Open access • Journal Article • DOI:10.1364/JOCN.9.000C25

Exploring the benefit of rerouting multi-period traffic to multi-site data centers [Invited] — Source link

Ting Wang, Brigitte Jaumard, Chris Develder

Institutions: Concordia University, Ghent University

Published on: 01 Apr 2017 - IEEE\/OSA Journal of Optical Communications and Networking (Optical Society of America)

Topics: Static routing, Backup, Column generation, Network topology and Anycast

Exploring the benefit of rerouting multi-period traffic to multi-site data centers

Ting Wang, Brigitte Jaumard, *Senior Member, IEEE* and Chris Develder, *Senior Member, IEEE*

(Invited Paper)

***Abstract*—In cloud-like scenarios, demand is served at one of multiple possible data center (DC) destinations. Usually, which DC exactly is used can be freely chosen, which leads to an anycast routing problem. Furthermore, the demand volume is expected to change over time, e.g., following a diurnal pattern. Given that virtually all application domains today heavily rely on cloud-like services, it is important that the backbone networks connecting users to the DCs is resilient against failures. In this paper, we consider the problem of resiliently routing multi-period traffic: we need to find routes to both a primary DC and a backup DC (to be used in case of failure of the primary one, or the network connection towards it), and also account for synchronization traffic between primary and backup DCs. We formulate this as an optimization problem and adopt column generation, using a path formulation in two sub-problems: the (restricted) master problem (RMP) selects "configurations" to use for each demand in each of the time epochs it lasts, while the pricing problem (PP) constructs a new "configuration" that can lead to lower overall costs (which we express as the number of network resources, i.e., bandwidth, required to serve the demand). Here, a "configuration" is defined by the network paths followed from the demand source to each of the two selected DCs, as well as that of the synchronization traffic in between the DCs. Our decomposition allows for PPs to be solved in parallel, for which we quantitatively explore the reduction of the time required to solve the overall routing problem. The key question that we address with our model is an exploration of the potential benefits in rerouting traffic from one time epoch to the next: we compare several (re)routing strategies, allowing traffic that spans multiple time periods to (i) not be rerouted in different periods, (ii) only change the backup DC and routes, or (iii) freely change both primary and backup DC choices and routes towards them.**

***Index Terms*—Networks, Assignment and routing algorithms, Network survivability, Anycast routing, Column generation**

I. Introduction

OPTICAL networks have enabled the increased reliance of both businesses and end users on data centers (DCs) to serve their applications and content, in particular due to the proliferation of cloud technologies [3]. Given the low latencies and high bandwidth capacities of that (optical) networking technology, the exact location of the DC serving a particular request in many cases has become largely irrelevant. Indeed, in cloud-like scenarios, users typically do not care where exactly their request for processing or storage is served. From a network routing optimization perspective, this introduces an extra degree of freedom: providers can more or less freely decide what DC to use, among several that can be geographically dispersed. This amounts to what is commonly referred to as "anycast routing": for a given service request, originating from a known node in the network topology, the destination is not fixed a priori, but rather can be chosen out of a set of candidate destinations. That anycast principle can furthermore be exploited for resiliency purposes: when a DC, or the network connection towards it, is affected by a failure, backup can be provided at an alternate DC at a different location. Previous work has studied quantifying the potential benefits in terms of reducing resource requirements (in terms of both network and server capacities) through relocation with anycast routing for static traffic (e.g., [4]).

In the current paper, we rather focus on time-varying traffic. Specifically, we consider the case where routing, and thus also DC selection, can be revised at discrete points in time: we assume the volume of service requests to vary over time, which we assume to be divided in multiple periods. Compared to our previous work in this area, i.e., [5], we provide the following contributions:

- A new column generation model that is path-based rather than link-based (Section III),
- A more extensive set of experiments that also consider variations in the choice of DC locations (Section IV), and
- An exploration of the effect of parallel execution of multiple so-called pricing problems (PPs) (Section IV-C).

Note that the work presented here is an extension of our initial summaries thereof at conference venues [1], [2]. In particular, we here provide

- The full mathematical models, listing both the restricted master problem (RMP) and PP formulations (Section III), and
- More details on the parallel execution results, in terms of the number of configurations generated by the PPs (Section IV-C).

Before detailing the problem statement (Section II), full model details (Section III), and experimental case study results (Section IV), we now first highlight related work. We will summarize the paper's conclusions in the final Section V.

A. Related work

The core idea in this paper is to possibly relocate requests to alternate destination data centers (DCs), if that proves beneficial in terms of (network) resource requirements. This assumes that we are dealing with so-called *anycast routing*,

T. Wang and B. Jaumard are with CSE, Concordia University, Montréal, Quebec, Canada. E-mail: bjaumard@cse.concordia.ca

C. Develder is with Ghent University – imec, IDLab, Dept. of Information Technology, Ghent, Belgium. E-mail: chris.develder@ugent.be

Note that smaller portions of this work were presented at ICTON 2016 [1] and NETWORKS 2016 [2].

Manuscript submitted October 17, 2016. Manuscript revised January 17, 2017. Accepted on January 24, 2017. This final version was created February 8, 2017.

which amounts to finding a path from a source to a destination to be chosen among a given set of candidate destinations, while minimizing a certain cost (e.g., bandwidth resource requirements). Note that this concept of anycast routing is more general than the case of optical circuit switched networks, which we assume in the current paper (e.g., see [6] in case of IP, or [7] in optical burst switching, OBS). In optical circuit-switched (OCS) networks, it amounts to the so-called anycast routing and wavelength assignment (ARWA) problem: we have to find wavelength paths and minimize, e.g., the total number of wavelengths used summed over all network links, and/or the load on the links. For the case where all requests are given at once, and are assumed to be static (i.e., do not vary over time), we refer to a more in-depth overview of ARWA literature in [4, Section II].

In the current paper we will consider time-varying traffic, assuming that traffic varies from one period to the next: we consider discrete points in time at which traffic volumes change, i.e., new requests need to be served while old ones are terminated. In the traditional setting of unicast traffic with fixed end points (as opposed to our anycast case), some works have studied the value of rearranging paths over time. For example, simulation experiments in case of wavelength division multiplexing (WDM) networks reported bandwidth savings of 10% when adopting sub-reconfiguration (with pre-computed backup paths) to rearrange paths when traffic changes [8]. Other works investigated protection schemes with either pre-emption or multiple protection paths, yet without reconfiguring backup paths [9], [10]. Here, we will consider changing both the primary working paths and/or (only) backup paths.

To the best of our knowledge, [11] was the first to study resilient multi-period anycast traffic routing.[1] Still, that work adopted an iterative approach, solving a single transition from one period to the next. Since then, we have developed optimization models to jointly optimize the routing (in terms of both primary and shared backup paths) over multiple periods together. As stated before, we reported initial results in the short conference papers [5], [2]. Next, we will introduce the exact problem statement and then disclose the full model details before reporting on experiments.

II. Problem statement

A. Overall problem

The problem we consider is a multi-period anycast routing problem, which we can formally state as follows. **Given**

- the *network topology* in terms of network nodes (e.g., optical cross-connects, OXCs) and fiber links interconnecting them, as well as the locations of the data centers (DCs) which constitute the candidate destinations, and
- the *service requests*, specified by the (i) the source node they originate from, (ii) their resource requirements, which we will express as unit demands representing an amount of bidirectional bandwidth to provide from the source to a DC to be chosen among the candidate DCs, and (iii) the duration (or holding time) they last, expressed as 1 or more consecutive periods from a given starting time,

find for each request, and each of the time periods it lasts, the routes from (i) its source to a selected primary DC, (ii) its source to an alternate backup DC, and (iii) between the primary and backup DCs, **such that** the total amount of required network resources, counted as the bandwidth crossing each link, summed over all links, is minimized and each request remains operational under given failure scenarios. For the latter, we will consider protection against single link or single DC site failures. The specific resilience strategy is detailed next.

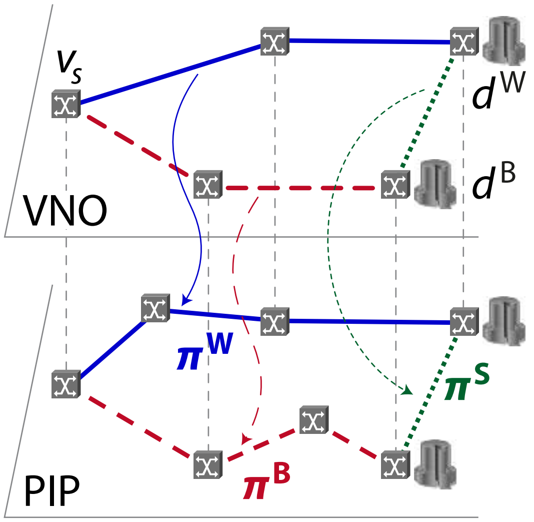


Fig. 1: The VNO-resilience scheme. (VNO: Virtual Network Operator; PIP: Physical Infrastructure Provider.)

B. Resilience strategies

We will treat servicing requests as a mapping of a virtual network, as sketched in Fig. 1: the virtual topology to set up comprises paths interconnecting three nodes, i.e., the given source node and two data centers (DCs) to be chosen among the given candidate ones. Three paths need to be set up, the first being the working path (π^{W}) that routes the services from their source node (v_S) towards the primary DC (d^{W}). Second, the protection path (π^{B}) connects the source towards the backup DC (d^{B}). To ensure resilience against network failures, π^{W} and π^{B} need to be disjoint in their physical layer mapping. The third path is the synchronization path (π^{S}) that connects primary and backup DCs, to handle migration and failure routing requirements when a DC failure occurs (by rerouting the primary d^{W} to backup d^{B}). Under the assumptions that (A1) the backup DC has a different location than the primary DC, (A2) π^{W} and π^{B} are link disjoint and, (A3) π^{W} and π^{S} are link disjoint, protection is guaranteed against any single link failure and any single DC failure. Note that we will consider sharing of backup resources: the capacity allocated for the backup paths π^{B} will only be used under failure conditions, and hence the same bandwidth can be reused by

[1] Note that other works also have considered multi-period traffic when optimizing routing in optical networks, e.g., to minimize the electricity bill [12].

other backup paths $\pi^{\mathrm{B}'}$ as long as both are not required to operate simultaneously (i.e., the respective primary paths π^{W} and $\pi^{\mathrm{W}'}$ are failure disjoint). Further, we note that we will assume the synchronization bandwidth to be allocated on π^{S} will be proportional to the bandwidth to serve the request (on either π^{W} or in case of failures π^{B}). Further details on these assumptions will be reflected in the mathematical model discussed in Section III.

C. Rerouting strategies

As specified in the problem statement, the objective of the routing choice will be to minimize the amount of required network resources to serve all requests. Besides the development of a model to solve that problem, we are primarily interested in assessing whether or not it makes sense to reroute traffic requests from one period to the next. Hence, we will compare three rerouting strategies:

(I) the baseline *Scenario I* fixes each request to the same routing configuration for all time periods of its holding time,

(II) *Scenario II* still keeps the same working path over all periods, but allows to change the backup and/or synchronization paths from one period to the next, and finally

(III) *Scenario III* permits complete rerouting of a request, including the working path.

Note that changing the working path from one period to the next in the latter Scenario III can clearly impact the service quality experienced when switching these routes (e.g., out-of-order delivery of traffic if the new path happens to be shorter, or a small disruption if the make-before-break principle is not followed).[2]

III. Mathematical model

Given the combinatorial growth of the number of possible configurations, solving a single exact optimization problem such as a traditional integer linear programming (ILP) model is not scalable to problem sizes of interest in practice. Hence, we resort to a column generation approach: the overall optimization problem is subdivided into two parts, a so-called restricted master problem (RMP) and an accompanying pricing problem (PP). The task of the RMP is to find the optimal combination of "configurations", selected from a (restricted) set C, while the PP will find/create new configurations to add to the pool C such that the optimization problem's objective value improves. More concretely, in our model a "configuration" is associated with a request source node v_{S} and comprises the virtual topology that we need to map, as sketched in Fig. 1: (i) a primary path π^{W} towards a chosen data center $v_{\mathrm{D}}^{\mathrm{W}}$, (ii) a backup path π^{B} towards an alternate data center $v_{\mathrm{D}}^{\mathrm{B}}$, and (iii) a synchronization path π^{S} interconnecting the two data centers.

Next we will detail the mathematical formulation of both the RMP (Section III-A) and the PP (Section III-B), and the solution scheme that we will use to iterate among both (Section III-C). Note that our model assumes that we can aggregate all traffic originating at the same source: we do not individually model unit requests that share the same source and holding time. This improves scalability of the model compared to, e.g., our earlier work [5].

A. Restricted Master Problem (RMP)

The master problem basically formulates the problem as we phrased it in Section II-A: it decides for each request, in each time period it covers, what configuration (i.e., combination of primary, backup and synchronization paths) to use to serve it. The master problem takes a pre-established set of candidate configurations C as input, and will be "restricted" in the sense that this candidate set will not comprise an exhaustive enumeration of all possible configurations.

The following are given **input parameters** of the overall problem:

$G = (V, L)$ is the undirected graph representing the optical backbone network, where V is the set of all nodes (optical switches) and L the set of optical fiber links interconnecting them (with a priori unlimited capacity). The subset $V_{\mathrm{S}} \subset V$ represents the source nodes of the requests, while $V_{\mathrm{D}} \subset V$ is the set of given data centers (being candidate destinations).

T is the set of discrete time periods (e.g., each hour of the day) for the multi-period time interval we want to solve the routing problem for. The set of all time periods except the first one will be denoted as T'.

$\Delta_{v,t}$ is the amount of bandwidth required by the requests originating from source node $v \in V_{\mathrm{S}}$ during period $t \in T$.

C is the set of all configurations (i.e., combinations of working, backup and synchronization paths interconnecting source nodes and data centers) that we will consider using for fulfilling the requests, with a subset $C_v \subset C$ grouping those associated with a particular source node $v \in V_{\mathrm{S}}$

δ is the scaling factor that relates the required synchronization bandwidth[3] to the full traffic bandwidths (Δ_v), i.e., we typically will have $\delta < 1$.

The **decision variables** in the RMP are the following:

$z_{c,t}$ is the prime decision variable that indicates which volume of requests that will be served by configuration c during time interval t. (Obviously, this will only be for requests that originate from the source node v_{S} of the configuration.)

BW_ℓ is an *auxiliary variable* that counts the bandwidth required on link ℓ, which will be the maximum over any period.

$\beta^{\mathbf{W}}_{\ell,t}, \beta^{\mathbf{B}}_{\ell,t}, \beta^{\mathbf{S}}_{\ell,t}$ all are *auxiliary variables* as well, quantifying the link bandwidth required on link ℓ during period t for respectively working (W), backup (B) and synchronization (S) paths.

$\gamma^{\mathbf{W}}_{\pi,t}, \gamma^{\mathbf{B}}_{\pi,t}, \gamma^{\mathbf{S}}_{\pi,t}$ also are *auxiliary variables* that will sum the amount of bandwidth carried during period t on a given

[2] In case of make-before-break, having both the old and the new path set-up at the same time (for a short while) may also further increase network capacity requirements (slightly) beyond what our model estimates. We do not further address this issue in the current paper, since we are mainly interested in assessing the maximal net capacity benefit that Scenario III could theoretically achieve compared to Scenario II.

[3] We can easily make this factor dependent on the source node, but refrain from doing so in the current model, for the sake of not making the notation overly complex.

path π for respectively working (W) , backup (B) and synchronization (S) purposes.

Additionally, the RMP uses the following parameters that define the configuration, which will be **PP results**:

$p^{\mathrm{W}}_{c,\ell}, p^{\mathbf{B}}_{c,\ell}, p^{\mathrm{S}}_{c,\ell}$ are binary parameters that are 1 if link ℓ is traversed by respectively the working (W), backup (B) and synchronization (S) path of configuration c, and 0 otherwise.

$a^{\mathrm{W}}_{c,v}$ is again a binary that is 1 if the data center node $v \in V_{\mathrm{D}}$ is used as primary data center in configuration c, and otherwise 0.

Furthermore, we will denote

Π_v as the set of all paths from all (current) configurations associated with source node $v \in V_{\mathrm{S}}$, and

$C^{\mathbf{W}}_{\pi}, C^{\mathbf{B}}_{\pi}, C^{\mathbf{S}}_{\pi}$ as the set of configurations that have path π, respectively as a working (W), backup (B) or synchronization (S) path.

These sets will also directly follow from the configurations as found by the PP.

1) Scenario III: For the least restrictive case of Scenario III, where we do not have any limitations on reconfiguring routes from one time period to the next, the full RMP model is specified by equations (1)–(10), which we explain below.

$$\min \sum_{\ell \in L} \mathrm{BW}_{\ell} \; \|\ell\| \tag{1}$$

subject to

$$\mathrm{BW}_{\ell} \geq \beta^{\mathrm{W}}_{\ell,t} + \beta^{\mathrm{B}}_{\ell,t} + \beta^{\mathrm{S}}_{\ell,t} \qquad t \in T \tag{2}$$

$$\sum_{c \in C_v} z_{c,t} \geq \Delta_{v,t} \qquad v \in V_{\mathrm{S}}, t \in T \tag{3}$$

$$\sum_{c \in C} p^{\mathrm{W}}_{c,\ell}\, z_{c,t} = \beta^{\mathrm{W}}_{\ell,t} \qquad \ell \in L, t \in T \tag{4}$$

$$\sum_{c \in C} p^{\mathrm{W}}_{c,\ell'}\, p^{\mathrm{B}}_{c,\ell}\, z_{c,t} \leq \beta^{\mathrm{B}}_{\ell,t} \qquad \ell' \in L, \ell \in L \setminus \{\ell'\}, t \in T \tag{5}$$

$$\sum_{c \in C} a^{\mathrm{W}}_{c,v'}\, p^{\mathrm{B}}_{c,\ell}\, z_{c,t} \leq \beta^{\mathrm{B}}_{\ell,t} \qquad v' \in V_{\mathrm{D}}, \ell \in L, t \in T \tag{6}$$

$$\sum_{c \in C} \delta\, p^{\mathrm{S}}_{c,\ell}\, z_{c,t} = \beta^{\mathrm{S}}_{\ell,t} \qquad \ell \in L, t \in T \tag{7}$$

$$z_{c,t} \in \mathbb{R}_{\geq 0} \qquad c \in C, t \in T \tag{8}$$

$$\mathrm{BW}_{\ell} \in \mathbb{R}_{\geq 0} \qquad \ell \in L \tag{9}$$

$$\beta^{\mathrm{W}}_{\ell,t},\, \beta^{\mathrm{B}}_{\ell,t},\, \beta^{\mathrm{S}}_{\ell,t} \in \mathbb{R}_{\geq 0} \qquad \ell \in L, t \in T \tag{10}$$

The objective (1) simply is the bandwidth cost. Here, we sum all the link bandwidths BW_{ℓ}, which is enforced to be the maximum bandwidth carried by link ℓ over all time periods $t \in T$ through constraint (2).

The main constraint (3) simply assures that we fulfill the demand. The subsequent constraints (4)–(7) count the bandwidths on link ℓ during time period t for the working, backup and synchronization paths. The backup path is designed such that we can survive failures of either any single link ℓ' (through (5)) or any single data center v' (via (6)).

The remaining constraints (8)–(10) simply express the domains of all variables as non-negative real numbers.

2) Scenario I: In the baseline Scenario I we need to add extra constraints to enforce that traffic that lasts from one period to the next does not change in terms of the working, backup and synchronization paths. This is achieved by the following equations, for all $v \in V_{\mathrm{S}}, \pi \in \Pi_v, t \in T'$:

$$\sum_{c \in C^{\mathrm{W}}_{\pi}} (z_{c,t} - z_{c,t-1}) \begin{cases} \geq 0 & \text{if } \Delta_{v,t} \geq \Delta_{v,t-1} \\ < 0 & \text{else (i.e., } \Delta_{v,t} < \Delta_{v,t-1}) \end{cases} \tag{11}$$

$$\sum_{c \in C^{\mathrm{B}}_{\pi}} (z_{c,t} - z_{c,t-1}) \begin{cases} \geq 0 & \text{if } \Delta_{v,t} \geq \Delta_{v,t-1} \\ < 0 & \text{else (i.e., } \Delta_{v,t} < \Delta_{v,t-1}) \end{cases} \tag{12}$$

$$\sum_{c \in C^{\mathrm{S}}_{\pi}} (z_{c,t} - z_{c,t-1}) \begin{cases} \geq 0 & \text{if } \Delta_{v,t} \geq \Delta_{v,t-1} \\ < 0 & \text{else (i.e., } \Delta_{v,t} < \Delta_{v,t-1}) \end{cases} \tag{13}$$

As an example, the first case in (11) enforces that the amount of traffic from source node v carried over path π does not decrease from period $t-1$ to period t if the total traffic increases (i.e., if $\Delta_{v,t} \geq \Delta_{v,t-1}$): if we have a volume x on that path for source v in period $t-1$, at least the same volume will still cross it during t.

3) Scenario II: For the case where we only want to keep the same working paths (but allow changing backup and/or synchronization paths), clearly we will only need to add constraint (11) to the baseline model (1)–(10).

Note that now we have in essence three different RMP formulations for each of the Scenarios I–III. To unify these into a single one, and allow to have a single PP formulation for all of them, we introduce auxiliary variables $\gamma^{\bullet}_{\pi,t} = \sum_{c \in C^{\bullet}_{\pi}} z_{c,t}$, and replace (11)–(13) by the following:

$$\sum_{c \in C^{\mathrm{W}}_{\pi}} z_{c,t} = \gamma^{\mathrm{W}}_{\pi,t} \qquad \pi \in \Pi_v, t \in T \tag{14}$$

$$\sum_{c \in C^{\mathrm{B}}_{\pi}} z_{c,t} = \gamma^{\mathrm{B}}_{\pi,t} \qquad \pi \in \Pi_v, t \in T \tag{15}$$

$$\sum_{c \in C^{\mathrm{S}}_{\pi}} z_{c,t} = \gamma^{\mathrm{S}}_{\pi,t} \qquad \pi \in \Pi_v, t \in T \tag{16}$$

Further, for all $v \in V_{\mathrm{S}}, \pi \in \Pi_v, t \in T'$:

$$\gamma^{\mathrm{W}}_{\pi,t} - \gamma^{\mathrm{W}}_{\pi,t-1} \begin{cases} \geq 0 & \text{if } \Delta_{v,t} \geq \Delta_{v,t-1} \\ < 0 & \text{else (i.e., } \Delta_{v,t} < \Delta_{v,t-1}) \end{cases} \tag{11'}$$

$$\gamma^{\mathrm{B}}_{\pi,t} - \gamma^{\mathrm{B}}_{\pi,t-1} \begin{cases} \geq 0 & \text{if } \Delta_{v,t} \geq \Delta_{v,t-1} \\ < 0 & \text{else (i.e., } \Delta_{v,t} < \Delta_{v,t-1}) \end{cases} \tag{12'}$$

$$\gamma^{\mathrm{S}}_{\pi,t} - \gamma^{\mathrm{S}}_{\pi,t-1} \begin{cases} \geq 0 & \text{if } \Delta_{v,t} \geq \Delta_{v,t-1} \\ < 0 & \text{else (i.e., } \Delta_{v,t} < \Delta_{v,t-1}) \end{cases} \tag{13'}$$

Now, we include (14)–(16) for all scenarios, and only add (11') for Scenario II, and (11')–(13') for Scenario I.

B. Pricing Problem (PP)

As highlighted above, the purpose of the pricing problem is to construct new configurations for a given source node, that will help to lower the cost of the overall routing, i.e., the objective function value of the RMP (extended with the newly found configuration from the PP).

The objective for the pricing follows directly from the RMP as the minimization of reduced cost (see, e.g., [13] for the general principle of column generation) and here amounts to:

$$\begin{aligned}\min \quad \overline{\mathrm{COST}}(z_{v,t}) &= 0 - u_{v,t}^{(3)} - \sum_{\ell \in L} p_\ell^{\mathrm{W}}\, u_{\ell,t}^{(4)} \\ &+ \sum_{\ell' \in L} \sum_{\ell \in L \setminus \{\ell'\}} p_{\ell'}^{\mathrm{W}}\, p_\ell^{\mathrm{B}}\, u_{\ell',\ell,t}^{(5)} + \sum_{v' \in V_{\mathrm{D}}} \sum_{\ell \in L} a_{v'}^{\mathrm{W}}\, p_\ell^{\mathrm{B}}\, u_{v',\ell,t}^{(6)} \\ &- \sum_{\ell \in L} \delta\, p_\ell^{\mathrm{S}}\, u_{\ell,t}^{(7)} + \sum_{\pi \in \Pi_v} u_{\pi,t}^{(14)} + \sum_{\pi \in \Pi_v} u_{\pi,t}^{(15)} + \sum_{\pi \in \Pi_v} u_{\pi,t}^{(16)} \qquad (17)\end{aligned}$$

with

$$u_{v,t}^{(3)} \geq 0, \quad u_{\ell,t}^{(4)} \lesseqgtr 0, \quad u_{\ell',\ell,t}^{(5)} \geq 0, \quad u_{v',\ell,t}^{(6)} \geq 0, \quad u_{\ell,t}^{(7)} \lesseqgtr 0,$$
$$u_{\pi,t}^{(14)} \lesseqgtr 0, \quad u_{\pi,t}^{(15)} \lesseqgtr 0, \quad u_{\pi,t}^{(16)} \lesseqgtr 0$$

Here, the $u_{\cdot}^{\cdot}$ are **parameters** for the PP, which are the values of the dual variables associated with the constraints from the RMP. The **decision variables** of the PP are the $p_{\cdot}^{\cdot}$ and $a_{\cdot}^{\cdot}$ variables, with the same meaning as before, but where we dropped the c index, since we are now constructing a new configuration (associated with source node $v \in V_{\mathrm{S}}$ and time slot $t \in T$). In addition, we define the following auxiliary decision variables to keep track of the flow constraints:

$d_{v'}^{\mathrm{W}}, d_{v'}^{\mathrm{B}}, d_{v'}^{\mathrm{S}}$ are defined for all$v' \in V$, and are binary variables that equal 1 if node v' is on respectively the working (W), backup (B) or synchronization (S) path, and else equal 0.

Note that the objective function contains quadratic terms, but these can be easily linearized through the introduction of auxiliary variables. For example, we can define $p_{\ell',\ell}^{\mathrm{WB}} \triangleq p_{\ell'}^{\mathrm{W}}\, p_\ell^{\mathrm{B}}$, and then enforce this equality through linear constraints:

$$\begin{cases} p_{\ell',\ell}^{\mathrm{WB}} \leq p_{\ell'}^{\mathrm{W}} \\ p_{\ell',\ell}^{\mathrm{WB}} \leq p_\ell^{\mathrm{B}} \\ p_{\ell',\ell}^{\mathrm{WB}} \geq p_{\ell'}^{\mathrm{W}} + p_\ell^{\mathrm{B}} \end{cases} \qquad (18)$$

On top of these auxiliary constraints to linearize the problem, the following constraints complete the PP formulation (where v is the current source node we are constructing a configuration for):[4]

$$\sum_{\ell \in \omega(v)} p_\ell^{\mathrm{W}} = \begin{cases} 1 & v' = v \\ 2\, d_{v'}^{\mathrm{W}} - a_{v'}^{\mathrm{W}} & v' \in V_{\mathrm{D}} \setminus \{v\} \\ 2\, d_{v'}^{\mathrm{W}} & \text{else} \end{cases} \qquad (19)$$

$$\sum_{\ell \in \omega(v)} p_\ell^{\mathrm{B}} = \begin{cases} 1 & v' = v \\ 2\, d_{v'}^{\mathrm{B}} - a_{v'}^{\mathrm{B}} & v' \in V_{\mathrm{D}} \setminus \{v\} \\ 2\, d_{v'}^{\mathrm{B}} & \text{else} \end{cases} \qquad (20)$$

$$\sum_{\ell \in \omega(v)} p_\ell^{\mathrm{S}} = \begin{cases} 1 & v' = v \\ 2\, d_{v'}^{\mathrm{S}} - a_{v'}^{\mathrm{S}} & v' \in V_{\mathrm{D}} \setminus \{v\} \\ 2\, d_{v'}^{\mathrm{S}} & \text{else} \end{cases} \qquad (21)$$

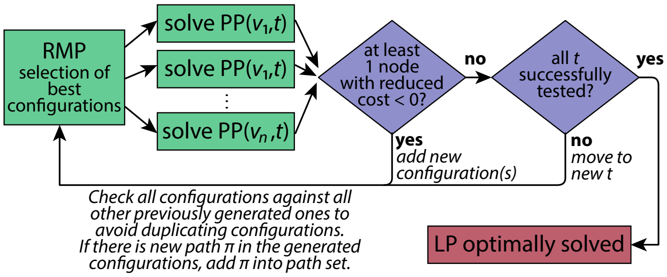

RMP
selection of best configurations
solve PP(v_1,t)
solve PP(v_1,t)
⋮
solve PP(v_n,t)
at least 1 node with reduced cost < 0?
no
all t successfully tested?
yes
yes
add new configuration(s)
no
move to new t
Check all configurations against all other previously generated ones to avoid duplicating configurations.
If there is new path π in the generated configurations, add π into path set.
LP optimally solved

Fig. 2: The parallel column generation solution scheme.

$$p_\ell^{\mathrm{W}} + p_\ell^{\mathrm{B}} \leq 1 \qquad \ell \in L \qquad (22)$$
$$\sum_{v' \in V_{\mathrm{D}}} a_{v'}^{\mathrm{W}} = 1 \qquad (23)$$
$$\sum_{v' \in V_{\mathrm{D}}} a_{v'}^{\mathrm{B}} = 1 \qquad (24)$$
$$a_{v'}^{\mathrm{W}} + a_{v'}^{\mathrm{B}} \leq 1 \qquad v' \in V_{\mathrm{D}} \qquad (25)$$
$$a_{v'}^{\mathrm{W}},\, a_{v'}^{\mathrm{B}} \in \{0,1\} \qquad v' \in V_{\mathrm{D}} \qquad (26)$$
$$d_{v'}^{\mathrm{W}},\, d_{v'}^{\mathrm{B}},\, d_{v'}^{\mathrm{S}} \in \{0,1\} \qquad v' \in V \qquad (27)$$
$$p_\ell^{\mathrm{W}},\, p_\ell^{\mathrm{B}} \in \{0,1\} \qquad \ell \in L \qquad (28)$$

Constraints (19)–(21) are the traditional flow conservation constraints. Further, we ensure path disjointness among working and backup paths through (22). We pick exactly 1 working and backup data center via respectively (23) and (24), which we enforce to be disjoint via (25). The final (26)–(28) are simply the domains of the binary decision variables.

C. Solution strategy

The general approach to solving a column generation problem is to re-solve the RMP each time we add a new configuration to the candidate configuration set C as found by the PP. In our current model, a PP is associated with a given source node v and time period t. This implies we can devise several strategies to choose to solve these different PPs. The straightforward, *serial scheme* is to add one configuration at a time for a selected source node, e.g., in round robin fashion, and resolve the RMP after adding each such newly found configuration. As an alternative, we will explore a *parallel scheme*, as sketched in 2, that solves PPs for all source nodes V_{S} in parallel and resolves the RMP after adding multiple configurations (at most one per source node). Note that solving all PP instances in parallel means that we will simultaneously use $|V_{\mathrm{S}}|$ processor cores.[5]

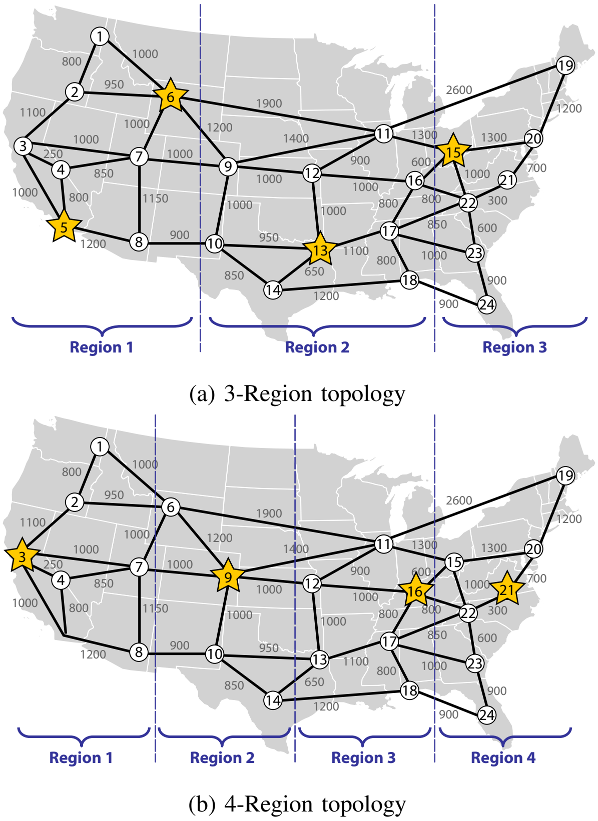
IV. Experimental case study results

A. Experiment setup

Our case study considers a 24-node US topology comprising 43 undirected links, as depicted in Fig. 3. Since we adopt anycast routing, traffic is specified in terms of its source node only. We vary the traffic in terms of time-of-day (i.e., period) as well as per region (which each is assumed to have its own

[4] Here, $\omega(v)$ denotes the set of incident links for node v.

[5] Clearly, if we use only a fraction (e.g., say $|V_{\mathrm{S}}|/k$), the speedup will decrease accordingly (in our example with a factor of about $1/k$).



(a) 3-Region topology

(b) 4-Region topology

Fig. 3: US network topology with the assumed regions and data center locations indicated with a star.

artificial time zone). The split of the total traffic volume across the various regions is as follows:

(i) for the *3-region case*, 33.3% originates from Region 1, 37.5% from Region 2, and the remaining 29.2% from Region 3,

(ii) for the *4-region case*, 29.2% originates from Region 1, 16.6% from Region 2, 25% from Region 3, and 29.2% from Region 4.

For a given Region, the traffic varies during the day, with 48% of the Region's traffic in 8 am–4 pm, 38% in 4 pm–12 am, and the remaining 14% from 12 am–8 am.

The volume of traffic, generated in one of the 3 time periods and a given Region, is further divided in a portion that just lasts that single period and the other portion that will continue into the next period. We consider three patterns, with respectively 20%, 50% or 80% of two-period traffic.

B. *Resource savings*

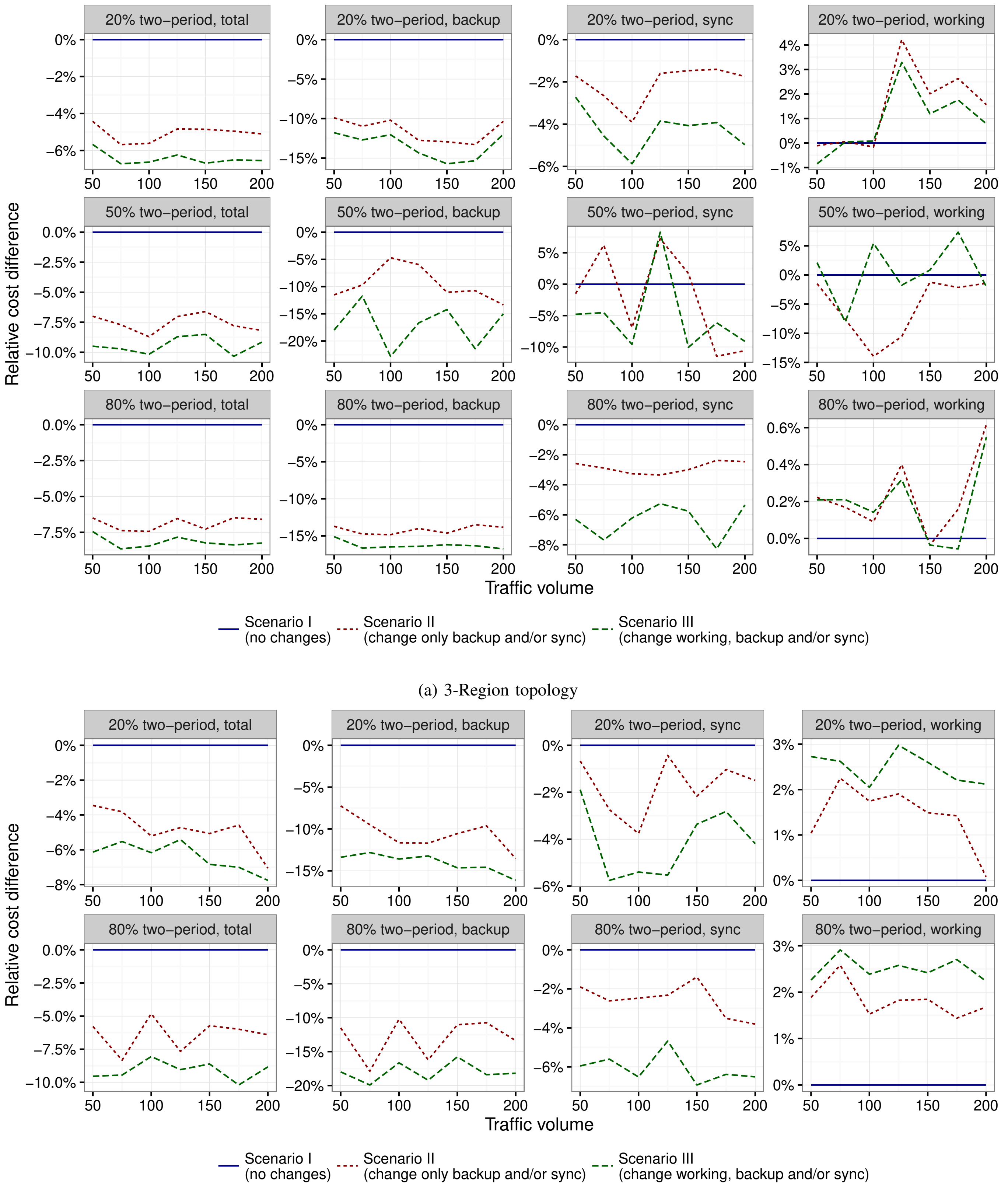
The major question we set at the outset of this study was: What savings van we attain in terms of bandwidth requirements by choosing to reroute multi-period traffic from one period to the next? Indeed, we expect to possibly achieve overall bandwidth savings by being more flexible, i.e., when going from Scenario I (that does not allow any rerouting), over Scenario II (where we can change backup and/or synchronization paths) to Scenario III (that is fully flexible and also admits changes in working paths). As to the impact of the volume of traffic that spans multiple period, intuition suggests that the relative savings could be more substantial when we have more traffic that is available for rerouting, i.e., when the fraction of multi-period (compared to single period traffic) increases. On the other hand, if the traffic in one period vs. the next does not change much, e.g., the volume of (different) single period traffic is negligible, then the incentive to change routing much will disappear.

Figure 4 shows the relative difference in bandwidth in detail for each of the considered traffic and topology scenarios. We draw the following quantitative observations: compared to the baseline Scenario I, the total bandwidth cost is reduced with on average 5.1% (resp. 6.4%) for Scenario II (resp. Scenario III) with traffic Pattern #1, and by 6.9% (resp. 8.2%) with Pattern #2 (where the average is taken over all traffic volumes). This net saving mainly stems from a reduction of bandwidth for the backup paths, due to increased sharing: we note an average reduction of the backup bandwidth cost of 11.5% (resp. 13.4%) for the case of 20% two-period traffic and 14.2% (resp. 16.3%) for the case of 80% two-period traffic, when only changing backup/sync paths, i.e., Scenario II (resp. Scenario III, where also the working route can change). Thus, this case study suggests that the maximal possible cost reduction (in terms of bandwidth requirements) achievable by full rerouting flexibility (Scenario III) can be largely achieved even if we only change the backup/synchronization paths (Scenario II): the additional advantage of allowing also the working path to be changed (i.e., the extra benefit of Scenario III compared to Scenario II) is much smaller than the cost reduction achieved by moving from a fixed routing (Scenario I vs. Scenario II). The net savings for this first 3-Region case study are modest, but non-negligible. When we consider a slightly more extreme 4-Region case (see Fig. 4b) the savings are a higher.

Studying the impact of the volume of multi-period traffic vs. single-period traffic, we note that our results confirm the aforementioned intuition: in Fig. 4a, maximal savings are obtained for the 50% two-period traffic scenario. If the portion of two-period traffic increases further (e.g., the 80% two-period traffic case), savings go down. We believe that the observed behavior is due to the fact that savings are realized by wisely choosing backup paths to increase sharing: the amount of traffic that multi-period traffic can freely share backup paths with (i.e., the next period's newly generated one-period traffic) goes down when going from 50% to 80% of two-period traffic, and so do the savings. Indeed, when looking at the resource savings split into working, backup and synchronization path capacities, we note that by introducing rerouting opportunities, it is the backup capacity that substantially goes down (with savings up to around 20%). To enable such increased sharing, the working paths tend to get slightly longer, as can be inferred from the (small) cost increase for working path capacity (see the rightmost graphs in Fig. 4).

C. *Benefit of parallel PP solving*

An advantage of splitting the overall problem in a column generation decomposition, with pricing problems (PPs) per



(a) 3-Region topology

(b) 4-Region topology

Fig. 4: The bandwidth requirements for time-varying traffic, for different traffic patterns and topologies. Traffic volume is expressed in number of unit requests.

source node, is that we can solve multiple PPs in parallel. Thus, the master problem (RMP) is (re-)solved after adding potentially in the order of $|V_{\mathrm{S}}|$ (i.e., the number of source nodes) new configurations. In our case study we have $|V_{\mathrm{S}}| = 20$ potential source nodes. We quantitatively study the achieved gains in terms of wall clock time[6] over the multiple consecutive rounds of the two solution strategies, namely the naive sequential one and the parallel strategy of Fig. 2 as discussed in Section III-C. We define a "round" as the whole set of RMPs/PPs that attempt to find a new configuration for each source node (by solving the corresponding pricing problems, PPs).

In the *Parallel* scheme, we re-solve the RMP only after adding all new configurations found by the PPs (that are executed in parallel): a single round comprises 1 restricted master problem (RMP) and for each source node one PP (so, $|V_{\mathrm{S}}|$ in total). In the *Serial* case, we solve one PP at a time, and re-solve the RMP each time we found a new configuration: one round thus comprises multiple RMPs (1 for each source node where the PP found a new configuration). Since solving RMPs dominates the running time, and we have in the order of 20 source nodes in our topology, we find that the time per individual round lies close to a factor 20 higher for the *Serial* strategy compared to *Parallel*. This is illustrated in Fig. 5, where we quantitatively plot the wall clock times as measured in case of the 4-region topology, 80% two-period traffic, for Scenario I. (We find qualitative results for other cases.)

We note that the graphs in Fig. 5 stop earlier, i.e., for fewer rounds, for the *Serial* case compared to *Parallel*. Indeed, the final solution is reached after fewer rounds. The reason is that when we solve the PPs in the *Parallel* scheme, they all use the dual values of the RMP solved previously. In the *Serial* scheme, after adding only a single configuration, the duals already change and thus affect the new configuration construction when solving the PP for the next source node. This implies that the *Serial* case will find the optimal set of configurations after fewer rounds. Also, the *Serial* scheme avoids generating unnecessary configuration that way, as can be observed in the rightmost graph of Fig. 5. Still, even though the *Parallel* scheme introduces unnecessary configurations, the savings in terms of overall wall clock time are substantial.

V. Conclusion

We have defined a new column generation model to solve a multi-period traffic dimensioning problem for resilient backbone networks for multi-site data centers. We applied it in an experiment on a 24-node US backbone network with 3 cyclic time periods, time-shifted across 3 or 4 regions with their own distinct time zones. Through these case studies, we quantitatively studied the potential bandwidth savings achievable by rerouting demands that span multiple time periods. We can summarize the main observations from our experiments as follows:

- The bandwidth savings mainly stem from *backup* paths (because of increased sharing with the requests starting in the 2nd period of two-period requests).
- A small part of those backup capacity savings are negated by longer *working* paths, chosen to avoid overlap among concurrent demands, and thus allow more sharing of backup capacity.
- When we allow to reroute working paths (Scenario III), the reduction in bandwidth requirements is slightly higher than when only rerouting backup/synchronization paths (Scenario II), but the difference seems not substantial.
- The overall bandwidth savings from rerouting multi-period requests from one period to the next, do not seem to exceed 10%.

Furthermore, we also demonstrated that by adopting a parallel solution strategy, we can achieve a substantial reduction of the (wall clock) time required to solve the complete anycast routing problem. That overall saving is achieved by solving multiple PPs in parallel (one for each source node), and only re-solving the RMP with the newly found configurations (at most one per source node) after they all completed. Compared to a naive serial approach that rather solves one PP at a time, and re-solves the RMP each time a new configuration from such PP is added to the RMP, we can reduce the time for solving RMPs with a factor of about $1/|V_{\mathrm{S}}|$ per round (where a "round" comprises solving one PP for each source node), with V_{S} the set of source nodes (and assuming we can use at least $|V_{\mathrm{S}}|$ processor cores). Even though in the parallel case we have to iterate over slightly more such rounds — indeed, adding a new configuration for a given source node v impacts what other configuration for a different source node v' might share bandwidth with the new v configuration — the net saving in total time still is substantial compared to a naive serial execution plan.

[6] This is the actual time passed between starting the solution process and the (intermediate) solution of the column generation problem. Thus, in case of parallel solving of multiple PPs, it amounts to the maximum time of the slowest PP. The CPU time would be the sum of the times required to solve each of the individual PPs.

References

[1] T. Wang, B. Jaumard, and C. Develder, "Anycast (re)routing of multi-period traffic in dimensioning resilient backbone networks for multi-site data centers," in *Proc. 18th Int. Conf. Transparent Optical Netw. (ICTON 2016)*, Trento, Italy, 10–14 Jul. 2016, pp. 1–6.

[2] ——, "Resilient backbone networks for multi-site data centers: Exploiting anycast (re)routing for multi-period traffic," in *Proc. OSA Photonic Netw. Devices*, Vancouver, BC, Canada, 18–20 Jul. 2016, pp. 1–3.

[3] C. Develder, M. De Leenheer, B. Dhoedt, M. Pickavet, D. Colle, F. De Turck, and P. Demeester, "Optical networks for grid and cloud computing applications," *Proc. IEEE*, vol. 100, no. 5, pp. 1149–1167, May 2012.

[4] C. Develder, J. Buysse, B. Dhoedt, and B. Jaumard, "Joint dimensioning of server and network infrastructure for resilient optical grids/clouds," *IEEE/ACM Trans. Netw.*, vol. 22, no. 5, pp. 1591–1606, Oct. 2014.

[5] T. Wang, B. Jaumard, and C. Develder, "A scalable model for multi-period virtual network mapping for resilient multi-site data centers," in *Proc. 17th Int. Conf. Transparent Optical Netw. (ICTON 2015)*, Budapest, Hungary, 5–9 Jul. 2015, pp. 1–8.

[6] T. Stevens, M. De Leenheer, C. Develder, F. De Turck, B. Dhoedt, and P. Demeester, "ASTAS: Architecture for scalable and transparent anycast services," *J. Commun. Netw.*, vol. 9, no. 4, pp. 1229–2370, 2007.

[7] B. G. Bathula and J. M. Elmirghani, "Constraint-based anycasting over optical burst switched networks," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 1, no. 2, pp. A35–A43, Jul. 2009.

[8] X. He and G.-S. Poo, "Sub-reconfiguration of backup paths based on shared path protection for WDM networks with dynamic traffic pattern," in *Proc. 9th Int. Conf. Commun. Systems (ICCS 2004)*, Singapore, China, 7 Sep. 2004, pp. 391–395.

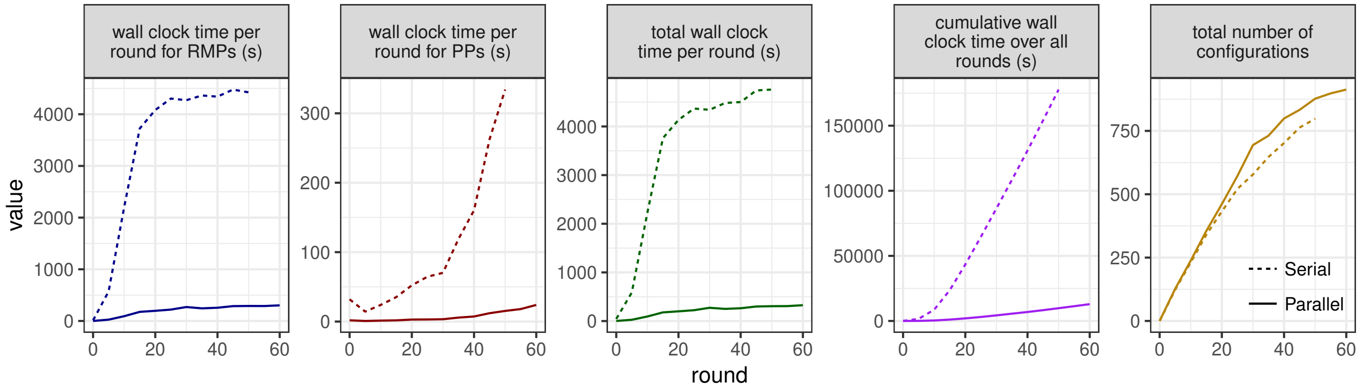


Fig. 5: The running times and number of configurations in case of the naive *serial* vs. the *parallel* solution scheme, for the 4-region case, with 80% two-period traffic and a traffic volume of 50 unit demands, when considering the no-rerouting Scenario I.

[9] S. Srivastava, S. R. Thirumalasetty, and D. Medhi, “Network traffic engineering with varied levels of protection in the next generation internet,” in *Performance evaluations and planning methods for the next generation internet*, A. Girard, B. Sansò, and F. Vazquez-Abad, Eds. Springer Verlag, 2005, pp. 99–124.

[10] S. Sebbah and B. Jaumard, “Differentiated quality-of-protection in survivable WDM mesh networks using p-structures,” *Computer Commun.*, vol. 36, pp. 621–629, Mar. 2013.

[11] M. Bui, T. Wang, B. Jaumard, D. Medhi, and C. Develder, “Time-varying resilient virtual network mapping for multi-location cloud data centers,” in *Proc. 16th Int. Conf. Transparent Optical Netw. (ICTON 2014)*, Graz, Austria, 6–10 Jul. 2014, pp. 1–8.

[12] Çicek Çavdar, A. Yayımlı, and L. Wosinska, “How to cut the electric bill in optical WDM networks with time-zones and time-of-use prices,” in *Proc. 37th Eur. Conf. Optical Commun. (ECOC 2011)*, Geneva, Switzerland, 19–21 Sep. 2011, pp. 1–3.

[13] V. Chvatal, *Linear Programming*. Freeman, 1983.