

Scalable algorithms for QoS-aware virtual network mapping for cloud services

Minh Bui, Brigitte Jaumard
Concordia University
Montreal (QC), Canada

Isil Burcu Barla Harter
Nokia Siemens Networks
Munich, Germany

Chris Develder
Ghent University – iMinds
Ghent, Belgium

Abstract—Both business and consumer applications increasingly depend on cloud solutions. Yet, many are still reluctant to move to cloud-based solutions, mainly due to concerns of service quality and reliability. Since cloud platforms depend both on IT resources (located in data centers, DCs) and network infrastructure connecting to it, both QoS and resilience should be offered with end-to-end guarantees up to and including the server resources. The latter currently is largely impeded by the fact that the network and cloud DC domains are typically operated by disjoint entities. Network virtualization, together with combined control of network and IT resources can solve that problem. Here, we formally state the combined network and IT provisioning problem for a set of virtual networks, incorporating resilience as well as QoS in physical and virtual layers. We provide a scalable column generation model, to address real world network sizes. We analyze the latter in extensive case studies, to answer the question at which layer to provision QoS and resilience in virtual networks for cloud services.

I. INTRODUCTION

Cloud services have become increasingly popular from the customer’s perspective mainly because of convenience: applications are offered “in the cloud” and thus facilitate access from anywhere on almost any device. Technically, this clearly relies on reasonably high bandwidth connectivity. The core network, carrying the aggregated end user traffic in bulk and providing connectivity towards the large scale data center infrastructures (where the aforementioned services are actually running), is cost effectively realized by optical network technology: we refer to such networks as optical clouds (see [1] for a discussion on the applications that have driven this evolution, and the optical network technology challenges). Traditional network design algorithms, such as the typical routing and wavelength assignment (RWA) strategies, however cannot be directly applied in the context of optical clouds. Fundamentally, this is due to two core principles underlying cloud technologies: *anycast routing* and *virtualization*.

Anycast routing refers to the fact that users do not greatly care about the exact location of the actual servers running the applications they are using. Thus, service providers have some flexibility in deciding where to serve what requests. From the network perspective, this means that the destination of traffic is not fully specified in advance. From the network’s perspective, it implies that the destination of traffic flows is not given a priori. Moreover, clearly the network infrastructure cannot be treated completely independent from the data center infrastructure capacity (since terminating traffic needs to be

served by the data center resources). The joint dimensioning of network and data center infrastructure to resiliently support cloud services has been studied, e.g., in [2].

Virtualization implies that physical infrastructure is logically partitioned in disjoint virtual resources. On the data center side, this means servers are running multiple so-called virtual machines (VMs) that have no access to each other’s resources. Similarly, in recent years the concept of virtualization has also been applied to networks [3]: different virtual networks (VNETs) can be run by independent virtual network operators (VNOs) that make use of the same physical network infrastructure, offered by physical infrastructure providers (PIPs). Both for server and network virtualization, the rationale is to share the same physical resources (thus reducing the capital expenditure for hardware), but still to provide isolation (by logically segregating the services over disjoint (virtual) resources).

Here, we study the provisioning of VNETs for cloud services both resiliently and with assurance of QoS. Requests need to be served by a VNO, who thus needs to allocate server capacity at a particular data center (DC), and provision network connectivity from its customers to their respectively assigned DCs. The VNO’s logical VNET will be provided through a mapping to physical resources offered by a PIP. Furthermore, we will ensure the request’s QoS requirements (i.e., end-to-end delay between source and destination) are respected, and consider 3 classes of virtual resources.

Our novel contributions are:

- Compared to our earlier works adopting column generation in (e.g., [2], [4]) and precursory work on VNET mapping [5] we (i) account for service QoS differentiation, and also (ii) adopt a more detailed/realistic VNO cost model (e.g., accounting for virtual node costs).
- Compared to initial work on QoS-aware mapping [6], we (i) consider anycast instead of unicast demands, (ii) adopt a more realistic delay modelling, and (iii) present a truly scalable column generation based formulation instead of a simple (non-scalable) ILP formulation.
- We demonstrate the near-optimality and scalability of our solution on a 28-node EU topology, thus providing a thorough assessment of the pros and contras of two resilience options in terms of both (i) VNO setup costs, and (ii) physical resource utilization.

II. RELATED WORK

Virtualization of cloud infrastructure has been well investigated, both in terms in network planning [7] (as an offline problem with static traffic) and in terms of traffic engineering [8] (as an online problem with dynamic provisioning), under anycast routing. Those virtual networks also need to be resilient with seamless migration of Virtual Machines (VMs) in order to guarantee an appropriate Quality of Service (QoS) [8].

In addition to the works cited in the previous section and those cited in [4], optimization models for the planning of virtual infrastructure can be found in [7] under the objective of minimizing the power consumption, and in [9] subject to resource consumption minimization and load balancing.

III. RESILIENT VIRTUAL NETWORK MAPPING WITH QoS

We consider the problem of mapping a given set of cloud requests into a virtual network design, such that it is resilient against failures of both the network and data center infrastructure, while respecting the requests' QoS constraints under all circumstances. We formalize this as follows:

Given:

- The network topology, described by
 - $G^{\text{PHY}} = (V^{\text{PHY}}, L^{\text{PHY}})$, the physical network comprising the physical nodes V^{PHY} and interconnecting links L^{PHY} .
 - $G^{\text{VIR}} = (V^{\text{VIR}}, L^{\text{VIR}})$, the virtual network with candidate virtual nodes V^{VIR} , as well as candidate virtual links L^{VIR} . There will be a one-to-one mapping between each virtual node $v' \in V^{\text{VIR}}$ and a single physical $v \in V^{\text{PHY}}$ (thus $V^{\text{VIR}} \subseteq V^{\text{PHY}}$), but multiple candidate virtual links will be considered between the same virtual node pair with mappings to distinct physical paths.
 - $V^{\text{DC}} \subseteq V^{\text{VIR}}$, the set of data center locations.
 - The set of all paths $\pi \in \Pi$ in the physical network corresponding to the mapping of any virtual link $\ell' \in L^{\text{VIR}}$.
- The cloud requests $d \in D$, each one characterized by
 - A source node $\text{SRC}_d \in V^{\text{VIR}}$,
 - The requested bandwidth Δ_d^{BW} ,
 - The requested number of virtual machines Δ_d^{VM} ,
 - The minimal QoS class of the VMs, $q_d \in Q$, and
 - A maximal end-to-end delay (i.e., between source and chosen DC) of δ_d .

Find: For each request $d \in D$, a working (w) and backup (B) data center to use, as well as routes in the virtual network G^{VIR} towards them, such that:

- Each request d can always be served, both in failure-free conditions as well as under any failure scenario,
- The QoS of every request d is respected,
- The total network cost is minimized, and
- The physical network capacity constraints are respected.

Hence, we face a *resilient* virtual network mapping problem. The failures we will protect against will be single failures of either a physical link ($\ell \in L^{\text{PHY}}$), or a complete data center ($v \in V^{\text{DC}}$). We will consider two resiliency approaches: *VNO-resilience* or *PIP-resilience* [10], [5]. As sketched in

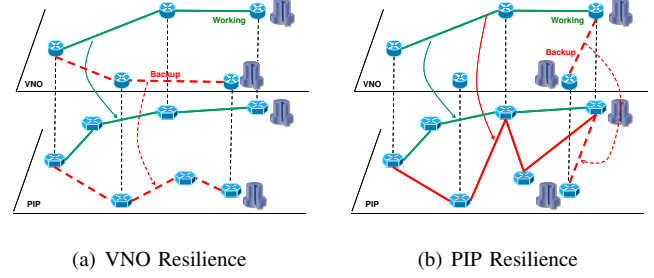


Fig. 1. Two resilience schemes.

Fig. 1, in case of *VNO-resilience*, the protection is handled by the virtual network operator, and requests are rerouted in the virtual network both in case of physical network failure and DC failure. On the other hand, in case of *PIP-resilience*, a virtual link is mapped resiliently to two failure-disjoint paths in the physical network¹. Thus, only in case of a data center failure, an explicit reroute to another data center is required (using an unprotected link). Note however that in reality, the B path will not be exposed to the VNO. Still, the PIP still has to provision it and it will have associated costs. Hence, from a modelling perspective, we do represent it in the VNO layer. Note that we do **not** consider shared protection: bandwidth will not be reused among protection paths that are activated under different failure scenarios. Furthermore, we will assume failure-independent rerouting: for a given request the backup route (and destination) will be the same regardless of the failure affecting the primary route.

The *QoS constraints* associated with a request d are first of all the QoS class of the VMs to be installed, and secondly the end-to-end delay from source node to destination DC. The latter is the sum of the virtual link and node delays. The delay of a virtual link depends on the propagation delay (i.e., the physical path length) and the sum of the delays over the intermediate physical nodes (for which we will use a fixed value, see Section VI). The delay of a virtual node depends on its QoS class: just as VMs, we assume to have the choice between different virtual node types of a given class $q \in Q$, each with their associated forwarding delay ($\delta^{\text{NODE},q}$).

The *cost model* comprises a data center component (the VMs) and a virtual network provisioning part. The data center cost will be the cost of installing VMs:

$C_v^{\text{VM},q}$: the cost per installed VM of class q at DC $v \in V^{\text{DC}}$.

The virtual network cost will be a summation of node and link costs, with a fixed part independent of the traffic volume crossing it, as well as a bandwidth-dependent part:

$C_{\text{SETUP}}^{\text{LINK},\ell'}$: cost of instantiating candidate virtual link $\ell' \in L^{\text{VIR}}$ as a class $q \in Q$ link. In our experiments, this cost will be dependent on both the class q and the link length

¹Remark that this means that in the PIP-resilience case, L^{VIR} may contain multiple parallel links between the same virtual node pair: defining $\pi_{\ell'}^{\text{PW}}$ resp. $\pi_{\ell'}^{\text{PB}}$ as the two paths in the physical layer, parallel virtual link candidates may share the same $\pi_{\ell'}^{\text{PW}}$, or $\pi_{\ell'}^{\text{PB}}$, but not both.

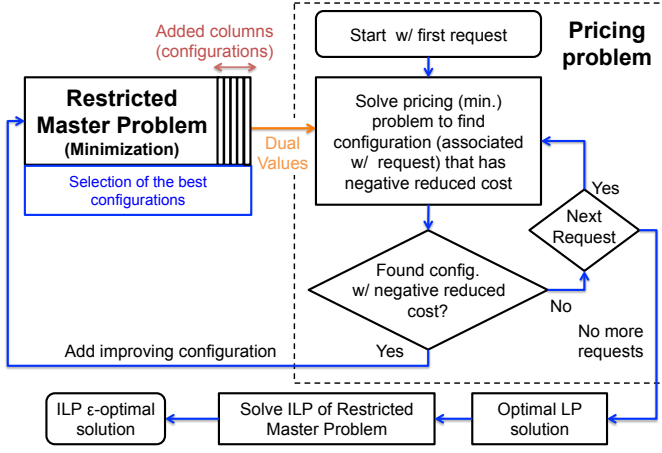


Fig. 2. Decomposition flow chart

$|\ell|$ (see further, Section VI).

C_{ℓ}^{LINK} : cost of using a single unit of bandwidth capacity on a class q virtual link ℓ' .

$C_{\text{SETUP}_v}^{\text{NODE},q}$: cost of instantiating a class q virtual node at v .

$C_v^{\text{NODE},q}$: the cost of forwarding a single unit of bandwidth capacity through a class q virtual node at $v \in V^{\text{VIR}}$.

The *capacity limits* of the physical links and virtual nodes are assumed to be given:

CAP^{LINK} : bandwidth limit on physical link $\ell \in L^{\text{PHY}}$

CAP_v^{NODE} : maximal virtual node capacity at node $v \in V^{\text{VIR}}$

CAP_v^{VM} : maximal VM capacity that is available in the DC at node $v \in V^{\text{DC}}$ (which will in practice depend on the physical server capacity). Note that we assume that the capacity of a single VM instance depends on its class q only, which we will denote as $CAP^{\text{VM},q}$.

IV. COLUMN GENERATION (CG) MODEL: VNO SCHEME

We adopt a column generation (CG) approach to obtain a highly scalable model (e.g., its application in [11], [2]). The model thus is split into a Restricted Master Problem (RMP) and a Pricing Problem (PP), as sketched in Fig. 2. Given a set of given configurations, the RMP decides which ones to select to achieve minimal cost. The PP will be responsible for finding such suitable configurations. PP and RMP will be solved alternately until the optimality condition is satisfied (i.e., no more configurations with a negative reduced cost). An integer solution is obtained by solving the last generated RMP. The approach is scalable because the set of PP configurations will be only a very small fraction of all possible ones. For details on column generation we refer to, e.g., [12].

A. Master Problem

1) *Parameters and variables*: We denote by ℓ a generic physical link and by ℓ' a generic virtual link.

A configuration γ is associated with a particular demand d_γ (where Γ_d is the set of all candidate configurations associated

with d and $\Gamma \supseteq \Gamma_d$ is the set of all configurations) and is characterized by:

- COST_γ , its cost for usage per unit request, which includes the cost for virtual nodes, links, and VMs;
- $p_\ell^\gamma = 1$ if $\ell \in L^{\text{PHY}}$ belongs to the working or backup path;
- $y_v^{\text{NODE},q,\gamma} = 1$ if virtual node v is set as a q class node in configuration, 0 otherwise;
- $y_{\ell'}^{\bullet,q,\gamma} = 1$ if virtual link $\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}$ is set as a q class w/B virtual link in configuration, 0 otherwise ($\bullet \in \{W, B\}$, W working, B backup respectively);
- $y_v^{\text{VM},q,\gamma} = 1$ if connect node v is set as a q class node in configuration, 0 otherwise;
- $\Delta_\gamma^{\text{BW}} = \Delta_{d_\gamma}^{\text{BW}}$ = requested bandwidth for demand d_γ ;
- $\Delta_\gamma^{\text{VM}} = \Delta_{d_\gamma}^{\text{VM}}$ = requested VM resources for demand d_γ .

Physical network parameters:

- $\delta_{\ell'}^q$ = end-to-end delay thresholds for the mapping of a class q virtual link ℓ' .
- $\delta_\ell^{\text{LINK}}$ = delay of physical link ℓ .
- δ^{NODE} = traversal delay of a physical node.
- $\delta^{\text{NODE},q}$ = traversal delay of a class q virtual node.
- L^{VIR} = set of virtual links with are created up to the current iteration of CG.
- $C_{\text{SETUP}_{\ell'}}^{\text{LINK}}$ = setup cost for the logical link $\ell' \in L^{\text{VIR}}$. This setup cost depend on the class of ℓ' and the length of its physical mapping.
- $C^{\text{LINK},q}$ = cost per unit bandwidth, which depends on the class (q) of virtual link.

Variables:

- $z_\gamma = 1$ if configuration γ is selected for provisioning and protecting its associated demand d , 0 otherwise.
- $x_v^{\text{NODE},q} = 1$ if virtual node $v \in V^{\text{VIR}}$ is selected with a q label, 0 otherwise.
- $x_v^{\text{VM},q} = 1$ if connected node $v \in V^{\text{DC}}$ is selected with a q label, 0 otherwise.
- $x_{\ell'}^{\text{LINK}} = 1$ if $\ell' \in L^{\text{VIR}}$ is used in at least one selected configuration, 0 otherwise.

2) Objective function:

$$\min \sum_{\gamma \in \Gamma} \text{COST}_\gamma z_\gamma + \sum_{\ell' \in L^{\text{VIR}}} C_{\text{SETUP}_{\ell'}}^{\text{LINK}} x_{\ell'}^{\text{LINK}} + \sum_{v' \in V^{\text{VIR}}} \sum_{q \in Q} C_{\text{SETUP}_v}^{\text{NODE},q} x_{v'}^{\text{NODE},q}, \quad (1)$$

where the cost of a configuration γ is $\text{COST}_\gamma =$

$$\Delta_d^{\text{BW}} \left[\sum_{\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}} \sum_{q \in Q} C^{\text{LINK},q} (y_{\ell'}^{\text{W},q,\gamma} + y_{\ell'}^{\text{B},q,\gamma}) + \sum_{v \in V^{\text{VIR}}} \sum_{q \in Q} C^{\text{NODE},q} y_v^{\text{NODE},q,\gamma} \right]$$

$$+ \Delta_d^{\text{VM}} \left[\sum_{v \in V^{\text{DC}}} \sum_{q \in Q} c^{\text{VM},q} y_v^{\text{VM},q,\gamma} \right] \quad (2)$$

3) *Constraints:*

$$\sum_{\gamma \in \Gamma_d} z_\gamma \geq 1 \quad d \in D \quad (3)$$

$$M x_{\ell'}^{\text{LINK}} \geq \sum_{\gamma \in \Gamma} p_{\ell'}^\gamma z_\gamma \quad \ell' \in L^{\text{VIR}} \quad (4)$$

$$M x_v^{\text{NODE},q} \geq \sum_{\gamma \in \Gamma} y_v^{\text{NODE},q,\gamma} z_\gamma \quad v \in V^{\text{VIR}}, q \in Q \quad (5)$$

$$\text{CAP}_\ell^{\text{LINK}} \geq \sum_{\gamma \in \Gamma} \Delta_\gamma^{\text{BW}} p_\ell^\gamma z_\gamma \quad \ell \in L^{\text{PHY}} \quad (6)$$

$$\text{CAP}_v^{\text{NODE}} \geq \sum_{\gamma \in \Gamma} \sum_{q \in Q} \Delta_\gamma^{\text{BW}} y_v^{\text{NODE},q,\gamma} z_\gamma \quad v \in V^{\text{VIR}} \quad (7)$$

$$\text{CAP}_v^{\text{VM}} \geq \sum_{\gamma \in \Gamma} \sum_{q \in Q} \Delta_\gamma^{\text{VM}} \text{CAP}^{\text{VM},q} y_v^{\text{VM},q,\gamma} z_\gamma \quad v \in V^{\text{DC}} \quad (8)$$

$$x_v^{\text{NODE},q} \in \{0, 1\} \quad v \in V^{\text{VIR}}; \quad z_\gamma \in \{0, 1\} \quad \gamma \in \Gamma \quad (9)$$

$$x_v^{\text{VM},q} \in \{0, 1\} \quad v \in V^{\text{DC}}; \quad x_{\ell'} \in \{0, 1\} \quad \ell' \in L^{\text{VIR}}. \quad (10)$$

Constraints (3) ensure that each demand d is granted. Constraints (4) count the number of distinct virtual link maps in order to compute the setup cost. Constraints (5) categorize nodes into gold, silver, or bronze group. Constraints (6) (resp. (7), (8)) guarantee that the bandwidth capacity is not exceeded on physical link $\ell \in L^{\text{PHY}}$ (resp. the resource capacity on virtual node $v \in V^{\text{VIR}}$, the VM resource capacity).

B. VNO Pricing problem

To route the network flow on virtual topology we define the set of virtual link candidates as $V^{\text{VIR}} \times V^{\text{VIR}}$

1) *Variables:* The variables of the pricing problem are in one to one correspondence with the following parameters of the master problem (but dropping the γ superscript to simplify the notation): p_ℓ , a_d , $y_v^{\text{NODE},q}$, $y_{\ell'}^{\bullet,q}$, and $y_v^{\text{VM},q}$. Their definition can therefore be easily deduced from the definition of those parameters in the master problem.

In addition, we need the following decision variables:

- $p_{\ell'} = 1$ if $\ell' \in L^{\text{VIR}}$ is used in the configuration.
- $\varphi_{\ell',\ell}^{\text{W}}$ (resp. $\varphi_{\ell',\ell}^{\text{B}}$) = 1 if physical link ℓ is used for mapping virtual link $\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}$ within the working (resp. backup) path
- $y_v^{\text{NODE},q,\bullet} = 1$ if the \bullet path contains v , $\bullet \in \{\text{w}, \text{b}\}$, and v belongs to class q .
- $y_v^{\text{VM},q,\bullet} = 1$ if v is the location of the \bullet DC, $\bullet \in \{\text{w}, \text{b}\}$, and v belongs to class q .
- $b_{\ell',v}^{\bullet} = 1$ if $v \in V^{\text{PHY}}$ belongs to the physical mapping of ℓ' , and $\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}$ is on the \bullet path, $\bullet \in \{\text{w}, \text{b}\}$.
- $y_{\ell'}^{\bullet,q} = 1$ if the \bullet physical mapping of virtual link $\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}$ has a q label, 0 otherwise, $\bullet \in \{\text{w}, \text{b}\}$.

Parameters: $\psi_{\ell',\ell} = 1$ if physical link $\ell \in L^{\text{PHY}}$ is used in the mapping of virtual link $\ell' \in L^{\text{VIR}}$.

2) *Objective:* The objective function of the pricing is straightforwardly derived from the RMP [12].

3) *Constraints:* We need to enforce $p_{\ell'} = 1$ if virtual link $\ell' = (\text{SRC}_{\ell'}, \text{DST}_{\ell'})$ in the configuration under construction is used for either the working or backup path, and this ℓ' has the physical mapping that completely coincides with the mapping of $\ell' \in L^{\text{VIR}}$. Thus we have:

$$p_{\ell'} \equiv p_{\ell'}^{\text{W}} \vee p_{\ell'}^{\text{B}} \quad \text{and} \quad p_{\ell'}^{\bullet} \equiv \bigwedge_{\ell \in L^{\text{PHY}}} p_{\ell',\ell}^{\bullet} \quad \bullet \in \{\text{w}, \text{b}\}$$

After some algebraic manipulations, we get:

$$p_{\ell'}^{\bullet} \leq \psi_{\ell',\ell} \cdot \varphi_{\ell',\ell}^{\bullet} + (1 - \psi_{\ell',\ell}) \cdot (1 - \varphi_{\ell',\ell}^{\bullet}) \quad \bullet \in \{\text{w}, \text{b}\}, \ell \in L^{\text{PHY}}, \ell' \in L^{\text{VIR}} \quad (11)$$

$$p_{\ell'}^{\bullet} + |L^{\text{PHY}}| - 1 \geq$$

$$\sum_{\ell \in L^{\text{PHY}}} \psi_{\ell',\ell} \cdot \varphi_{\ell',\ell}^{\bullet} + (1 - \psi_{\ell',\ell}) \cdot (1 - \varphi_{\ell',\ell}^{\bullet}) \quad \bullet \in \{\text{w}, \text{b}\}, \ell' \in L^{\text{VIR}} \quad (12)$$

$$p_{\ell'} \geq p_{\ell'}^{\bullet} \quad \bullet \in \{\text{w}, \text{b}\} \quad (13)$$

$$p_{\ell'} \leq \sum_{\bullet \in \{\text{w}, \text{b}\}} p_{\ell'}^{\bullet} \quad (14)$$

Next, we have flow constraints to establish the working and the backup *virtual* paths within the anycast paradigm, which involves the selection of the destination connecting nodes for both paths. For all $v \in V^{\text{VIR}}$,

$$\sum_{\ell' \in \text{IN}(v')} \varphi_{\ell'}^{\bullet} = \begin{cases} 1 & \text{if } v = d_{\text{SRC}} \\ 2 \sum_{q \in Q} y_v^{\text{NODE},q,\bullet} - \sum_{q \in Q} y_v^{\text{VM},q,\bullet} & \text{otherwise.} \end{cases} \quad (15)$$

Constraints (16) manage the flow on the *physical* network:

$$\sum_{\ell \in \text{IN}(v)} \varphi_{\ell',\ell}^{\bullet} = \begin{cases} \varphi_{\ell'}^{\bullet} & \text{if } v = \ell'_{\text{SRC}} \text{ or } v = \ell'_{\text{DST}} \\ 2 b_{\ell',v}^{\bullet} & \text{otherwise} \end{cases} \quad v \in V, \ell' \in V^{\text{VIR}} \times V^{\text{VIR}}. \quad (16)$$

Constraints (17) check if a physical link is used in a configuration. Since $p_\ell \leq 1$, it also enforces the disjointness of physical mapping of working and backup virtual paths for each request:

$$p_\ell = \sum_{\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}} \left(\varphi_{\ell',\ell}^{\text{W}} + \varphi_{\ell',\ell}^{\text{B}} \right). \quad (17)$$

Each configuration, i.e., demand/service, has one primary DC and one backup DC:

$$\sum_{q \in Q: q \geq q_d} \sum_{v \in V^{\text{DC}}} y_v^{\text{VM},q,\bullet} = 1 \quad (18)$$

$$y_v^{\text{VM},q,\bullet} = 0 \quad q \in Q: q < q_d \quad (19)$$

$$\sum_{q \in Q} (y_v^{\text{VM},q,W} + y_v^{\text{VM},q,B}) \leq 1 \quad v \in V^{\text{DC}}. \quad (20)$$

Each selected virtual node should be gold, silver or bronze:

$$\sum_{q \in Q} y_{d_{\text{SRC}}}^{\text{NODE},q,\bullet} = 1 \quad \bullet \in \{W, B\} \quad (21)$$

$$M \cdot y_v^{\text{NODE},q} \geq y_v^{\text{NODE},q,W} + y_v^{\text{NODE},q,B} \quad v \in V^{\text{VIR}} \quad (22)$$

First, we compute the end-to-end delay for each virtual link $\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}$:

$$\delta_{\ell'}^{\text{LINK},\bullet} = \sum_{\ell \in L^{\text{PHY}}} \varphi_{\ell\ell'}^{\bullet} (\delta_{\ell}^{\text{LINK}} + \delta_{\ell}^{\text{NODE}}) \quad (23)$$

Virtual links are labeled with gold/silver/bronze categories accordingly to their end-to-end delay in comparison with the best end-to-end delay between two ends of a virtual link $\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}$:

$$M \cdot (y_{\ell'}^{\bullet,G} + 1 - \varphi_{\ell'}^{\bullet}) \geq \delta_{\ell'}^G - \delta_{\ell'}^{\text{LINK},\bullet} - \delta_{\ell'}^{\text{NODE}} \quad (24)$$

$$M \cdot (y_{\ell'}^{\bullet,G} + y_{\ell'}^{\bullet,S} + 1 - \varphi_{\ell'}^{\bullet}) \geq \delta_{\ell'}^S - \delta_{\ell'}^{\text{LINK},\bullet} - \delta_{\ell'}^{\text{NODE}} \quad (25)$$

$$\sum_{q \in Q} y_{\ell'}^{\bullet,q} = \varphi_{\ell'}^{\bullet} \quad (26)$$

The delay requirement for the request must be satisfied by both working and backup path:

$$\sum_{v \in V^{\text{VIR}}} \sum_{q \in Q} y_v^{\text{NODE},\bullet,q} \delta_{\text{NODE},q} + \sum_{\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}} \delta_{\ell'}^{\text{LINK},\bullet} + \sum_{\ell' \in V^{\text{VIR}} \times V^{\text{VIR}}} \varphi_{\ell'}^{\bullet} \delta_{\ell'}^{\text{NODE}} \leq \delta_d \quad (27)$$

$$\delta_{\ell'}^{\text{LINK},\bullet} \geq 0; \quad \bullet \in \{W, B\}, \quad \ell' \in V^{\text{VIR}} \times V^{\text{VIR}} \quad (28)$$

All other variables are binary.

V. COLUMN GENERATION (CG) MODEL: PIP SCHEME

The master problem for the PIP scheme is identical to that of the VNO scheme. However, the pricing problem needs to be modified to accommodate the PIP characteristics in the definition of a configuration:

- The backup path **B** now connects the primary DC and the backup DC (see Fig. 1).
- Each virtual link has two physical link-disjoint paths connecting two end points.
- The delay for a virtual link is set as the delay of the longest of its two physical paths: whenever one path gets disconnected, the traffic will be switched (by the PIP) to the other, and the delay constraint must still be satisfied.



Fig. 3. NobelEU network with 4 DC locations indicated with a star symbol.

- The request's delay constraints must be satisfied for the concatenated paths going first from the source to the primary DC, then to the backup DC: in case of the failure of the primary DC, the traffic will follow that path to reach the backup DC.

Due to space limitations, we omit the full mathematical model.

VI. NUMERICAL EXPERIMENTS

A. Data instances

We conducted experiments on the NobelEU network with 28 nodes and 41 undirected links (see Fig. 3). We randomly generated between 10 and 80 requests, each with a bandwidth requirement randomly generated in $\{1 \dots 9\}$ and a number of virtual machines randomly generated in $\{1, 2, 3\}$. We consider 4 DC locations (see Fig. 3), where each DC has a computation limit of 300 units. The bandwidth limit of each virtual node is 200 bandwidth units, the capacity limit of each physical link is 100 units. Virtual links are classified according to their length: gold (resp. silver) links have a length less than 1.25 (resp. 1.50) times that of the shortest path between two endpoints. The delay requirement for requests depends on their QoS class (gold, silver, bronze), i.e., 16, 22, 30 ms respectively. Other cost parameters are presented in Table I. Note that the cost units are arbitrary, we only pay attention to their relative values.

The LP/ILP programs from our models have been implemented using OPL and solved using IBM ILOG CPLEX 12.6, running on a 4-core 2.2 GHz AMD Opteron 64-bit processor.

B. Results

We investigated the distribution of the costs for different quality of services, for a given distribution of the services among the gold, silver and bronze ones (10%, 30%, 60%). Results are presented in Fig. 4. For both models, the cost obviously increases with the number of requests increases, but the cost distribution on gold/silver/bronze resource classes is not exactly at the gold/silver/bronze demand split. This stems from the cost structure that encourages to reuse existing

Table I
COST PARAMETERS

Parameters	Cost
Virtual node setup cost (gold, silver, bronze)	10, 6, 4
Virtual node bandwidth unit cost (gold, silver, bronze)	5, 3, 2
Virtual link setup cost (gold, silver, bronze)	10, 6, 4
	+ 10 × physical hopcount
Virtual link bandwidth unit cost (gold, silver, bronze)	5, 3, 2
Virtual machine unit cost (gold, silver, bronze)	5, 3, 2
Delay of a physical node	1
Delay of a physical link	1
Delay of a logical node (gold, silver, bronze)	2, 3, 5
Capacity of a virtual machine (gold, silver, bronze)	5, 3, 2

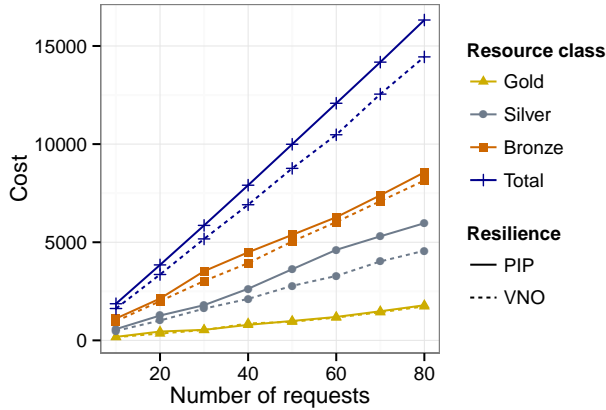


Fig. 4. Cost Distribution

virtual nodes and links when possible (thus possibly using non-shortest paths for traffic, if the delay constraints allow it).

We observe that the overall cost of the PIP scheme is higher than the cost of the VNO scheme. There is a difference of about 10%. This is due to the greater flexibility of the VNO model for selecting the best DC, while in the PIP scheme, one must select the best DC location subject to the condition that both the working and the backup paths must have the same endpoints.

The major difference between VNO- and PIP-resilience stems from the Silver, and to a lesser extent the Bronze resource class, while the cost of Gold resources is almost identical. In the PIP-resilience case, the physical hopcount of virtual links includes both the working and backup mapping and hence is more expensive than a virtual link in the VNO case: thus, there is a higher incentive to try and share them, which becomes easier if the paths in the virtual layer are multi-hop ones (as illustrated in Fig. 1). Bronze links are high delay and hence less likely to be feasible to reuse (or if split into subparts, these sub-parts become Silver because of the reduced virtual link delay). Gold links are there to keep the delay under control and hence there are few opportunities to

split them without violating the delay for the request(s) they support. Thus, the cost increase largely falls down to the Silver network resources.

VII. CONCLUSION

We developed a quite comprehensive model in terms of Quality of Service for the design of resilient logical topologies in clouds, considering two different resilience schemes (VNO vs PIP). This model is significantly more scalable than the previous model of Barla *et al.*, in addition to be more realistic. In future work, we plan to investigate different cost policies, and their consequences on the bandwidth usage.

ACKNOWLEDGMENT

B. Jaumard was supported by NSERC (Natural Sciences and Engineering Research Council of Canada) and by a Concordia University Research Chair (Tier I).

REFERENCES

- [1] C. Devellder, M. Leenheer, B. Dhoedt, M. Pickavet, D. Colle, F. Turck, and P. Demeester, "Optical networks for grid and cloud computing applications," in *Proceedings of the IEEE*, vol. 100, May 2012, pp. 1149 – 1167.
- [2] C. Devellder, J. Buysse, B. Dhoedt, and B. Jaumard, "Joint dimensioning of server and network infrastructure for resilient optical grids/clouds," *IEEE/ACM Transactions on Networking*, pp. 1–16, Oct. 2013.
- [3] N. Mosharaf, K. Chowdhury, and R. Boutaba, "A survey of network virtualization," *Journal Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 54, pp. 862–876, april 2010.
- [4] M. Bui, B. Jaumard, C. Cavdar, and B. Mukherjee, "Design of a survivable VPN topology over a service provider network," in *Workshop on Design of Reliable Communication Networks - DRCN*, March 4-7 2013.
- [5] I. Barla, D. Schupke, M. Hoffmann, and G. Carle, "Optimal design of virtual networks for resilient cloud services," in *Workshop on Design of Reliable Communication Networks - DRCN*, Budapest, Hungary, March 2013.
- [6] A. Basta, I. Barla, M. Hoffmann, and G. Carle, "QoS-aware optimal resilient virtual networks," in *IEEE International Conference on Communications - ICC*, Budapest, Hungary, June 2013, pp. 1–5.
- [7] K. Georgakilas, A. Tzanakaki, M. Anastasopoulos, and J. Pedersen, "Converged optical network and data center virtual infrastructure planning," *Journal of Optical Communications and Networking*, vol. 4, no. 9, pp. 681–691, Sept. 2012.
- [8] F. Hao, T. V. Lakshman, S. Mukherjee, and H. Song, "Enhancing dynamic cloud-based services using network virtualization," *Newsletter ACM SIGCOMM Computer Communication Review*, vol. 40, pp. 67–74, Jan. 2010.
- [9] M. Melo, S. Sargento, U. Killat, A. Timm-Giel, and J. Carapinha, "Optimal virtual network embedding: Node-link formulation," *IEEE Transactions on Network and Service Management*, vol. 10, pp. 356–368, Dec. 2013.
- [10] M. Bui, B. Jaumard, and C. Devellder, "Anycast end-to-end resilience for cloud services over virtual optical networks (invited)," in *IEEE International Conference on Transparent Optical Networks - ICTON*, Cartagena, Spain, June 2013, pp. 1–4.
- [11] A. Shaikh, J. Buysse, B. Jaumard, and C. Devellder, "Anycast routing for survivable optical grids: Scalable solution methods and the impact of relocation," *Journal of Optical Communications and Networking*, vol. 3, pp. 767–779, 2011.
- [12] V. Chvatal, *Linear Programming*. Freeman, 1983.