of each VNR, *i.e.*, they are *a priori* metrics.

This work proposes for the first time the simultaneous optimization of both cited objectives at each mapping, instead of considering only one objective at a time, as already proposed in previous works, i.e., our proposal will simultaneously try to:

- *Minimize the utilization of physical resources:* the requirement of virtual nodes cannot be minimized, so it is intended to minimize the cost of used physical links, in order to satisfy the requirements with the least quantity of resources as possible, looking for efficiency. This objective has a direct relation with metric E^{VNR}; and

- *Maximize load balance:* in this case, the physical network resources will be assigned in such a way that the network remains with balanced resource availability in the nodes and links, in order to facilitate subsequent allocations and avoiding isolation of resources. In this way, we will try to improve the Acceptance Ratio A^{VNR} metric.

In the wake of the above descripted ideas, the algorithm proposed in this work, named here VNE-MO-ILP *(Virtual Network Embedding-Multi-objective-Integer Lineal Program-ming)* will:

- *Define a mapping cost SP*, given by equation (11), where *SP* is limited for a defined *a priori* minimum and maximum values. The algorithm takes (s+1) possible values of SP, equally spaced between maximum and minimum values, as will be shown in equation (15).

$$SP = \sum_{ij} \sum_{mn} \ y_{ij}^{mn} * B_{mn}^{v} \tag{11}$$

- *Optimize Load Balance:* the algorithm looks for resources in physical nodes with more available capacity, and in physical links with more bandwidth; therefore, the objective function of each ILP formulation will be (12), where γ defines a relation between the costs of physical nodes and links.

$$minimize\ f = \gamma.\left(\mathcal{L}_{max}^{C^p}\right) + \left(\mathcal{L}_{max}^{B^p}\right) \tag{12}$$

The values of maximum load in physical nodes and links, $\mathcal{L}_{max}^{C^p}$ and $\mathcal{L}_{max}^{B^p}$ respectively, are defined according to the following new restrictions which are added to the former restrictions already formulated in Section 2:

$$\forall i \in N^p:\ \ C_i^p - \sum_{m} x_i^m * C_m^v \le \mathcal{L}_{max}^{C^p} \tag{13}$$

$$\forall l_{ij} \in L^p:\ B_{ij}^p - \sum_{mn} y_{ij}^{mn} * C_{mn}^v \le \mathcal{L}_{max}^{B^p} \tag{14}$$

The proposed VNE-MO-ILP algorithm calculates an approximation of the Pareto Front, using the above summarized ideas. To the best of our knowledge, no work previous to [22] proposed a multi-objective approach to the VNE problem, even when more than a single performance metric to evaluate the efficiency of VNE algorithms is used (usually, as a weighted sum of different objectives).

In this way, the proposed algorithm offers to a network operator a whole Pareto Front approximation with different solutions (all optimal in a Pareto sense [23]) for a single mapping, in order to allow him to choose a Pareto optimal solution that best fits his requirements, in a specific situation, at a given moment in time.

The Algorithm VNE-MO-ILP presented in Table 1, initializes variables and parameters, then calculates the acceptable limits of parameter *SP* defined in (11), and the interval [SP_{min}, SP_{max}] is divided in s sub-intervals, to later assign consecutively those (s+1) values to the SP parameter:

$$SP = SP_{min} + q\ (SP_{max} - SP_{min})/s \quad \text{with } q = 0, 1, 2, \ldots, s \tag{15}$$

Then, an ILP formulation is built with the following elements:

- Parameters, assignation variables and restrictions presented in Section 2;
- Additional restrictions of expressions (11), (13) and (14);
- The objective-function defined in (12).

Table 1: VNE-MO-ILP Algorithm

Input: Updated physical network G, requirement VNR, ILP formulation, number of ILP executions s, Max_VNR_Cost.

Output: Pareto Front FP.

Initialize FP
SP_{min} = Sum of virtual links in VNR
SP_{max} = Max_VNR_Cost
for SP = SP_{min} to SP_{max} en hops of (SP_{min} – Sp_{max})/s **do**
    Assign value of SP to expression (15)
    Add (15) as a restriction of ILP formulation ILP
    Send ILP formulation to the ILP SOLVER
    if SOLVER () = feasible **then**
        Add Solution(SOLVER) to FP
    end-if
end-for
if FP= ϕ
    return (VNR-Blocked)
else
    eliminate dominated solutions in FP
    return (FP)
end-if

Subsequently, at each iteration, an *ILP Solver* tries to solve this ILP problem, returning a solution, if it exists. Each solution calculated by the ILP Solver is a point of the Pareto Front approximation, which will be the output of the VNE-MO-ILP algorithm.

This way, an *ILP solver*, which solves mono-objective problems, is used to find an approximation of the Pareto Front, being this Pareto Front a typical result of a pure multi-objective problem [23].

Although ILP method is not scalable to large instances of any problem, our proposed algorithm uses anyway this tool, taking into account the very good execution times reported in the reference work [8], and being aware that for more complex networks, efficient heuristics or meta-heuristics will be needed.

In this paper we take three different network topologies, which are representative of medium and large sizes ISP *(Internet Service Provider)* networks, with very good results and reasonable execution times, proving the viability of this proposal, leaving for future work the development of efficient metaheuristics for the VNE problem in a pure multi-objective context.

Table 2 highlights main differences between the algorithms proposed in this work VNE-MO-ILP with the VNE-NLF algorithm presented in [8], which is taken as a reference to evaluate the presented experimental results.

4 Experiments and Results

4.1 Discrete-Event Network Simulator

We have developed a discrete-event Network Simulator in *Java* language for the evaluation of the proposed algorithm, which is outlined in Figure 1. This simulation tool covers exactly all restrictions of the proposed VNE problem, as we presented in Section III, and permits the execution of a complete instance of the VNE problem (a set of VNR stored on a database, treated one by one in a time basis). This network simulator is available for the scientific community at: http://www.cc.pol.una.py/VNE-MO-ILP/.

After defining parameters of the physical network and a set of M virtual network requirements (VNR), the simulator randomly takes a VNR and it passes this requested VNR to the proposed VNE-MO-ILP algorithm, which interacts with the *ILP Solver* to build a Pareto Front approximation in a point-to-point basis with each calculated non-dominated solution (optimal in the Pareto sense). If the algorithm could not find any solution, the performance metrics are updated. On the contrary, if a Pareto Front approximation is found with more than a single solution, it is necessary to apply a (possibly subjective) criterion to choose a single solution, to update performance metrics and network parameters and to pass to the next time slot to continue with the simulation.

4.2 Test Networks and experimental environment

Three substrate networks were used in the following reported experiments: *IRIS*, *MARNET*, and *BESTEL*. These topologies were obtained from *The Internet Topology Zoo* [25] and are shown in Figures 2, 3 and 4 respectively. *IRIS* is a *mesh* topology with 51 nodes and 64 links; MARNET is a hybrid topology (*star/mesh*) with 20 nodes and 27 links. Both are representative of medium size networks. The BESTEL mesh topology has 84 nodes and 102 links and it is a typical large size network. The propagation delay at each physical link is proportional to its length. For the reported simulations, capacities at each node were generated randomly with values ranging from 200 to 300 capacity processing units. The transmission rate at each link is considered to be 512 Mbps.

The VNRs were generated using network topology generator *BRITE* [26]. The number of virtual nodes varies between 2 and 10 nodes. The virtual link bandwidth requirement takes an integer value uniformly distributed between 2 and 8 Mbps at each link, while the requirements in virtual nodes were uniformly distributed between 2 and 40 capacity processing units.

For the resolution of the ILP sub-problems, solver *IBM Ilog CPLEX* ® [27] version 12.6 was used. The execution times in the tests were registered for later comparisons.

Table 2: Comparison between VNE-MO-ILP and VNE-NLF Algorithms

Characteristic	VNE-NLF [8]	New proposal VNE-MO-ILP
Number of ILP Solver executions per each VNR allocation attempt	One	s + 1
Objective in allocating each single VNR	Optimize Resources Utilization and Load Balance, treated as a single objective in a fixed-weighted ILP Objective Function	Optimize Resources Utilization and Load Balance, treated as two independent objectives
Output of a VNR allocation attempt	A single solution or a “blocked” indication	A set of solutions (approximate Pareto Front), or a “blocked” indication

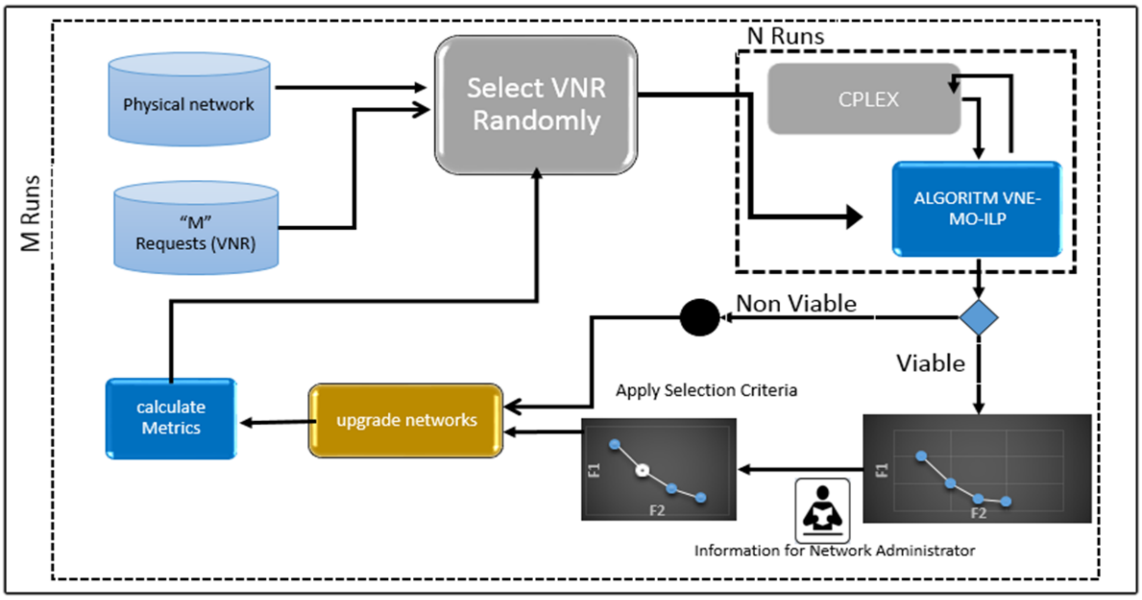

Figure 1: Discrete-event network simulator

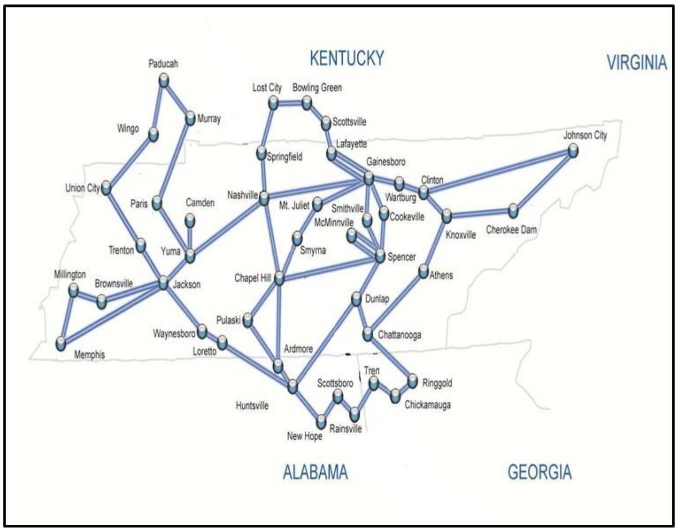

Figure 2: IRIS network topology

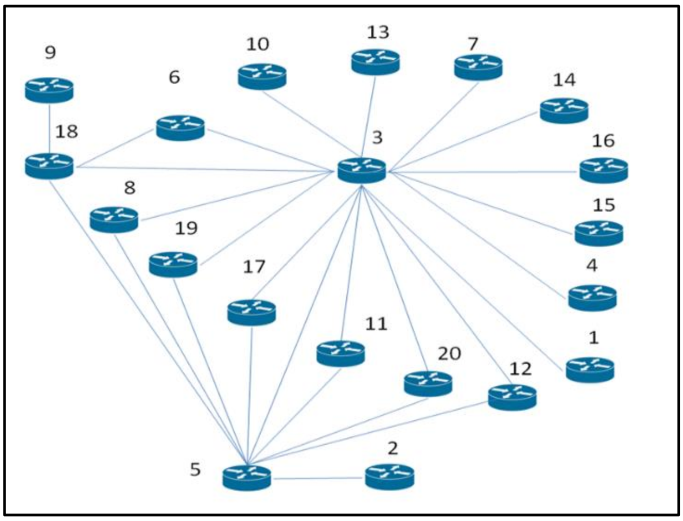

Figure 3: MARNET network topology

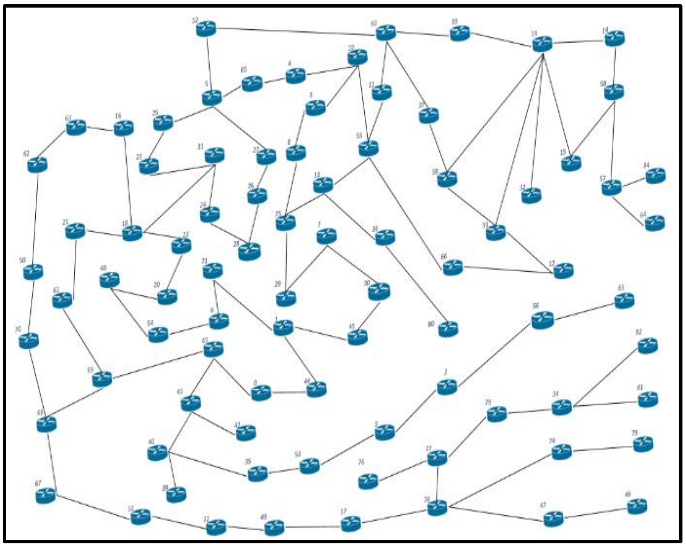

Figure 4: BESTEL network topology

4.3 Experimental Results

We performed groups of experiments taking VNRs with average life time values of: 10, 20, and so on, until 100 time units. We assumed the arrival of a new VNR at each time slot. The experiments are evaluated after 500, 1000, and so on until 5000 discrete time slots, allowing the network to reach its stable state regarding the number of installed VNRs. All simulations were executed in machines with Intel Core (TM) I7-4770 (3.4 GHz) with a RAM of 8 GB.

As was mentioned early, this paper presents a multi-objective approach for the first time in VNE literature. Therefore, a fair comparison with any previous work is difficult to develop. For comparison purposes, we have implemented the VNE-NLF algorithm presented in [8] in our simulation tool, and made an execution of the same instances of the problem. VNE-NLF algorithm was chosen as representative of the state of the art considering its great performance in terms of the same metrics used in this work, and with reasonable execution times. Besides, the VNE formulation used in this work is the same as the one used in [8].

4.3.1 Solution selection Criteria

For experimental purposes, we have considered three possible criteria for the selection of a unique solution from a Pareto Front approximation, in order to assign resources to the considered VNR at each discrete time interval, using this way the selected solution to simulate the situation at the next time slot.

1) The first criterion is to take the solution of the Pareto Front with the least *Assignation Cost SP* (11), and worst *Load Balance* (12) which is called *Left Criterion (LC).*

2) For the second criterion, we take the nearest solution to origin of coordinate axis, looking for a trade-off between both objectives. This criterion will be called *Central Criterion (CC).*

3) Last, we take the solution of the Pareto Front with the largest Assignation Cost but with the best Load Balance *(Right Criterion, RC).*

Figure 5 shows an example of a Pareto Front, corresponding to a single VNR assignation, in which the three criteria of selection are indicated.

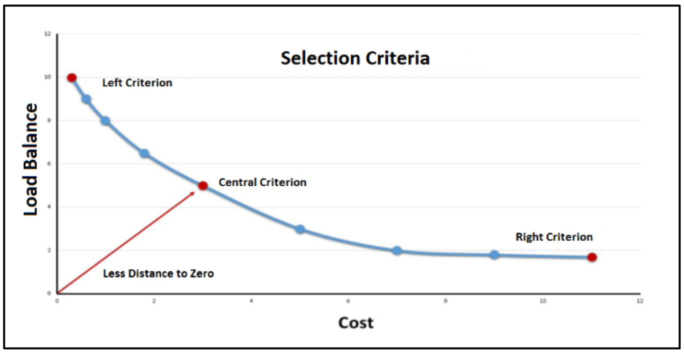

Figure 5: Considered selection criteria from a Pareto Front, for a single VNR allocation, including Left, Central and Right criteria.

The obtained results, comparing the 3 proposed *Selection Criteria*, are shown in Tables 3, 4 and 5 for IRIS, MARNET and BESTEL networks respectively. Presented values are average values of five runs for each instance. It is clear from the presented Tables that the proposed VNE-MO-ILP algorithm outperforms the reference algorithm VNE-NLF presented in [8] in almost all comparisons, except for the values marked in bold and red in Table 4 (MARNET topology), for *Lifetimes* of 60, 70 and 90. In fact, Tables 3 (IRIS topology) and 5 (BESTEL topology) show that our proposed algorithms systematically outperformed the reference algorithm proposed by Melo *et al*. in all the experiments when simultaneously considering both performance metrics: Acceptance Ratio (A^{VNR}) and Revenue/Cost Relation E^{VNR}.

Even when considering the experimental results presented in Table 3 for the *hybrid* MARNET network, the best values of Revenue/ Cost Relation E^{VNR} (considered alone) were reached by the proposed VNE-MO-ILP algorithm, while the reference algorithm VNE-NLF only outperforms VNE-MO-ILP in 3 out of 10 instances. In other words, VNE-MO-ILP finds solutions that dominate in the Pareto sense the solutions calculated by the NVE-NLF algorithm while the latter is not able to find even one solution that dominates the ones calculated by the proposed algorithm.

In short, solutions calculated by the proposed VNE-MO-ILP algorithm are systematically non-dominated (Pareto optimal) and they dominate most solutions calculated with the state of the art algorithm of Melo *et al*. On the contrary, no solution calculated by this state of the art VNE-NLF algorithm is able to dominate (in the Pareto sense) even a single solution calculated with the proposed algorithm, proving this way the advantages of using the VNE-MO-ILP algorithm.

4.3.2 Revenue/Cost Relation

Figures 6, 7 and 8 present experimental results considering this E^{VNR} metric for each of the 3 considered topology as a function of VNR's lifetime, which varies from 10 to 100 time units. Those three figures show that as the lifetime average increases, a larger number of virtual networks remains simultaneously installed, and more physical network resources are utilized.

For the IRIS network (Fig. 6), the three selection criteria outperform the results obtained by the reference algorithm. The best selection criterion is *LC* (*Left Criterion*) as expected, given that it mainly optimizes resources utilization in the mapping of each VNR.

For the MARNET network (Fig. 7), the results using VNE-MO-ILP are also better than the ones using the reference algorithm. However, this time only two criteria obtain better results than the reference algorithm: the *Left Criterion* (*LC*) and the *Central Criterion* (*CC*). For this experiment, the *Right Criterion* (*RC*) only obtains a performance similar to the reference algorithm without significant improvement, as it was the case with the IRIS network shown in Fig. 6.

Table 3: Results – IRIS topology

		VNE-MO-ILP		**VNE-NLF**	
Lifetime	***Criteria***	A^{VNR}	E^{VNR}	A^{VNR}	E^{VNR}
10	LC	100%	0.785	95.6%	0.277
	CC	100%	0.600		
	RC	100%	0.580		
20	LC	100%	0.782	95.6%	0.379
	CC	100%	0.592		
	RC	100%	0.567		
30	LC	100%	0.780	95.6%	0.379
	CC	100%	0.596		
	RC	100%	0.571		
40	LC	100%	0.780	94.7%	0.412
	CC	100%	0.583		
	RC	100%	0.563		
50	LC	100%	0.781	95.2%	0.450
	CC	99.8%	0.595		
	RC	99.9%	0.568		
60	LC	100%	0.780	95.2%	0.450
	CC	99.7%	0.595		
	RC	99.9%	0.559		
70	LC	99.9%	0.778	95.8%	0.482
	CC	99.7%	0.595		
	RC	99.9%	0.556		
80	LC	100%	0.778	96.1%	0.495
	CC	99.9%	0.588		
	RC	99.9%	0.549		
90	LC	99.9%	0.779	96.6%	0.514
	CC	99.8%	0.591		
	RC	99.9%	0.549		
100	LC	99.9%	0.779	94.7%	0.510
	CC	99.9%	0.590		
	RC	99.9%	0.544		

Table 4: Results – MARNET topology

		VNE-MO-ILP		VNE-NLF	
Lifetime	***Criteria***	A^{VNR}	E^{VNR}	A^{VNR}	E^{VNR}
10	LC	100%	0.933	100%	0.717
	CC	100%	0.885		
	RC	100%	0.721		
20	LC	100%	0.929	100%	0.717
	CC	100%	0.882		
	RC	100%	0.716		
30	LC	100%	0.904	100%	0.713
	CC	100%	0.877		
	RC	100%	0.714		
40	LC	99.9%	0.872	100%	0.706
	CC	100%	0.858		
	RC	100%	0.713		
50	LC	99.9%	0.852	100%	0.702
	CC	99.9%	0.839		
	RC	100%	0.713		
60	LC	94.5%	0.835	**96.9%**	0.689
	CC	94.3%	0.827		
	RC	96.7%	0.705		
70	LC	84.5%	0.828	**86.7%**	0.680
	CC	85.1%	0.824		
	RC	85.6%	0.705		
80	LC	76.4%	0.825	77.1%	0.676
	CC	86.9%	0.824		
	RC	77.1%	0.706		
90	LC	70.3%	0.825	**70.4%**	0.679
	CC	70.2%	0.811		
	RC	69.4%	0.708		
100	LC	64%	0.819	63.9%	0.672
	CC	63.3%	0.821		
	RC	63.8%	0.711		

Table 5: Results – BESTEL topology

		VNE-MO-ILP		VNE-NLF	
Lifetime	***Criteria***	A^{VNR}	E^{VNR}	A^{VNR}	E^{VNR}
10	LC	100%	0.626	100%	0.290
	CC	100%	0.502		
	RC	100%	0.487		
20	LC	100%	0.622	100%	0.300
	CC	100%	0.505		
	RC	100%	0.491		
30	LC	100%	0.624	100%	0.329
	CC	100%	0.512		
	RC	100%	0.499		
40	LC	100%	0.621	100%	0.359
	CC	100%	0.517		
	RC	100%	0.504		
50	LC	100%	0.630	100%	0.395
	CC	100%	0.536		
	RC	100%	0.524		
60	LC	100%	0.631	100%	0.418
	CC	100%	0.539		
	RC	100%	0.535		
70	LC	100%	0.636	100%	0.436
	CC	100%	0.554		
	RC	100%	0.547		
80	LC	100%	0.637	100%	0.453
	CC	100%	0.565		
	RC	100%	0.558		
90	LC	100%	0.640	99,9%	0.469
	CC	100%	0.567		
	RC	100%	0.560		
100	LC	100%	0.638	99,8%	0.452
	CC	100%	0.562		
	RC	100%	0.555		

This difference may be explained taking into account that the VNE-NLF algorithm considers, in a single ILP formulation, both components: load balance and resource utilization, without looking at the trade-off between both objectives. On the contrary, proposed VNE-MO-ILP algorithm considers both aspects separately, calculating a Pareto Front approximation, which characterizes this trade-off, achieving in this way better results.

For the BESTEL network, the results presented in Figure 8 show that they are very similar to the experimental results for the IRIS network (Fig. 6). The three selection criteria outperform the reference algorithm. In particular, the *LC* criterion is once more the best option, reaching values of near 80% for E^{VNR}. In fact, VNE-NLF begins with very low values, getting better with more loaded instances, until reaching an E^{VNR} of only 50%, while the other two selection criteria (*CC* and *RC*) get very similar results, in the range of 50-60%.

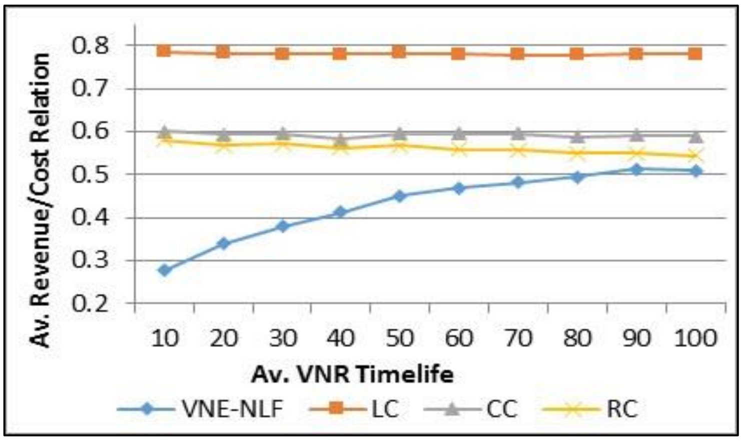

Figure 6: Results – Revenue/Cost Relation, IRIS network

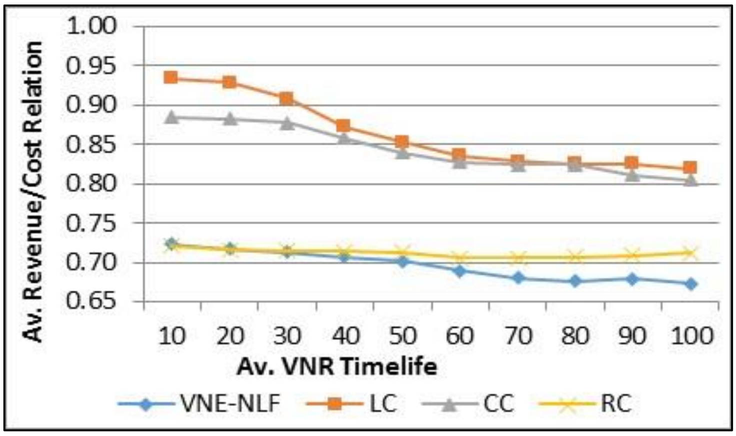

Figure 7: Results – Revenue/Cost Relation, MARNET network

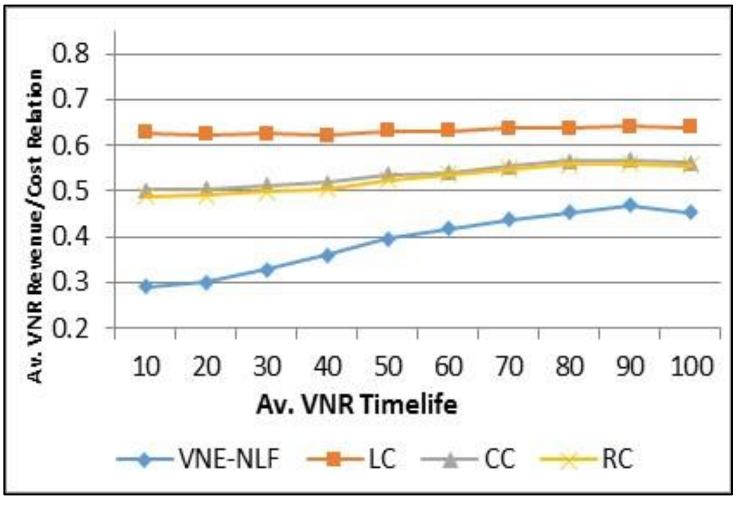

Figure 8: Results – Revenue/Cost Relation, BESTEL network

4.3.3 VNR Acceptance Ratio

Figures 9, 10 and 11 show experimental results for this metric. It can be noticed that for the IRIS network (Fig. 9) the proposed algorithm successfully assigns almost 100% of the VNRs, with all 3 selected criteria (*LC, CC, RC*). However, the reference algorithm only gets up to 95% to 97%, without attending all VNRs. On the contrary, for the MARNET network (Fig. 10) there is no significant difference among the 4 compared algorithms, i.e., the performance of all algorithms are very similar.

For the BESTEL network (see Fig. 11), the proposed algorithm consistently gets 100% in all instances, probably because BESTEL is a large mesh network and the tests could not saturate the network reaching to the limits where we can observe some blocked requests. On the contrary, the reference algorithm could not avoid some blocked requests for loaded instances, proving once more the advantage of using the proposed VNE-MO-ILP algorithm.

These successful results for the proposed VNE-MO-ILP algorithm may be explained considering that it carries out several attempts for mapping each VNR, achieving this way a better search. While some of these candidate solutions may not be feasible, the algorithm tries with almost all possible values of the parameter SP, accomplishing an efficient mapping of almost all VNRs. On the contrary, the reference algorithm only attempts to find a solution once; therefore, it blocks the considered VNR if a solution is not found in this single attempt.

The MARNET network is a star/mesh topology, so it does not offer many options at the time of a blocked request; therefore, the results are very similar with both algorithms, as shown in Fig. 11.

4.3.4 Execution Times

Average execution times for the allocation of a single VNR for the VNE-MO-ILP algorithm took values between two and five seconds while for the VNE-NLF algorithm took about one second, given that it only needs to solve one ILP for each VNR. Logically, it is faster than the VNE-MO-ILP that needs several calls to the ILP Solver, consequently needing a shorter running time.

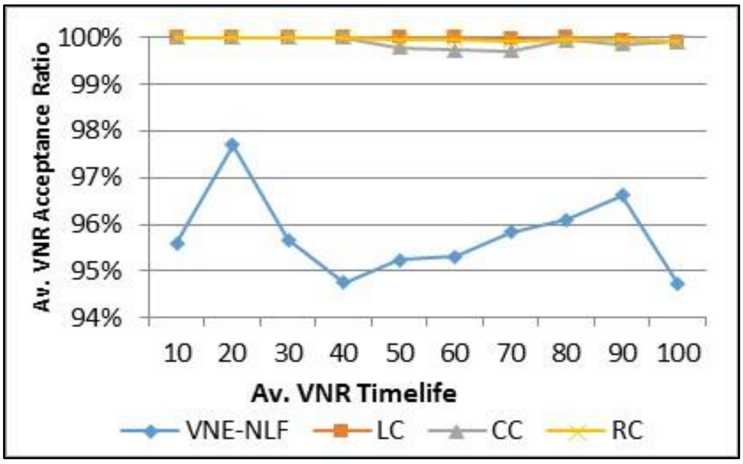

Figure 9: Results – VNR Acceptance Ratio, IRIS network

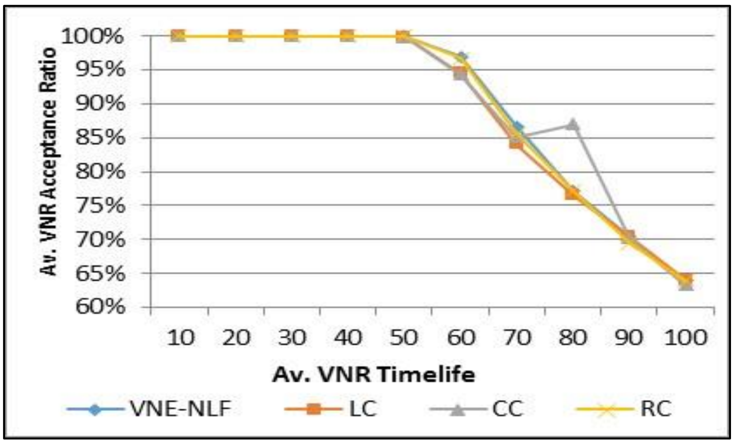

Figure 10: Results – VNR Acceptance Ratio, MARNET network

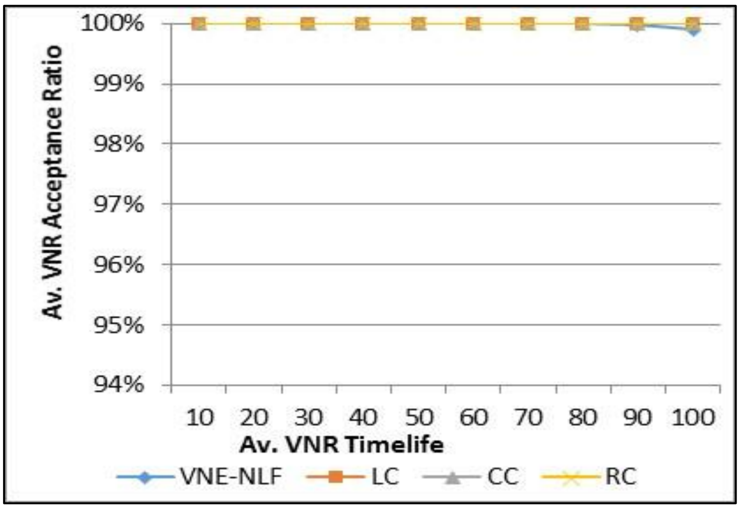

Figure 11: Results – VNR Acceptance Ratio, BESTEL network

Furthermore, in this work we extend the experiments to a large network (BESTEL topology), verifying experimentally that the execution times still remain acceptable. In any case, we propose as future work the development of new efficient algorithms as ACO (Ant Colony Systems), Evolutionary Algorithms (EA), Harmony Search (HS) or PSO (Particle Swarm Optimization), for networks with more than 100 nodes.

5 Conclusions and Future Works

The VNE problem deals with the efficient allocation of resources from a physical network (nodes and links) to virtual network requirements. In this work, the algorithm VNE-MO-ILP *(Virtual Network Embedding-Multi Objective- Integer Lineal Programming)* is proposed for the resolution of the VNE problem in a dynamic (*or online*) context, using a multi-objective approach.

The objectives usually considered for the allocation of a single VNR are two: (i) *Allocation cost*, which is related to the efficient utilization of physical resources, and (ii) *Load Balance*, that looks for a uniform distribution of resources in nodes and links of a physical network.

The VNE-MO-ILP proposed algorithm obtains an approximation of the Pareto Front for each single requirement, through multiple executions of ILP formulations, finding trade-off alternative solutions between the two considered objectives: (1) utilization of physical links and (2) load balancing. By calculating a Pareto front approximation, this algorithm gives the network operator several options for selecting a specific solution. It is also worth mentioning that this work is an extension of [22], which was, in the best of our knowledge, the first work in proposing a multi-objective approach for the VNE problem.

Many experimental tests were performed comparing the efficiency of the proposed algorithm to a state of the art algorithm as VNE-NLF [8], considering three network topologies of medium and large sizes. Presented experimental results clearly prove that the proposed algorithm systematically outperforms the reference algorithm in the two considered *a posteriori* metrics: (i) VNR Acceptance Ratio, and (ii) Revenue / Cost Relation, with reasonable execution time.

As this work highlights, the solutions found by the proposed VNE-MO-ILP algorithm dominate in most cases the solutions calculated with the reference algorithm. In other words, the solutions found by the VNE-MO-ILP algorithm are better (in the Pareto sense) than the ones calculated with the reference algorithm, considering simultaneously both performance metrics. We emphasize the relevant experimental fact that no solution calculated with the proposed algorithm was dominated by any solution on the reference state of the art algorithm.

It is worth remembering that several experiments were performed with three different automatic criteria for the selection of a single solution from each Pareto Front approximation (*Left Criterion - LC, Center Criterion -CC* and *Right Criterion -RC*). Experimental results indicate that the *Left Criterion-LC*, which uses the solution with the best physical resource utilization, achieved the best experimental results. This does not necessarily mean that this is the only possible option, since the network operator could change the chosen option at each stage of the process, using different criteria at different moments, depending on the specific needs at each decision time. Logically, any solution of a Pareto Front cannot be considered a bad decision, given that it is a non-dominated solution (optimal in a Pareto sense); however, in the long run, one choice may result better than another, depending on the dynamics of the VNR requirements.

As it was already said, the utilization of ILP formulations in large networks can be prohibitive because of its execution time. However, in this work we presented experimental results with a network of 84 nodes and 102 links (BESTEL network of Fig. 4) with reasonable execution times. For larger networks, the authors propose the utilization of different multi-objective meta-heuristics for the VNE resolution, as ACO (Ant Colony Systems) and PSO (Particle

Swarm Optimization). Besides, this approach may be used to analyze similar problems, as the VNR (*Virtual Network Reconfiguration*) problem, given that the VNR problem can be easily treated as a simple extension of this work.

Acknowledgments

This work is supported by the *"Vinculación de Científicos y Tecnólogos – Convocatoria 2015"* Program of PROCIENCIA - CONACYT, Paraguay.

References

[1] Feamster, N.; Gao, L.; Rexford,J., "How to lease the Internet in your spare time," ACM SIGCOMM Computer Communication Review, vol. 37, No. 1, pp. 61-64, 2007. DOI: 10.1145/1198255.1198265.

[2] Anderson, T.; Peterson, L.; Shenker, S.; Turner, J., "Overcoming the Internet impasse through virtualization," IEEE Computer Magazine , vol.38, no.4, pp.34-41, April 2005. DOI: 10.1109/MC.2005.136.

[3] Peterson, L; Anderson, T.; Blumenthal, D.; Casey, D.; Clark, D.; Estrin, D.; Evans, J.; Raychaudhuri, D.; Reiter, M.; Rexford, J.; Shenker, S.; Wroclawski, J., "GENI design principles," IEEE Computer Magazine, vol. 39, no. 9, pp. 102-105, Sept. 2006. DOI: 10.1109/MC.2006.307.

[4] University of Waterloo, "User controlled lightpaths project," http://uclp.uwaterloo.ca/.

[5] Tzanakaki, A.; Anastasopoulos, M.; Georgakilas, K.; Buysse, J.; De Leenheer, M.; Develder, C.; Shuping Peng; Nejabati, R.; Escalona, E.; Simeonidou, D.; Ciulli, N.; Landi, G.; Brogle, M.; Manfredi, A.; Lopez, E.; Riera, J.F.; Garcia-Espin, J.A.; Donadio, P.; Parladori, G.; Jimenez, J., "Energy Efficiency in integrated IT and optical network infrastructures: The GEYSERS approach," in Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on , vol., no., pp.343-348, 10-15 April 2011. DOI: 10.1109/INFCOMW.2011.5928835.

[6] Fischer, A.; Botero, J.F.; Till Beck, M.; de Meer, H.; Hesselbach, X., "Virtual Network Embedding: A Survey," inCommunications Surveys & Tutorials, IEEE , vol.15, no.4, pp.1888-1906, Fourth Quarter 2013. DOI: 10.1109/SURV.2013.013013.00155.

[7] Yu, M.; Yi, Y.; Rexford, J.; Chiang, M., "Rethinking virtual network embedding: Substrate support for path splitting and migration," ACM SIGCOMM CCR, vol. 38, no. 2, pp. 17–29, April 2008. DOI: 10.1145/1355734.1355737.

[8] Melo, M.; Sargento, S.; Killat, U.; Timm-Giel, A.; Carapinha, J., "Optimal Virtual Network Embedding: Node-Link Formulation," in Network and Service Management, IEEE Transactions on , vol.10, no.4, pp.356-368, December 2013. DOI: 10.1109/TNSM.2013.092813.130397.

[9] Chowdhury, N.M.M.K.; Boutaba, R., "Network virtualization: state of the art and research challenges," in Communications Magazine, IEEE , vol.47, no.7, pp.20-26, July 2009. DOI: 10.1109/MCOM.2009.5183468.

[10] Lu, J.; Turner, J., "Efficient mapping of virtual networks onto a shared substrate," Washington University in St. Louis, Technical Report, 2006. DOI: 10.7936/K78W3BJ4.

[11] Chowdhury, N.M.M.K.; Rahman, M.R.; Boutaba, R., "Virtual Network Embedding with Coordinated Node and Link Mapping," in INFOCOM 2009, IEEE , vol., no., pp.783-791, 19-25 April 2009. DOI: 10.1109/INFCOM.2009.5061987.

[12] Cheng, X.; Su, S.; Zhang, Z.; Wang, H.; Yang, F.; Luo, Y.; Wang, J., "Virtual network embedding through topology-aware node ranking," SIGCOMM Computer Communication Review, vol. 41, pp. 38–47, April 2011. DOI: 10.1145/19711162.1971168.

[13] The Google Search Engine: Commercial Search Engine founded by the Originators of PageRank, http://www.google.com/, 2003.

[14] L. Gong, Y. Wen, Z. Zhu and T. Lee, "Toward profit-seeking virtual network embedding algorithm via global resource capacity," INFOCOM, 2014 Proceedings IEEE, Toronto, ON, 2014, pp. 1-9. DOI: 10.1109/INFOCOM.2014.6847918.

[15] Lischka, J.; Karl, H., "A virtual network mapping algorithm based on subgraph isomorphism detection," in Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures, pp 81-88, New York, USA, August 2009. DOI: 10.1145/1592648.1592662.

[16] S. Zhang, Z. Qian, J. Wu, S. Lu and L. Epstein, "Virtual Network Embedding with Opportunistic Resource Sharing," in IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 3, pp. 816-827, March 2014. DOI: 10.1109/TPDS.2013.64.

[17] M. C. Luizelli, L. R. Bays, L. S. Buriol, M. P. Barcellos and L. P. Gaspary, "Reconnecting Partitions on Physical Infrastructures: Towards an Expansion Strategy for Efficient Virtual Network Embedding," Computer Networks and Distributed Systems (SBRC), 2014 Brazilian Symposium on, Florianopolis, 2014, pp. 335-343. DOI: 10.1109/SBRC.2014.40

[18] R. Mijumbi, J. Serrat, J. L. Gorricho and R. Boutaba, "A Path Generation Approach to Embedding of Virtual Networks," in IEEE Transactions on Network and Service Management, vol. 12, no. 3, pp. 334-348, Sept. 2015. DOI: 10.1109/TNSM.2015.2459073.

[19] Long Gong; Zuqing Zhu, "Virtual Optical Network Embedding (VONE) Over Elastic Optical Networks," Lightwave Technology, Journal of , vol.32, no.3, pp.450,460, Feb.1, 2014. DOI: 10.1109/JLT.2013.2294389.

[20] Xiujiao Gao; Zilong Ye; Weida Zhong; Chunming Qiao; Xiaojun Cao; Hanjia Zhao; Hongfang Yu; Anand, V., "Multicast service-oriented Virtual Network mapping over Elastic Optical Networks," in Communications (ICC), 2015 IEEE International Conference on , vol., no., pp.5174-5179, 8-12 June 2015. DOI: 10.1109/ICC.2015.7249145.

[21] F. T. Hsu and C. H. Gan, "Resource Allocation with Spectrum Aggregation for Wireless Virtual Network Embedding," Vehicular Technology Conference (VTC Fall), 2015 IEEE 82nd, Boston, MA, 2015, pp. 1-5. DOI: 10.1109/VTCFall.2015.7391117.

[22] Davalos, E.; Aceval, C.; Franco, V.; Baran, B, “A multi-objective approach for virtual network embedding”, in CLEI 2015, XLI Conferencia Latinoamericana en Informática, October 2015. DOI: 10.1109/CLEI.2015.7360004.

[23] Von Lücken, C.; Barán, B.; Brizuela, C., “A survey on multi-objective evolutionary algorithms for many-objective problems”. Computational Optimization and Applications, vol. 58, no. 3, pp. 707-756, 2014. DOI: 10.1007/s10589-014-9644-1.

[24] Even, S.; Itai, A.; Shamir, A., "On the complexity of time table and multi-commodity flow problems," Foundations of Computer Science, 16th Annual Symposium on , pp.184-193, 13-15, October 1975. DOI: 10.1109/SFCS.1975.21.

[25] University of Adelaide. Internet topology Zoo. http://www.topology-zoo.org/index.html, 2013.

[26] Matta, I.: Byers, J.; Medina, A.; Lakhina, A., BRITE topology generator tool. http://www.cs.bu.edu/brite/, 2014.

[27] IBM ILOG CPLEX Optimization Studio. https://www.ibm.com/developerworks/downloads/ws/ilogcplex/, 2014.