Universitat Politècnica de Catalunya

Optical Communications Group

Orchestrating Datacenters and Networks to Facilitate the Telecom Cloud

Adrián Asensio García

A thesis presented in partial fulfillment of the requirements for the degree of

Philosophy Doctor

Advisor:

Dr. Luis Velasco Esteban

June 2016

ISBN 978-84-608-7287-0

Optical Communications Group (GCO)

Universitat Politècnica de Catalunya (UPC)

C/ Jordi Girona, 1-3

Campus Nord, D4-213

08034 Barcelona, Spain

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola de Doctorat

Assessment results for the doctoral thesis

Academic year:

Full name

Doctoral programme

Structural unit in charge of the programme

Decision of the committee

In a meeting with the examination committee convened for this purpose, the doctoral candidate presented the topic of his/her doctoral thesis entitled

_____

_____.

Once the candidate had defended the thesis and answered the questions put to him/her, the examiners decided to award a mark of:

☐ UNSATISFACTORY ☐ SATISFACTORY ☐ GOOD ☐ VERY GOOD

(Full name and signature) Chairperson	(Full name and signature) Secretary

(Full name and signature) Member	(Full name and signature) Member	(Full name and signature) Member

_____, _____

The votes of the members of the examination committee were counted by the Doctoral School at the behest of the Doctoral Studies Committee of the UPC, and the result is to award the CUM LAUDE DISTINCTION:

☐ YES ☐ NO

(Full name and signature) Chair of the Standing Committee of the Doctoral School	(Full name and signature) Secretary of the Standing Committee of the Doctoral School

Barcelona, _____

International doctoral degree statement

As the secretary of the examination committee, I hereby state that the thesis was partly (at least the abstract and the conclusions) defended in a language commonly used in scientific communication in the field that is not an official language of Spain. This does not apply if the stay, report or expert is from a Spanish-speaking country.

(Nom, cognoms i signatura) Secretary of the Examination Commitee

Agradecimientos

Esta tesis no habría sido posible sin la confianza y el apoyo de todas aquellas personas que han estado junto a mí a lo largo de estos últimos años.

Quiero agradecer especialmente a Luis Velasco, director de esta tesis, su impecable dirección y profesionalidad en todo momento. Gracias a su dedicación, apoyo y guía ha hecho de esta tesis un trabajo apasionante para mí, así como una de las experiencias más gratificantes tanto a nivel académico como personal.

También quiero agradecer la confianza y ayuda que me han brindado Gabriel Junyent, Jaume Comellas y Marc Ruiz, así como el resto de compañeros del GCO de la UPC, quienes me han acogido desde el primer momento.

Finalmente, quiero agradecer a mis padres, Dani y Montse, y mi hermano, Martí, su generosidad y apoyo, así como dedicar esta tesis a Leila y Eric.

Abstract

In the Internet of services, information technology (IT) infrastructure providers play a critical role in making the services accessible to end-users. IT infrastructure providers host platforms and services in their datacenters (DCs). The cloud initiative has been accompanied by the introduction of new computing paradigms, such as Infrastructure as a Service (IaaS) and Software as a Service (SaaS), which have dramatically reduced the time and costs required to develop and deploy a service.

However, transport networks become crucial to make services accessible to the user and to operate DCs. Transport networks are currently configured with *big static fat pipes* based on capacity over-provisioning aiming at guaranteeing traffic demand and other parameters committed in Service Level Agreement (SLA) contracts. Notwithstanding, such over-dimensioning adds high operational costs for DC operators and service providers. Therefore, new mechanisms to provide reconfiguration and adaptability of the transport network to reduce the amount of over-provisioned bandwidth are required. Although *cloud-ready* transport network architecture was introduced to handle the dynamic cloud and network interaction and Elastic Optical Networks (EONs) can facilitate elastic network operations, orchestration between the cloud and the interconnection network is eventually required to coordinate resources in both strata in a coherent manner.

In addition, the explosion of Internet Protocol (IP)-based services requiring not only dynamic cloud and network interaction, but also additional service-specific SLA parameters and the expected benefits of Network Functions Virtualization (NFV), open the opportunity to telecom operators to exploit that cloud-ready transport network and their current infrastructure, to efficiently satisfy network requirements from the services. In the telecom cloud, a *pay-per-use* model can be offered to support services requiring resources from the transport network and its infrastructure.

In this thesis, we study connectivity requirements from representative cloud-based services and explore connectivity models, architectures and orchestration schemes to satisfy them aiming at facilitating the telecom cloud.

The main objective of this thesis is demonstrating, by means of analytical models and simulation, the viability of orchestrating DCs and networks to facilitate the telecom cloud.

To achieve the main goal we first study the connectivity requirements for DC interconnection and services on a number of scenarios that require connectivity from the transport network. Specifically, we focus on studying DC federations, live-TV distribution, and 5G mobile networks. Next, we study different connectivity schemes, algorithms, and architectures aiming at satisfying those connectivity requirements. In particular, we study polling-based models for dynamic inter-DC connectivity and propose a novel notification-based connectivity scheme where inter-DC connectivity can be delegated to the network operator. Additionally, we explore virtual network topology provisioning models to support services that require service-specific SLA parameters on the telecom cloud. Finally, we focus on studying DC and network orchestration to fulfill simultaneously SLA contracts for a set of customers requiring connectivity from the transport network.

It shall be mentioned that part of the work reported in this thesis has been done within the framework of European and National projects, namely IDEALIST (FP7-ICT-2011-8), funded by the European Commission, and ELASTIC (TEC2011-27310) and SYNERGY (TEC2014-59995-R) funded by the Spanish Science Ministry.

Resumen

En la Internet de los servicios, los proveedores de recursos relacionados con tecnologías de la información juegan un papel crítico haciéndolos accesibles a los usuarios como servicios. Dichos proveedores, hospedan plataformas y servicios en centros de datos. La oferta plataformas y servicios en la nube ha introducido nuevos paradigmas de computación tales como ofrecer la infraestructura como servicio, conocido como IaaS de sus siglas en inglés, y el *software* como servicio, SaaS. La disponibilidad de recursos en la nube, ha contribuido a la reducción de tiempos y costes para desarrollar y desplegar un servicio.

Sin embargo, para permitir el acceso de los usuarios a los servicios así como para operar los centros de datos, las redes de transporte resultan imprescindibles. Actualmente, las redes de transporte están configuradas con conexiones estáticas y su capacidad sobredimensionada para garantizar la demanda de tráfico así como los distintos parámetros relacionados con el nivel de servicio acordado. No obstante, debido a que el exceso de capacidad en las conexiones se traduce en un elevado coste tanto para los operadores de los centros de datos como para los proveedores de servicios, son necesarios nuevos mecanismos que permitan adaptar y reconfigurar la red de forma eficiente de acuerdo a las nuevas necesidades de los servicios a los que dan soporte. A pesar de la introducción de arquitecturas que permiten la gestión de redes de transporte y su interacción con los servicios en la nube de forma dinámica, y de la irrupción de las redes ópticas elásticas, la orquestación entre la nube y la red es necesaria para coordinar de forma coherente los recursos en los distintos estratos.

Además, la explosión de servicios basados el Protocolo de Internet, IP, que requieren tanto interacción dinámica con la red como parámetros particulares en los niveles de servicio además de los habituales, así como los beneficios que se esperan de la virtualización de funciones de red, representan una oportunidad para los operadores de red para explotar sus recursos y su infraestructura. La nube de operador permite ofrecer recursos del operador de red a los servicios, de forma similar a un sistema basado en pago por uso.

En esta Tesis, se estudian requisitos de conectividad de servicios basados en la nube y se exploran modelos de conectividad, arquitecturas y modelos de orquestación que contribuyan a la realización de la nube de operador.

El objetivo principal de esta Tesis es demostrar la viabilidad de la orquestación de centros de datos y redes para facilitar la nube de operador, mediante modelos analíticos y simulaciones.

Con el fin de cumplir dicho objetivo, primero estudiamos los requisitos de conectividad para la interconexión de centros de datos y servicios en distintos escenarios que requieren conectividad en la red de transporte. En particular, nos centramos en el estudio de escenarios basados en federaciones de centros de datos, distribución de televisión en directo y la evolución de las redes móviles hacia 5G. A continuación, estudiamos distintos modelos de conectividad, algoritmos y arquitecturas para satisfacer los requisitos de conectividad. Estudiamos modelos de conectividad basados en sondeos para la interconexión de centros de datos y proponemos un modelo basado en notificaciones donde la gestión de la conectividad entre centros de datos se delega al operador de red. Estudiamos la provisión de redes virtuales para soportar en la nube de operador servicios que requieren parámetros específicos en los acuerdos de nivel de servicio además de los habituales. Finalmente, nos centramos en el estudio de la orquestación de centros de datos y redes con el objetivo de satisfacer de forma simultánea requisitos para distintos servicios.

Parte del trabajo realizado durante el desarrollo de esta Tesis ha sido financiado por el proyecto Europeo IDEALIST (FP7-ICT-2011-8), y los proyectos nacionales ELASTIC (TEC2011-27310) y SYNERGY (TEC2014-59995-R).

Table of Contents

Page

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

In the Internet of services, information technology (IT) infrastructure providers play a critical role in making the services accessible to end-users. IT infrastructure providers host platforms and services in their datacenters (DCs). The cloud initiative has been accompanied by the introduction of new computing paradigms, such as Infrastructure as a Service (IaaS) and Software as a Service (SaaS), which have dramatically reduced the time and costs required to develop and deploy a service [Ar10]. These paradigms are playing a role of paramount importance in the way companies invest their money regarding IT resources: they are moving from a model where large amounts of capital expenditures (CAPEX) are needed to build their own IT infrastructure and additional cost to operate and maintain it (operational expenditures, OPEX), to a pure OPEX model where IT resources are requested to cloud providers in a *pay-as-you-go* model.

However, that new model and the requirements derived from cloud services are contributing to the increasing amount of data that require connectivity from the transport networks. Cisco global cloud index [GCI12] forecasts DC traffic to reach 554 Exabytes (EB) per month by 2016. Two main components of traffic leaving from/arriving to DCs can be distinguished: traffic among DCs (DC2DC) and traffic between DCs and end-users (referred to as U2DC in this thesis).

From the network point of view, transport networks are currently configured with *big static fat pipes* based on capacity over-provisioning aiming at guaranteeing traffic demand and other parameters committed in Service Level Agreement (SLA) contracts. The capacity of optical connections for inter-DC connectivity is dimensioned in advance based on some volume of foreseen data to transfer.

From the cloud perspective, scheduling algorithms inside cloud management run periodically trying to optimize some cost function, such as energy costs, and organize data transferences as a function of the available bitrate. Obviously, this static connectivity configuration adds high costs since large connectivity capacity remains unused during periods where small amount of data needs to be transferred. Thus, cloud connectivity services require new mechanisms to provide reconfiguration and adaptability of the transport network to reduce the amount of over-provisioned bandwidth. Although cloud-ready transport network architecture [Co12] was introduced to handle this dynamic cloud and network interaction allowing on-demand connectivity provisioning, the efficient integration of cloud-based services among distributed DCs, including the interconnecting network, becomes a challenge.

Moreover, the explosion of Internet Protocol (IP)-based services requiring not only dynamic connectivity, but also service-specific SLA parameters and the expected benefits of Network Functions Virtualization (NFV), open the opportunity to telecom operators to exploit that cloud-ready transport network and their current infrastructure to efficiently satisfy network connectivity requirements from the services. This results in the telecom cloud, where a *pay-per-use* model can be offered to support applications and services requiring connectivity from the transport network.

Evolution towards cloud-ready transport networks, entails dynamically controlling network resources, considering cloud requests in the network configuration process. Hence, that evolution is based on elastic data and control planes, which can interact with multiple network technologies and cloud services. Although Elastic Optical Networks (EONs) can facilitate elastic network operations, orchestration between the cloud and the interconnection network is eventually required to coordinate resources in both strata in a coherent manner.

In this thesis, we study connectivity requirements from representative cloud-based services and explore connectivity models, architectures and orchestration schemes to satisfy them aiming at facilitating the telecom cloud.

The work here presented has been developed within the Optical Communications Group. This thesis continues, in part, the work developed during the PhD thesis of Dr. Alberto Castro.

1.2 Goals of the thesis

The main objective of this thesis is demonstrating, by means of analytical models and simulation, the viability of orchestrating DCs and networks to facilitate the telecom cloud.

The following goals are defined to achieve this main objective:

G1 – Connectivity requirements for DC interconnection and services

This goal focuses on the study of a number of scenarios that require connectivity from the transport network. Specifically, we focus on studying DC federations, live-TV distribution, and 5G mobile networks.

- **G1.1 – Elastic operations in DC federations**

  This sub-goal focuses on the study of the connectivity requirements for elastic operations among DCs in DC federations, requiring huge capacity on-demand.

- **G1.2 – Services on the telecom cloud**

  This sub-goal is oriented to study services that can take advantage of telecom cloud and require stringent Quality of Service (QoS) requirements.

- **G1.3 – 5G mobile networks**

  This sub-goal focuses on the study of optical network-supported Cloud-Radio Access Network (C-RAN), where requirements in terms of QoS and bitrate guarantees are critical to ensure service.

G2 –Connectivity schemes

This goal focuses on studying different connectivity schemes, algorithms, and architectures aiming at satisfying the connectivity requirements from goal G1.

- **G2.1 – Dynamic and elastic connections**

  The objective of this sub-goal is to study polling-based models for dynamic inter-DC connectivity to facilitate elastic operations among federated DCs.

- **G2.2 – Transfer-based DC connections**

  This sub-goal focuses on studying a novel connectivity scheme where inter-DC connectivity can be delegated to the network operator.

- **G2.3 – Virtual network topology provisioning**

  The objective of sub-goal G2.3 is studying virtual network topology provisioning models to support services on telecom cloud requiring service-specific SLA parameters in terms of QoS.

G3 – Cross-stratum orchestration schemes

Goal G3 focuses on studying DC and network orchestration for a set of customers, each with its own SLA parameters, and requiring connectivity from the interconnection network.

Table 1-1 summarizes the goals of the thesis.

Table 1-1 Thesis goals

<table>
<tr><th colspan="2">Goals</th></tr>
<tr><td rowspan="3">G1
Connectivity requirements for DC interconnection and services</td><td>G1.1
Elastic operations in DC federations</td></tr>
<tr><td>G1.2
Services on the telecom cloud</td></tr>
<tr><td>G1.3
5G mobile networks</td></tr>
<tr><td rowspan="3">G2
Connectivity schemes</td><td>G2.1
Dynamic and elastic connections</td></tr>
<tr><td>G2.2
Transfer-based DC connections</td></tr>
<tr><td>G2.3
Virtual network topology provisioning</td></tr>
<tr><td colspan="2">G3
Cross-stratum orchestration schemes</td></tr>
</table>

1.3 Methodology

To carry out the studies presented in this thesis, the methodology illustrated in Fig. 1-1 was followed.

As the starting point of each study, an idea related to a thesis objective, is conceived. Then, due to the nature of this thesis' goals, an optimization problem is devised and formally stated, i.e., mathematically formulated as a Mixed Integer Linear Programming (MILP). The mathematical model is then implemented in Matlab [MATLAB] and problem instances solved using IBM's commercial solver CPLEX [CPLEX]. Because the realistic scenarios tackled in this thesis lead to large problem sizes in most of the studies, MILP formulations require long computation times; therefore, for those scenarios requiring very short computation times (e.g., once in operation or to manage dynamic connection requests) heuristic algorithms are required. To that end, once a MILP formulation is validated, an algorithm is designed. To run simulations, the algorithm is implemented in the event-driven

OMNeT++ [OMNETPP] simulator. Some works used the cloud management OpenNebula [ONEBULA] framework.

Next, the performance of the algorithm is evaluated as follows: first, its performance is compared against that of solving the MILP, and, if required, the algorithm is revised; then the performance of the algorithm is compared against a certain benchmark (e.g., other algorithms or the optimal solutions). Finally, relevant results are disseminated and considered as the conception of a new idea requiring further research.

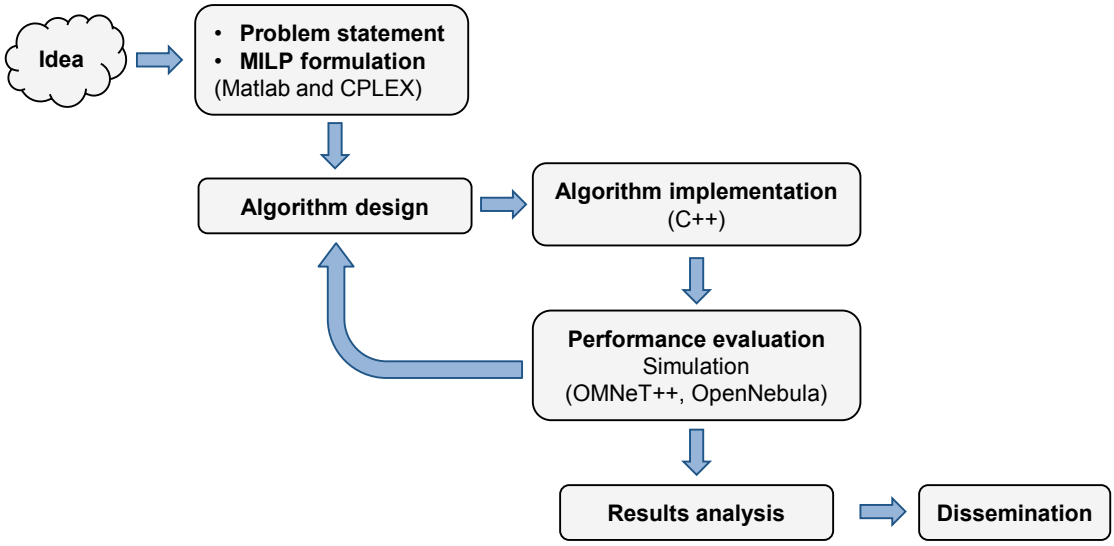

Fig. 1-1 Methodology.

1.4 Thesis outline

The remainder of this thesis is as follows:

Chapter 2 briefly introduces the main background related to EON, cloud computing, including DC federations and telecom cloud, as well as the basics on live-TV distribution, Radio Access Networks (RAN) and NFV.

Chapter 3 explores the related works relevant to the thesis objectives, such as elastic operations in DC federations, inter-DC connectivity, DC and networks orchestration, live-TV distribution, C-RAN, and virtual networks.

Chapter 4 studies elastic operations in DC federations (goal G1.1). To that end, this chapter presents the Elastic Operations in Federated DCs for Performance and Cost Optimization (ELFADO) problem, where federated DCs are orchestrated and workload in each DC is scheduled taking into account not only energy costs but also communication costs related to data migration among them. This chapter is based on the already published work [Ve14.1].

Chapter 5 studies live-TV distribution (goal G1.2) and C-RAN (goal G1.3) as services that can be deployed on telecom cloud. We explore architectures with different centralization levels and their impact on CAPEX for these services. In addition, this chapter summarizes connectivity requirements for DC interconnection resulting from these studies and the study in the previous chapter. This chapter is based on the already published work [As15.4] and the accepted paper [As16.3].

Chapter 6 proposes a cross-stratum orchestrator (CSO) (goal G3) and studies two dynamic connectivity models (dynamic and dynamic elastic) that allow DC operators to manage connection requests to Application-Based Network Operations (ABNO) architecture in charge of the interconnection network (goal G2.1). In addition, network capabilities to implement elastic operations on already established connections are analyzed. This chapter is based on the already published works [Ve13.1] and [As13.2]

Chapter 7 studies polling-based and notification-based models and motivates the need of transfer- based connections (goals G2.1 and G2.2). The Stochastic-ELFADO (STC-ELFADO) problem is presented and polling-based and notification-based models are studied to solve it. This chapter is based on the already published work [As15.1].

Chapter 8 proposes an Application Service Orchestrator (ASO) on top of ABNO (goal G3) and transfer-based operations for inter-DC connectivity (goal G2.2). Both ASO and transfer-based operations are studied in scenarios where the network operator is responsible for managing the interconnection between DCs while simultaneously fulfilling SLA of a set of customers. This chapter is based on the already published works [Ve14.2] and [As14.1].

Chapter 9 takes advantage of ASO on top of ABNO to study virtual network provisioning to support service-specific SLA parameters (goal G2.3 and G3). This chapter is based on the already published works [As15.2] and [As16.4].

Finally, Chapter 10 presents the main conclusions of the thesis.

Chapter 2

Background

In this chapter, we introduce the main concepts and technologies that serve as a starting point for the research work described in this thesis. Firstly, concepts related to cloud computing, such as virtualization, DC federations and DC architecture, and power model are reviewed. Then, the main concepts related to EON, flexgrid technology, provisioning, recovery and multilayer networks are introduced. Next, control and management architectures for DCs and optical networks are described. After that, the telecom cloud is introduced and the basics on live-TV distribution, future RAN to support 5G networks, and NFV-related concepts are presented.

2.1 Cloud computing

Cloud computing has transformed the IT industry shaping the way IT hardware is designed and purchased [Ar10]. DCs contain hardware and software to provide services over the Internet. Such services result in extremely dynamic workload mixes and intensities; therefore, dimensioning DCs is a challenging task: dimensioning DCs for the peak load can be extremely inefficient, whereas reducing its capacity might result in a poor QoS causing SLA breaches. In addition, the huge energy consumption of DCs requires an elastic resource management, e.g., by turning off physical machines (PMs) when they are not used or turning them on to satisfy increments in the demand.

In this section, background related to cloud computing and energy consumption in DCs is presented.

2.1.1 Virtualization

Thanks to virtualization, mixed workloads (e.g., web applications and high performance computing jobs) can be easily consolidated and performance isolated, their consumptions tailored and placed in the most proper PM according to its performance goals. By encapsulating jobs in Virtual Machines (VMs), a cloud resource manager can migrate jobs from one PM to another looking for reducing energy consumption (or some other OPEX objective) while ensuring the committed Quality of Experience (QoE). In addition, some web applications are encapsulated into VMs and replicated across DCs to keep services close to their users and for load balancing purposes; each replica generates some amount of data that need to be propagated to other replicas.

2.1.2 DC federations

Large Internet companies, such as Google or Microsoft, own their infrastructures that usually consist in a number of large DCs. These DCs, placed in geographically diverse locations, guarantee good QoE to users and are interconnected through a wide area network (WAN) [ZhVu13]. Using that scheme, those companies can move workloads among DCs to take advantage of reduced energy cost during off-peak energy periods in some locations (in addition to load balancing) while using green energy when is available in some other locations and turning off PMs when they are not used; thus, minimizing their energy expenditure.

Nonetheless, there exist a large number of smaller independently operated infrastructures, which cannot perform such elastic operations. Notwithstanding, those medium-size DCs can cooperate by creating DC federations [Go10] to increase their revenue from using IT resources that would otherwise be under-utilized, and to expand their geographic coverage without building new DCs.

2.1.3 DC architecture

The internal DC architecture has become crucial to deploy energy-efficient infrastructures. A certain number of switches is necessary to provide connectivity among servers in the DC and to interface the DC with the Internet. Consequently, according to the DC architecture being adopted, a corresponding power is consumed, basically dependent on the number and type of switches used. Among the various intra-DC architectures studied in literature (see [ZhAn13] for a detailed survey), the so-called *flattened butterfly* architecture has been identified as the most power-efficient DC architecture thanks to its power-proportional behavior, i.e., its power consumption is proportional to the number of currently used PMs. However, the most widely-deployed architecture for DC is the so-called *fat-tree* topology [Fa08], which is based on a hierarchical structure, where large higher-order switches represent the interface of the DC towards the network

infrastructure and are connected to the servers via a series of lower-order switches providing the intra-DC connectivity.

In this thesis, we assume the *fat-tree* architecture (Fig. 2-1), which consists of three switching layers; from top to bottom: *Core*, *Aggregation,* and *Edge*.

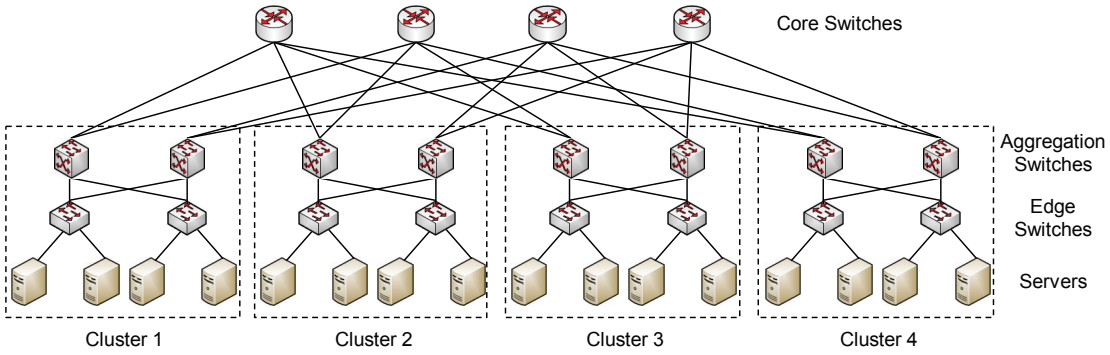

Fig. 2-1 Example of fat-tree DC architecture (M = 4).

The lower layers, aggregation and edge, together with the servers are organized in a number of clusters M. In each of these clusters, switches have M interfaces operating at the same bitrate. Each cluster has $M/2$ edge switches and $M/2$ aggregation switches, all with M ports; it constitutes a *bipartite* graph by connecting each edge to every aggregation switch. In each edge switch, $M/2$ ports are connected directly to servers and the other $M/2$ ports are connected to the aggregation switches. Thus, each cluster has $M^2/4$ servers and there are $M^3/4$ servers in total in the DC. In addition, there are $(M/2)^2$ M-port core switches, each having one port connected to each cluster, whilst each cluster is connected to every core switch.

2.1.4 DC power model

Two main contributions to the power consumption of a DC can be distinguished: *i*) the power consumed by IT devices, P_{IT}, which comprises both the servers located in the DC, as well as the switches employed to interconnect those servers; *ii*) the power consumption of the non-IT equipment, $P_{non\text{-}IT}$, such as cooling, power supplies, and power distribution systems. Thus, total power consumption of a DC can be computed as $P_{DC} = P_{IT} + P_{non\text{-}IT}$. P_{IT} can be easily estimated by counting the number of servers and switches of a DC and computing their energy consumption according to their load. However, it is difficult to evaluate the power consumption of non-IT devices since it depends on several details and factors which cannot be easily estimated. For instance, the power consumption of the cooling system strongly depends on the geographical location of the DC and on the building hosting that DC.

An indirect way to estimate a numerical value for $P_{non\text{-}IT}$ is to consider the Power Usage Effectiveness (PUE) metric [GreenGrid]. PUE can be used as a measure of

the energy efficiency of a DC and quantifies the amount of power consumed by non-IT equipment in that DC: $PUE = P_{DC} / P_{IT}$. Therefore, if P_{IT} and PUE can be estimated for a given DC, the total power consumed in a DC can be computed as $P_{DC}=PUE \cdot P_{IT}$.

Regarding P_{IT}, we can distinguish between the power consumed by the servers and by network equipment. The power consumed by a server, $P_{server}(k)$, depends mainly on the CPU load (k) utilization, expressed as the ratio between the current load and the maximum capacity of the server. According to [Fa07], the power consumption of a server can be estimated as $P_{server}(k) = P_{server\text{-}idle} + (P_{server\text{-}max} - P_{server\text{-}idle}) \cdot k$, where $P_{server\text{-}idle}$ and $P_{server\text{-}max}$ represent the power consumed by the server when it is idle and when it operates at its maximum capacity, respectively. The power consumed by network equipment depends on the specific architecture of the DC.

We consider that clusters are active when one or more servers are loaded; otherwise the complete cluster is turned-off. Then, the power consumption of cluster i, $P^i_{cluster}$, can be estimated as,

$$P^i_{cluster} = a^i \cdot \left(\frac{M}{2} \cdot \left(P_{agg} + P_{edge} \right) + \sum_{s=1}^{M^2/4} P_{server}(k^i_s) \right), \tag{2.1}$$

where a^i indicates whether the cluster is active and P_{agg} and P_{edge} denote the power consumption of aggregation and edge switches. The power consumption of the IT devices in the DC can eventually be computed as follows, where P_{core} denote the power consumption of core switches.

$$P_{IT} = \frac{M^2}{4} \cdot P_{core} + \sum_{i=1}^{M} P^i_{cluster} \tag{2.2}$$

2.2 Elastic optical networks

On the contrary to wavelength division multiplexing (WDM) fixed-grid networks, in which the spectrum width of optical channels is constant and equal for every connection (lightpath), in EON the spectral resources allocated to a connection may be expanded or reduced when the required bitrate of the connection increases or decreases.

2.2.1 Flexgrid technology

Among the technologies facilitating EON, *flexgrid* [Ji09], [Ji10], [Ge12] highlights as a potential candidate; flexgrid uses the flexible-grid defined in [G694.1]. According to [G694.1], the optical spectrum is divided into equally-sized frequency *slices* (e.g., 12.5 GHz). Interestingly, by using a variable number of consecutive

frequency slices arbitrary-sized frequency *slots* can be created and allocated to lightpaths; thus, allowing to elastically adapt the allocated resources to the connections' capacity requirements.

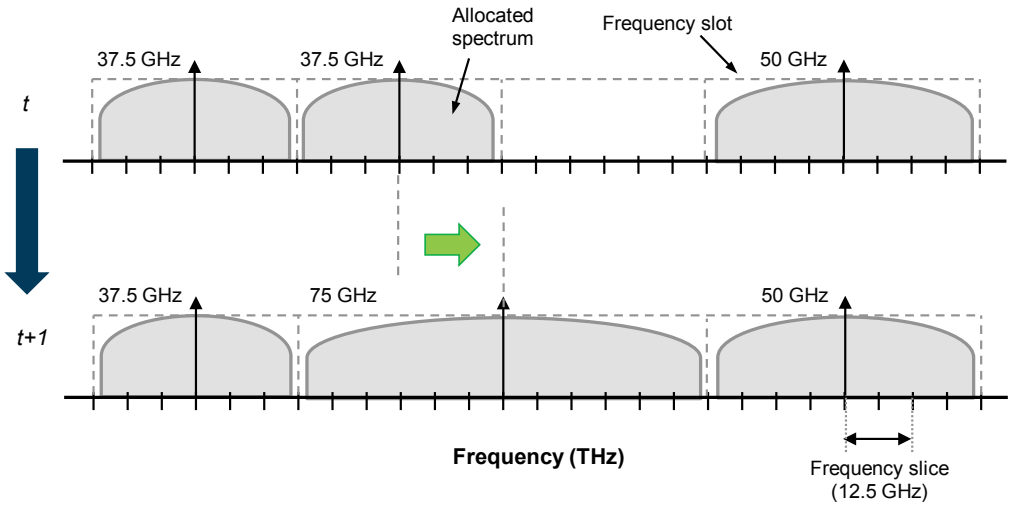

Fig. 2-2 Example of flexible and elastic spectrum allocation.

Fig. 2-2 illustrates flexible and elastic spectrum allocation (SA), where resources to support three connections requiring 37.5 GHz, 37.5 GHz, and 50 GHz are allocated at a given time *t*. In the example, at time *t*+1, the resources allocated to support the connection in the middle, have been increased, from 37.5 GHz to 75 GHz, to satisfy the demanded capacity at that time.

In [G694.1], the nominal *central frequency* (CF) granularity, i.e., the spacing between neighboring CFs, is equal to 6.25 GHz.

Additional key technologies that are paving the way to devise novel EON architectures based on the flexgrid technology are:

- The availability of flexgrid ready bandwidth-variable wavelength selective switches to build bandwidth-variable optical cross-connects (BV-OXC), in comparison with traditional optical cross-connects (OXC).
- The development of advanced modulation formats and techniques, both single-carrier (such as k-PSK, k-QAM) and multi-carrier (such as O-OFDM), to increase efficiency and being capable of extending the reach of optical signals avoiding expensive electronic regeneration (known as 3R).
- Bandwidth-variable transponders (BVTs) and multi-flow transponders (also known as sliceable BVTs, SBVTs) that are able to deal with several flows in parallel; thus, adding even more flexibility and reducing costs [Ji12].

2.2.2 Provisioning

Aiming at providing end-to-end (e2e) connectivity, connection requests (i.e., demands) to be served in the network need to specify some information such as source and destination nodes and required capacity. Then, similarly to the Routing and Wavelength Assignment (RWA) problem in WDM networks, the Routing and Spectrum Allocation (RSA) problem needs to be solved in flexgrid networks to find lightpaths. The objective of the RSA problem is to find the shortest route between source and destination nodes having a frequency slot with enough spectral resources to serve the required bandwidth to provide the capacity of the traffic demands, while satisfying the *contiguity* and *continuity* constraints of the spectrum.

For illustrative purposes, Fig. 2-3 represents a 4-node flexgrid network with three lightpaths (a) and the spectral resources allocated to them (b). Lightpaths p_1, blue line in the figure, from X1 to X4, and p_2, red line in the figure, from X1 to X2, use 4-slice frequency slots; whereas lightpath p_3, yellow line in the figure, from X4 to X3, uses an 8-slice slot. Note that, in all cases, slices are contiguous in a given link and slots remain continuous along links in the path.

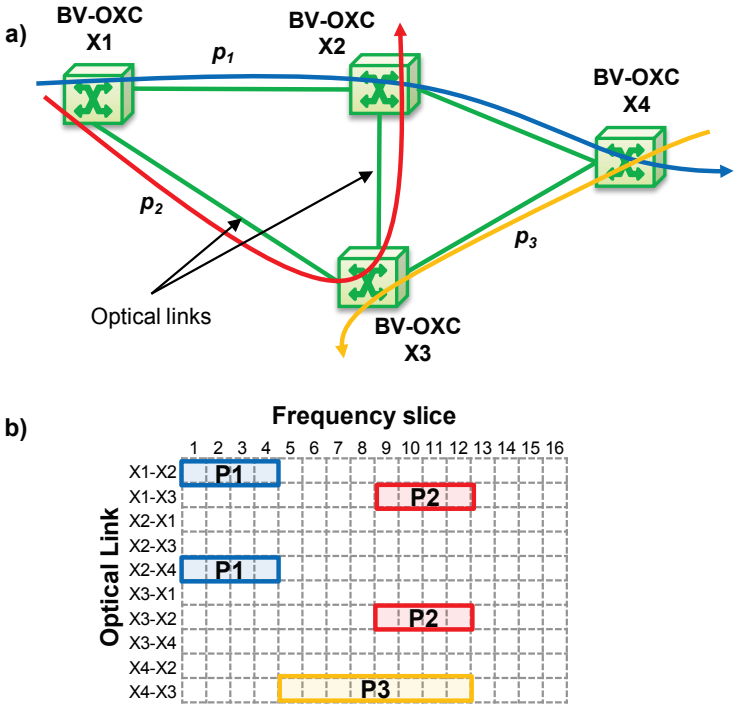

Fig. 2-3 Example of flexgrid-based network with three lightpaths established.

2.2.3 Recovery

Once in operation, established lightpaths may be affected by failures in the network's physical resources. Aiming at reducing network failures' impact on the service, recovery techniques based on disjoint paths can be applied. To that end, considering shared-risk link groups (SRLGs) [St01], [Li02], [Se01] allows to identify sets of resources that are affected by the failure of a physical resource in the network.

Among the different recovery techniques that are available in the literature, in this thesis we focus on *partial protection* and *diversity* techniques to guarantee certain portion of the bitrate in the event of a failure in a link in the network. In partial protection, two SRLG-disjoint paths are established: a working path and a protection path; the latter with enough capacity to satisfy the bitrate to be guaranteed [Gr04]. Another, option is *diversity*, where two SRLG-disjoint paths are set-up with a combined capacity to satisfy the one requested and being the minimum individual capacity equal to the one to be guaranteed.

Fig. 2-4 illustrates a scenario where lightpaths to support a 10 Gb/s connection are represented considering both partial protection and diversity techniques. In both cases, 5 Gb/s need to be guaranteed in the case of a single failure in any optical link. In the example, each optical link has been arbitrarily assigned to a SRLG; i.e., SRLG = {1}, {2}, {3}, {4} or {5}. When partial protection is considered, a primary path (working path p_w) supports the 10 Gb/s connection, whereas a secondary path (backup path p_b), SRLG-disjoint with the primary path, supports a 5 Gb/s connection in the event of a failure of a link in the route of the primary path.

When diversity is considered, two primary paths (p_w and p'_w), SRLG-disjoint between them, support jointly the required 10 Gb/s connection; in the example, each supports 5 Gb/s; thus, guaranteeing a 5 Gb/s connection in the case of a failure of a link in the route of p_w or p'_w.

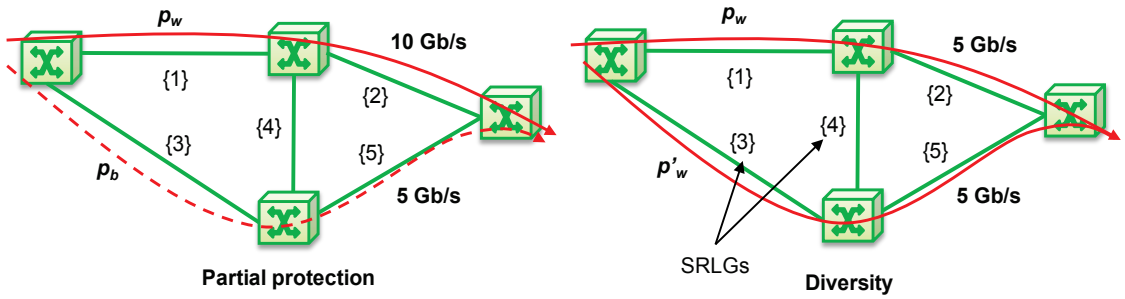

Fig. 2-4 Example of SRLG-disjoint paths under partial protection and diversity.

2.2.4 Multilayer networks

Optical networks can be deployed to support IP-over- Multi-Protocol Label Switching (MPLS) [RFC3031] networks, referred to as IP/MPLS networks, on top.

A demand in the IP/MPLS layer is routed in the corresponding virtual network topology (VNT). A Label-Switched Path (LSP) between two IP/MPLS routers is set-up using virtual links, which in turn, are supported by the corresponding lightpaths in the underlying optical network. Fig. 2-5 illustrates an IP/MPLS network over an optical network with an MPLS LSP established between IP/MPLS routers R2 and R3. In the figure, three virtual links can be identified in the IP/MPLS layer; i.e., virtual link between R1-R3, between R1-R2 and between R2-R4. In addition, for illustrative purposes, the lightpaths supporting those virtual links are depicted (red lines in the figure). Since routers operate in the electrical domain, transponders need to be used to convert signals from the electrical to the optical domain; BVTs and SBVTs can be considered when the underlying optical network is based on the flexgrid technology.

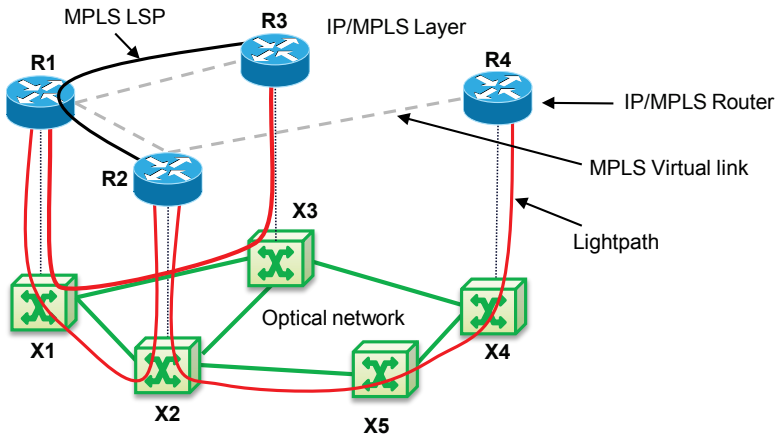

Fig. 2-5 Multilayer network.

2.3 Control and management architecture

2.3.1 Transport networks

Core network control architectures based on a centralized element such as the Path Computation Element (PCE) [RFC4655] are commonly used to control optical networks' data planes.

Although in this thesis we assume a PCE on top of the control architecture, a Software Defined Network (SDN) controller would result in similar conclusions. The PCE includes, among others, routing algorithms, the LSP database (LSP-DB) and Traffic Engineering Database (TED). The LSP-DB stores information related to currently established connections, whereas TED stores the state of network resources. Fig. 2-6a represents a generic PCE-based control architecture on top of a transport network.

Among control architectures based on the PCE, the Internet Engineering Task Force (IETF) has recently standardized a centralized architecture referred to as ABNO [RFC7491]. ABNO is defined as an entity in charge of controlling the network in response to requests from the application layer or the Network Management System (NMS). The ABNO architecture consists of a number of functional elements, such as:

- The ABNO *controller* receiving high-level requests and translating them into simpler operations that can be performed by other functional elements.
- A *policy agent* which enforces the set of policies received from an NMS.
- A *PCE* [Cr13] to perform path computation; in particular, an active stateful PCE is able to modify already established LSPs.
- The LSP-DB and TED *databases*.
- A *provisioning manager*, e.g., an SDN controller [ONF], in charge of implementing connections in the network.
- An *operations*, *administration*, and *management* (OAM) handler that receives notifications from the network elements.

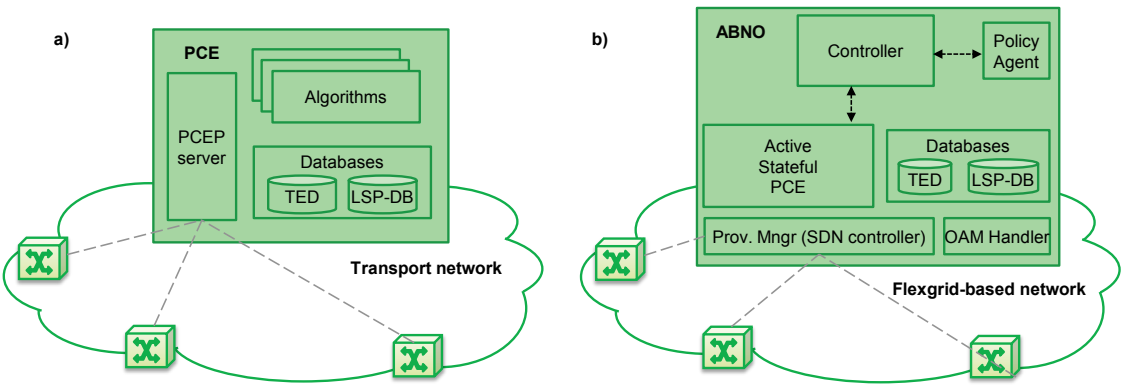

Fig. 2-6 Example of PCE-based control architectures.

Fig. 2-6b illustrates the ABNO architecture, including the elements described above, to control a flexgrid-based network.

2.4 Telecom cloud

Transport networks are currently configured with big static fat pipes based on capacity over-provisioning. The rationality behind that is guaranteeing traffic demand and QoS. The capacity of each inter-DC optical connection is dimensioned in advance based on some volume of foreseen data transference. Once in operation, scheduling algorithms inside cloud management run periodically trying to optimize

some cost function, such as energy costs, and organize data transferences. To avoid transference overlapping, which may eventually lead to performance degradation, some over-dimensioning is needed. Obviously, this static connectivity configuration adds high costs since large part of connectivity capacity remains unused in low periods.

Thus, demands of cloud services require new mechanisms to provide reconfiguration and adaptability of the transport network to reduce the amount of over-provisioned bandwidth; the efficient integration of cloud-based services among distributed DCs, including the interconnecting network, becomes then a challenge. *Cloud-ready transport network* [Co12] was introduced as an architecture to handle this dynamic cloud and network interaction allowing on-demand connectivity provisioning.

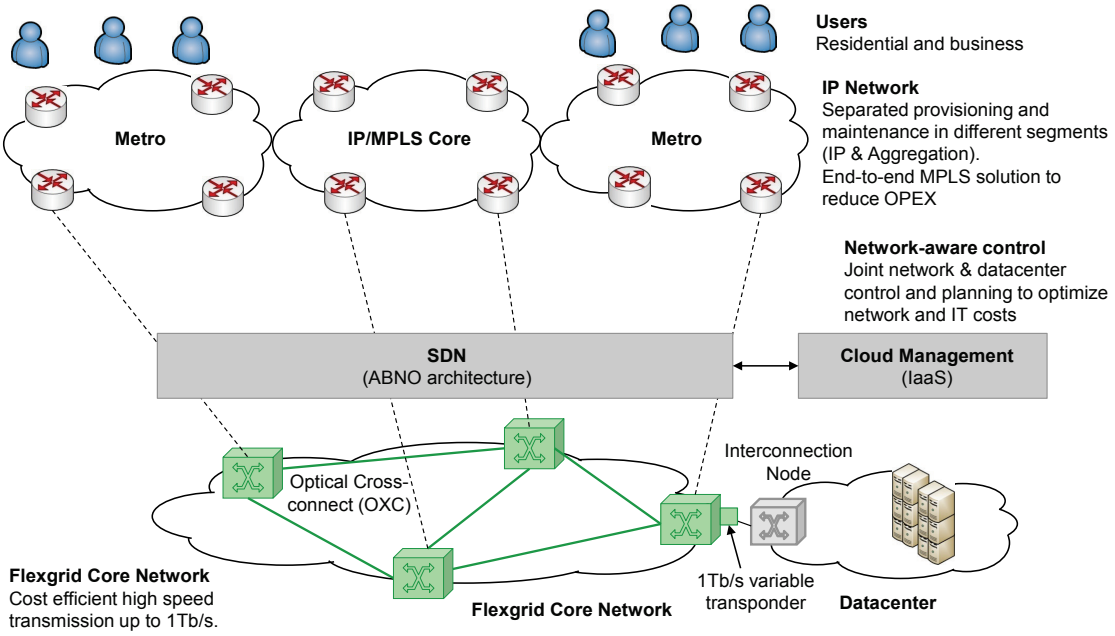

Fig. 2-7 Architecture to support cloud-ready transport networks. Reproduced from [Co12].

Evolution towards cloud-ready transport networks entails dynamically controlling network resources, considering cloud requests in the network configuration process. Hence, that evolution is based on elastic data and control planes, which can interact with multiple network technologies and cloud services. Cross-stratum orchestration between cloud and interconnection network is eventually required to coordinate resources in both strata in a coherent manner. The reference architecture to support cloud-ready transport networks proposed in [Co12] is reproduced in Fig. 2-7. An ABNO architecture interfacing users (e.g., service-layer) and controlling a flexgrid network is assumed and communication between ABNO and the cloud management devised.

In addition to facilitate on-demand connectivity provisioning thanks to the cloud-ready transport network, the telecom cloud concept includes, among others, the ability to provide the infrastructure to deploy a service, e.g., taking advantage of current telecom facilities, such as central offices (COs), which can be used as DCs.

Among the wide range of IP-based services requiring resources from the transport network and that could be deployed in a telecom cloud, in this thesis we focus on live-TV distribution and the evolution of Radio Access Network, referred to as RAN, towards a centralized approach to support 5G networks. The next subsections briefly introduce the main concepts related to them.

2.4.1 Live-TV distribution

Video signal distribution is one of the stringent and popular services that telecom networks need to provide. The bandwidth needed to convey a video stream is actually determined by its quality. In the live-TV broadcasting industry, uncompressed video streaming formats are used before the video is produced. Notwithstanding, stringent QoS is required since uncompressed video streaming in the 4K Ultra-High Definition (UHD) TV format ranges from 6 to 48 Gb/s, according to ST 2036-1 [ST2036-1]. In addition, 4K UHD digital cinema has been standardized and commercialized in the movie industry, and 8K quality is in the roadmap of some operators [Na11]; uncompressed real time 8K transmission needs 72 Gb/s connections.

Once the video has been produced, distribution to end-users is based on compressed video, which quality is adapted to the one that fits better the user's device. Compressed streams for video distribution require up to hundreds Mb/s, depending on its quality, i.e., Standard Definition (SD), High Definition (HD) or UHD. Digital TV and online video are expected to show the higher penetration percentages among the residential services; in fact forecasts show that 79% of the global IP traffic will be related to video traffic by 2018 [VNI14].

2.4.2 Radio access networks to support 5G networks

The growth of mobile devices and the wide variety of services accessing the Internet highly contribute to the explosion of data to be conveyed [VNI13] not only between users' devices but also between machines. As a result, an ever-increasing Total Cost of Ownership (TCO), which includes CAPEX and OPEX, is expected when traditional RAN architectures are considered. Aiming at reducing TCO, centralized RAN ([Li10], [CMR]) architectures were proposed to support next generation mobile networks.

Among the main factors contributing to CAPEX increase, highlight the new sites building, radio frequency (RF) and baseband hardware, and power and cooling

equipment acquisition. Factors contributing to OPEX increase are, among others, site rental and power consumption.

In traditional distributed RAN architecture, RF and baseband processing hardware is co-located in the cell site and not shared among different sites. In centralized RAN architectures, baseband processing in baseband units (BBUs) is not only separated from RF processing hardware, i.e., remote radio heads (RRHs), but also centralized and can be shared among different sites. Fig. 2-8 illustrates both distributed and centralized RAN architectures, showing that in distributed RAN, RF and baseband hardware are co-located in the site and not shared with other sites; whereas in centralized RAN, BBUs from different sites are co-located in the same BBU pool and can be shared among different RRHs along the time.

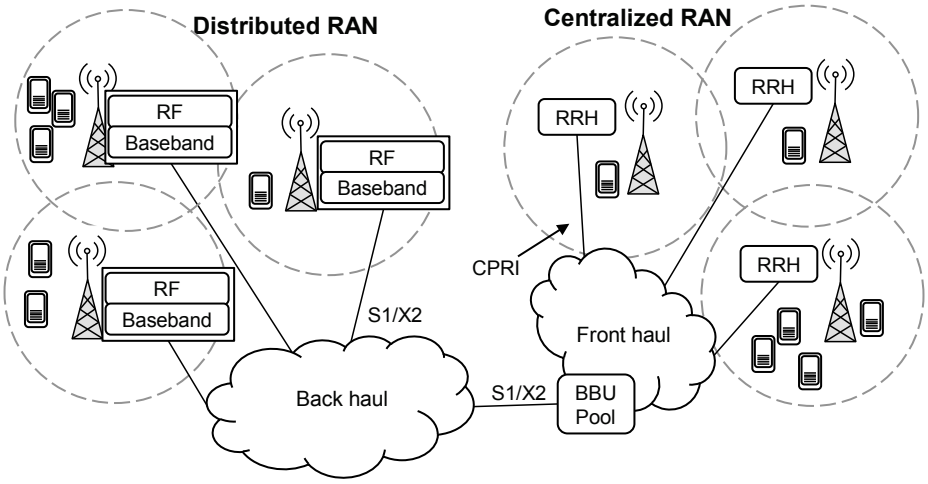

Fig. 2-8 Distributed and centralized RAN architectures.

According to [Pf15], centralized RAN architectures can be implemented in the cloud, where virtualized BBU pools hosted in different central locations can be flexibly configured and serve RRHs from different virtualized BBU pools each time (in this thesis, we refer to this case as Cloud-Radio Access Network, C-RAN).

From the mobile core network perspective, considering Long-Term Evolution (LTE) or LTE-Advanced (LTE-A) RAN architectures, both distributed and centralized architectures require to interconnect base stations and their peering point through a backhaul network (e.g., and IP/MPLS network supported by WDM). In addition to backhaul connections transporting user and control data (S1 interface), interconnection among neighboring cells' base stations may also be required (X2 interface). In current LTE RAN architecture, latencies in the order of tens of milliseconds are allowed in S1 interfaces, whereas for tight coordination schemes between base stations, delays below 1 ms are required in X2 interfaces.

Moreover, compared to distributed RAN, centralized RAN architectures require a fronthaul network aiming at providing connectivity between RRHs and BBUs in remote BBU pools and convey radio interface data. Among the different radio

interface protocols, Common Public Radio Interface (CPRI) [CPRI] is widely used. CPRI is a bidirectional protocol and its bitrate is constant and depends on the cell site configuration. Fig. 2-8 illustrates the logical links supporting S1 and X2 interfaces in both distributed and centralized RAN architectures and CPRI in the centralized approach. In LTE and LTE-A architectures, CPRI requires not only huge capacity (in the order of Gb/s and tens of Gb/s), but also strict delay constraints, i.e., in the order of few hundreds μs round trip time (RTT) delay.

2.5 Network functions virtualization

Among the wide range of services over IP networks, those related to NFV are of particular interest. Expected NFV benefits are leading the research community and industry to explore network architectures and technologies to satisfy new challenges arising (see [GSNFV], [VeCo15]).

Based on the ABNO architecture and aiming at facilitating network resources virtualization to support NFV, the IETF is working on the Abstraction and Control of Transport Networks (ACTN) framework [Ce15]. The business model in the ACTN framework defines three key entities: customers, service providers, and network providers. Customers can request on-demand connectivity between their end-points (EPs) using a Customer Network Controller (CNC) to modify their virtual topology to satisfy service requirements, referred to as customer virtual network (CVN). Since network operators can act both as service and network providers, a generic landscape with two main actors is devised:

- network operator, which owns the network infrastructure. Since the network operator controls the network infrastructure, is aware of resource availability, and is able to collect performance monitoring data, such as effective throughput and delay, and correlate them into QoS indicators;
- set of customers requiring virtual network services to connect EPs in geographically disperse locations, e.g., a Virtual Network Functions (VNFs) orchestrator requiring on-demand connectivity between DCs to implement service chaining among VNFs.

2.6 Conclusions

Relevant background related to cloud computing, EON, telecom cloud, and virtualization has been presented in this chapter. First, cloud computing concepts were introduced as well as architectural and energy consumption aspects related to DCs and DC federations. Next, EON was reviewed. Flexgrid technology was introduced as a promising candidate technology to support EON. Provisioning, recovery basics, and the main concepts related to flexible-grid and multilayer

networks were presented. Next, architectures to manage and control DCs and transport networks were reviewed. The telecom cloud was introduced and main concepts related to live-TV and RAN to support 5G networks were described. Finally, NFV was introduced.

Chapter 3

Review of the State-of-the-Art

In this chapter, the state-of-the-art related to the goals of this thesis introduced in Chapter 1 is reviewed. Relevant works that can be found in the literature are described and the niches that motivate the studies in this thesis are clearly identified.

3.1 Connectivity requirements for DC interconnection, live-TV distribution and 5G mobile networks

In this section, we describe relevant works related to elastic operation in DCs, approaches for live-TV video signal distribution in the telecom cloud, and recent research studies to facilitate 5G networks.

3.1.1 Elastic operations in DC federations

As described in Chapter 2, consolidation and load balancing techniques in DCs require VM migration among servers hosted within the same DC or in remote DCs. The authors in [Mi12], presented the details of VM migration aiming at providing dynamic resource management considering different goals, including consolidation and load balancing. For each goal, the main questions to address were described: when to migrate, which VMs to migrate and to where VM must be migrated. Among others, a periodic approach was discussed considering that a given DC, placed in certain region of the world, may be highly loaded during daytime hours, whereas during the night hours its load may be considerably lower. In addition, the idea of placing the VMs close to the users depending on the time of the day was

introduced. Consolidation was described as a technique that may result in energy savings. Although VM migration was mainly considered in local area networks (LANs), the authors described issues to be addressed when WANs are considered. Despite the limited bandwidth and high-latency of the current WANs, *follow-the-sun* strategies can result in OPEX savings in scenarios considering geographically distributed DCs. The scenario description for dynamic resource management in DCs studied in [Mi12] motivates, in part, the works in this thesis related to elastic operations in DCs.

Since minimizing energy expenditures is really important for DC operators, many papers can be found in the literature partially addressing that problem. In [Go12], the authors proposed scheduling workload in a DC coinciding with the availability of green energy, consolidating all the jobs in time slots with solar energy available, increasing green energy consumption up to 31%. Specifically, the authors proposed GreenHadoop, focused on data-processing frameworks. Data-processing jobs were scheduled in a single-DC within certain time bounds and the amount of available solar energy was predicted aiming at properly placing workloads to maximize green energy consumption. Additionally, brown energy costs were taken into account in the event that power needs to be drawn from the grid so as to satisfy job's time constraints. The authors in [Li11], referred to the problem of load balancing DC workloads in internet-scale systems based on a *follow-the-renewals* strategy, i.e., following green energy availability, to reduce the amount of brown energy consumed focusing mainly on wind energy and the capability to store energy.

Based on VM consolidation, authors in [Li09] proposed a DC architecture that is able to reduce power consumption, while guaranteeing QoE to the users. They considered a monitoring service providing information to a migration manager that takes decisions regarding VMs optimal placement accordingly triggering the corresponding actions for VM migration and switching on/off PMs and networking equipment. Energy savings of up to 27% were reported.

Although taking energy from green sources and job consolidation to allow switching off equipment translates into power consumption reduction, the work in [Pi11] describes a much more complex scenario. The author remarks the idea that all IT equipment counts when consuming energy and highlights that additional parameters related to the quality of the green energy and its production and transportation are important factors that need to be considered.

From the DC interconnection perspective, the network can be based on the optical technology, since elastic operations for VM migration require huge bitrate to be available among DCs for some time periods. Some works consider optical networks to interconnect DCs. For instance, the authors in [Bu13] presented routing and scheduling algorithms and compared energy savings with respect to a scenario in which routing and scheduling problems were solved separately. Aimed at minimizing the CO_2 emissions related to DCs by following the current availability

of renewable energies, authors in [Ga14] devised routing algorithms for connections supporting cloud services.

To the best of our knowledge, no work compares the way to compute scheduling in DC federations considering both energy and communications costs in a single framework. Therefore, in this thesis, elastic operations in DCs are studied jointly considering IT equipment in DCs and the transport network.

3.1.2 Live-TV distribution on the telecom cloud

In a recent demonstration, the authors in [Po14] reported 4K UHD TV video streaming over an IP network; thus, enabling the migration from traditional Serial Digital Interface (SDI) -based transmission to all-IP environments. The authors demonstrated that current IP-network infrastructure is able to transport uncompressed 4K UHD flows. However, since traffic sourced by different services is mixed in the transport network, accurate management of network resources is required aiming at preserving video signal integrity.

In addition, although current IP networks can transport uncompressed UHD video signals, the authors in [Na11] showed that the current IP-technology cannot scale to satisfy expected traffic-growth to support that video quality, being the huge energy consumption one of the main drawbacks. The authors proposed using optical circuit switching and the concept of dynamic optical-path network (DOPN) while discussing how it could scale from LAN to WAN. Dense WDM (DWDM) was highlighted as a candidate enabling technology to support DOPN. Later, in [KuTa13], the authors experimentally demonstrated dynamic optical-path switching to transport real UHD uncompressed signals in DOPN-based LAN. Recently, the authors in [Ku15] experimentally demonstrated the value of DOPN by conveying live 8K-video signals. These works were based on the DOPN, which is focused on reducing the huge energy-consumption of current IP equipment.

Although experimental demonstrations to convey UHD uncompressed signals in IP and DOPN-based scenarios are available in the literature, research work has been carried out neither studying the scalability of telecom cloud-based architectures for video signal processing nor live-TV distribution from the network CAPEX perspective.

3.1.3 5G mobile networks

Next generation mobile networks are attracting the interest of the research community from different points of view: from improving radio resources usage to novel architecture designs and network convergence. Research works focusing specifically on centralized RAN architectures, which were originally proposed in [Li10] and [CMR], are of particular interest for this thesis, since centralized RAN architectures can be explored from the telecom cloud perspective. In [CMR], the

authors described the requirements and technical challenges in C-RAN, and a large scale BBU pool that they developed, which was based on the fat-tree topology as well as a C-RAN prototype based on IT equipment using general purpose computing hardware.

The authors in [Ch14] presented the problem of optimizing small cells deployment to take advantage of C-RAN. They studied cell traffic profiles (TPs) and showed the conditions that impact the statistical multiplexing gain when centralized BBU pools are considered against traditional distributed BBU placement. Savings as high as 70% were reported in the number of required BBUs. Then, the authors proposed using an Ethernet-based fronthaul, aiming at dynamically assigning RRHs to BBU pools. Both BBU cost and fiber cost per km were considered and the ratio between them was analyzed to provide a threshold to discern between scenarios where deploying C-RAN is worth aiming at minimizing TCO.

Because of the stringent requirements of the several interfaces needed in centralized RAN and the maturity and evolution of different optical network technologies, optical networks have been proposed to support interfaces between RRHs and BBUs in the fronthaul (CPRI) and among BBUs (X2), and between BBUs and their peering point in the mobile core network (S1). Although the authors in [Ch14] considered Ethernet-based networks and took into account fiber cost, in their study they did not focus specifically on the optical network.

In [Po13], the authors proposed using the WDM technology to support centralized RAN in the access/aggregation network, i.e., up to the metro segment. Moreover, a practical implementation was demonstrated and links up to 10 Gb/s interconnecting RRHs and BBUs were reported. Also focusing on the same network segment, the authors in [Ca14] proposed an energy-efficient WDM aggregation network and formally defined the BBU placement optimization problem as an Integer Linear Programming (ILP) model aiming at optimizing the aggregation network in terms of power based on the Aggregation Infrastructure Power (AIP) performance indicator. They considered three architectures: two of them were based on a centralized BBU placement approach and the other was based on a distributed approach. AIP savings between 40% and 65% were shown when the centralized approaches were considered compared to the distributed one.

The authors in [Mu16] have recently proposed an ILP model for optimal BBU hotel placement over WDM networks in centralized RAN. Among others, they evaluated the impact of joint optimization of BBU and electronic switches placement on the BBU consolidation.

Finally, for the particular case of C-RAN, virtualized BBU pools running in VMs are hosted in different central locations and can be flexibly configured to serve RRHs from different virtualized BBU pools each time. The authors in [Wu15] presented a 3-layered logical structure for C-RAN taking advantage of computation in a cloud environment. Moreover, they proposed algorithms for multi-cell cooperation processing in C-RAN and compared them with equivalent algorithms

in traditional RAN. Results showed higher spectrum efficiency when the proposed algorithms for C-RAN are considered, thanks to the cloud computation resources.

Although the viability of centralized RAN and the advantages offered by the cloud environment in C-RAN have been demonstrated in different studies, to the best of our knowledge no works can be found in the literature including mathematical models and considering optical network equipment cost in centralized RAN architectures. In addition, according to [Pf15], centralized RAN architectures based on the BBU cloud variant, are still a research open topic that is worth to investigate.

3.2 Connectivity schemes

In this section, approaches to facilitate connectivity in transport networks are described from the perspective of both single connections and virtual network provisioning.

3.2.1 Dynamic and elastic connections

Although network providers still maintain a relatively rigid structure currently offering static connection provisioning, cloud-ready transport networks are expected to permit network providers to facilitate DCs interconnection by allowing DC managers to request connections' setup on-demand in a pay-as-you-go model with the desired bitrate. Those connections are torn down when they are not needed. To that end, network operators can use some automated interface to allow DC resource managers to request such connections. As an example, to facilitate an automated management system that is able to provide connections dynamically, the Open Grid Forum (OGF) [OGF] Network Services Interface (NSI) Working Group has been focused on the standardization of the NSI that allows to request connections to the network. The authors [Ku13] demonstrated the use of NSI in multi-domain network scenarios for DC interconnection. From the network side, architectures that allow dynamic connection requests need to be considered. In line with this, the ABNO architecture has attracted the interest of the network operators. ABNO's northbound interface can accept connection requests from the application layer as well as control the network based on complex algorithms and workflows.

However, dynamic connections also need to be explored from the network data plane perspective. Of particular interest are those works where EONs are studied to support connections in scenarios where the required capacity of each connection varies in time. It is worth recalling that one of the main advantages of EON is the capability to allocate spectrum resources elastically, according to traffic demands. Indeed, the resources may be used efficiently, firstly, because of the higher

granularity of the flexgrid, which allows fitting closely the allocated spectrum and the signal bandwidth and secondly, due to the elastic (adaptive) SA in response to traffic variations.

In this context, adaptive SA with a known a priori 24-hour traffic patterns has been addressed in [VeKl12.2], [Kl13], [Sh11] and spectrum adaptation under dynamic traffic demands was studied in [Ch11], [Ch13]. Concurrently, different policies for elastic SA were proposed, including symmetric [VeKl12.2], [Kl13], [Sh11] and asymmetric [VeKl12.2], [Kl13], [Ch13] spectrum expansion/reduction around a reference frequency, as well as the entire spectrum re-allocation policy [VeKl12.2], [Kl13]. First the authors in [VeKl12.2] and later those in [Kl13] showed that in EON with time-varying traffic and under a priori known (deterministic) traffic demands, elastic SA policies performing lightpath adaptation (either symmetric, asymmetric, or reallocation-based) allow to serve more traffic than when fixed SA is applied.

A different approach was described in [Ch11], where an OFDM-based network was compared against a traditional WDM-based network. The authors assumed that traffic between nodes is known based on a probabilistic traffic model and, accordingly, a set of subcarriers are pre-reserved between each pair of nodes. In addition, a set of not pre-reserved subcarriers that are shared among the different connections can be allocated/deallocated to/from connections to satisfy demands' requirements.

Although research work has been carried out to standardize interfaces to request connections dynamically and the elastic ability of EONs to adapt to traffic changes has been studied in scenarios considering off-line planning and dynamic connection requests, to the best of our knowledge the ability to dynamically request elastic operations on currently established connections remains unexplored. In addition, no work addresses the elastic SA problem under dynamic scenarios. In this thesis, dynamic elastic connections are proposed and the study of SA policies presented in [Kl13] is extended to facilitate elastic SA in dynamic scenario.

3.2.2 Transfer-based DC connections

Because dynamic operation frequently derives into not optimal use of network resources, network reoptimization can be performed triggered by some event, e.g., when a connection request cannot be served [VeKi14]. Notwithstanding its feasibility, some other techniques can be applied to improve network efficiency aiming at increasing operators' revenues.

Given that scheduling algorithms in DCs usually run periodically to evaluate the performance of the services and the quality perceived by end-users, connections could be scheduled in advance. In that regard, several works have proposed to use advance reservation (AR), where requests specify a deadline to establish the connection; e.g., the authors in [Lu13] proposed algorithms for AR requests in

flexgrid-based networks. The authors in [Lu14] proposed two algorithms for revenue-driven AR provisioning and experimentally demonstrated them in scenarios considering OpenFlow controlled Software Defined-EONs (SD-EONs). In [Sh13], dynamic advance reservation multicast was studied in DC networks considering EON and recently, in [Li15], also in multicast scenarios, the authors proposed fragmentation-aware AR algorithms in SD-EONs aiming at alleviating fragmentation.

Nevertheless, the use of AR involves both, delaying connection's set-up from the moment data volume to be transferred is calculated, and estimating the acceptable deadline; thus, adding more complexity to the cloud.

After reviewing relevant works in the literature, we found that no work explores the functionalities of an abstraction layer between cloud and network management and control architectures to manage transfer-based connections and to study scheduling techniques in addition to elastic operations. Note that this is with the twofold objective of fulfilling SLA contracts and increasing network operator revenues by accepting more connection requests.

3.2.3 Virtual network topology provisioning

Several works can be found in the literature studying topics related to virtual network topology provisioning. Among them, those tackling the particular case of virtual optical networks (VONs) and those where virtual networks are considered aiming at satisfying certain service-related requirements are the most relevant for this thesis.

Focused on scenarios requiring both VON topology reconfiguration and additional service-related requirements, the authors in [Tz13] studied VON topology reconfiguration to support uncertain traffic demands requiring service resilience and security. They introduced the concept of adaptive VONs and proposed modeling approaches to satisfy uncertain traffic demands by means of VON reconfiguration, which was based on periodic re-planning. Results showed significant benefits from adaptive VON planning to support the services, while satisfying their specific requirements. Also considering adaptability to traffic changes, the authors in [Yo14] proposed a method to control virtual networks that adapt to changes in the traffic by means of reconfiguration.

The authors in [Tz14] described a multilayer architecture and the procedures to provide virtual infrastructure and demonstrated their ability to provide optical network and IT resources provisioning for e2e cloud service delivery. Their procedures were focused on planning and re-planning.

Finally, the benefits shown from adaptive VONs and the standardization work in progress carried out by the IETF towards ACTN, opens the opportunity to devise new algorithms and connectivity models based on virtual network provisioning.

3.3 Cross-stratum orchestration

Although there are some frameworks implementing IaaS for intra-DC (see, e.g., [ONEBULA]), some even provisioning network resources on-demand enabling the so-called Network as a Service (NaaS) [OSTACK], inter-DC connectivity reveals some challenges that need to be addressed.

Huge research and standardization work have been carried out in the last years defining control and management architectures and protocols to automate connection provisioning, allowing to request connection provisioning dynamically. Starting from a distributed paradigm, control architectures have lately moved towards a centralized one led by the development of the SDN concept with the introduction of OpenFlow [OPENFLOW]. However, cloud and transport network management are still totally separated: DC operators are aware of DC resources whereas network operators are aware of the network resources and no information between them is currently shared. Therefore, cross-stratum orchestration between the cloud and the interconnection network is eventually required to coordinate resources in both strata in a coherent manner in cloud-ready network scenarios.

To facilitate cross-stratum orchestration, the interface between applications and networks needs to be considered. As described in Section 3.2, NSI enables to dynamically request circuit services, it supports connection set-up and tear down. Authors in [ZhZh13] proposed an application controller that interfaces an OpenFlow controller for the flexgrid network, similarly to the approach recently followed by Google [Ja13]. In addition, in [Dh12], the described architecture enables cross-stratum resource optimization.

However, in view of the works that can be found in the literature, no work considers a distributed CSO to coordinate cloud and network, and elasticity is currently supported neither by the NSI interface nor by any other cloud framework.

3.4 Conclusions

In this chapter, we have reviewed the state-of-the-art of the works relevant to this thesis. Works tackling elastic operations in DCs have been presented. Although several studies address the problem of scheduling workloads and VM migration aiming at minimizing energy consumption, and some works consider routing algorithms in scenarios requiring inter-DC connectivity, scheduling algorithms focused on minimizing energy consumption considering both energy and communications costs need to be explored in scenarios where DCs need resources from the interconnection network.

Regarding the services, current IP technology has been proved to allow conveying uncompressed UHD signals. However, its scalability has been argued. Similarly, works showing benefits from centralized RAN approaches to support future mobile networks, as well as considering optical-based networks in the fronthaul network have been presented. Nevertheless, no studies proposing mathematical models and studying C-RAN from the network equipment cost perspective can be found in the literature. In both cases, current works open the opportunity to study these topics from the network perspective.

Different connectivity approaches have been described in the literature. Despite the current static connection provisioning approach offered by most of the network providers, several research works have been published and standardization efforts have been carried out towards dynamic connection provisioning. In addition, works considering EONs instead of traditional WDM fixed-grid networks have been attracting the interest of the research community. Nonetheless, considering dynamic connectivity models taking advantage of EONs elastic abilities is an interesting topic that requires further study. Moreover, connectivity models beyond dynamic connections are needed aiming at facilitating cloud services requirements.

Regarding virtual network provisioning, VONs have been demonstrated as candidates to satisfy demands with service-specific requirements. Because of the standardization work currently in progress related to ACTN, on-demand virtual network provisioning is a hot topic. Virtual network provisioning needs to be studied under different service-specific requirements in dynamic scenarios.

Applications and services in the cloud may require connectivity from an interconnection network. Since both DC and network might be operated independently, cross-stratum orchestration is required to improve resource usage and coordinate both strata. The cloud-ready transport network opens the opportunity to study cross-stratum orchestration.

Finally, Table 3-1 summarizes the above survey of the state-of-the-art. As a conclusion, the goals of this thesis have been not covered so far in the literature mainly due to the novelty of the telecom cloud and the technologies to facilitate it.

Table 3-1 State-of-the-art summary

Goals		**Related work**
G1 **Connectivity requirements for DC interconnection and services**	**G1.1** Elastic operations in DC federations	Energy expenditures in DCs [Mi12], [Go12], [Li11], [Li09], [Pi11]
		Routing algorithms [Bu13], [Ga14]
	G1.2 Live-TV distribution	UHD video signal distribution [Po14] [Na11], [KuTa13], [Ku15]
	G1.3 5G mobile networks	Architecture and network [Li10], [CMR], [Po13], [Ca14], [Wu15], [Mu16], [Pf15]
		BBU pools [CMR], [Ch14], [Ca14], [Mu16]
		Mathematical models [Ca14], [Mu16]
G2 **Connectivity schemes**	**G2.1** Dynamic and elastic connections	Architectures and interfaces [OGF], [Ku13], [RFC7491]
		EONs [VeKl12.2], [Kl13], [Sh11], [Ch11], [Ch13]
	G2.2 Transfer-based DC connections	Network reoptimization [VeKi14]
		Advance reservation [Lu13], [Lu14], [Sh13], [Li15]
	G2.3 Virtual network topology provisioning	Adaptive VON and architectures [Tz13], [Tz14], [Yo14], [Ce14]
G3 **Cross-stratum orchestration schemes**		Frameworks [ONEBULA], [OSTACK]
		Architecture and protocols [Co12], [OPENFLOW], [RFC7491], [Dh12], [Ku13]
		EONs [ZhZh13], [Ja13]

Chapter 4

Elastic operations in DC federations

The huge energy consumption of DCs providing cloud services over the Internet has motivated different studies regarding cost savings in DCs. Since energy expenditure is a predominant part of the total OPEX for DC operators, energy aware policies for minimizing DCs' energy consumption try to minimize energy costs while guaranteeing a certain QoE to end-users. Federated DCs can take advantage of its geographically distributed infrastructure by managing appropriately the green energy resources available in each DC at a given time, in combination with workload consolidation and VM migration policies. In this scenario, inter-DC networks play an important role and communications cost must be considered when minimizing OPEX.

In this chapter, we study the influence of the interconnection network on the elastic operations in DC federations. We tackle the Elastic Operations in Federated DCs for Performance and Cost Optimization (ELFADO) problem for scheduling workload orchestrating federated DCs. Two approaches, *distributed* and *centralized*, are studied and MILP formulations and heuristics are provided. Using those heuristics, cost savings with respect to a fixed workload placement are analyzed. For the sake of a compelling analysis, exhaustive simulation experiments are carried out considering realistic scenarios. Results show remarkable energy cost savings when the centralized ELFADO approach is considered.

4.1 Elastic Operations in Federated DCs for Performance and Cost Optimization (ELFADO) problem

Aiming at reducing energy expenditure, DC operators can use, or even generate themselves, *green* energy coming from solar or wind sources; green energy would replace either partially or totally energy coming from *brown*, polluting sources. The drawback is that green energy is not always available, depending on the hour of the day, weather and season, among others. In contrast, brown energy can be drawn from the grid at any time, although its cost might vary along the day.

We assume a set of federated DCs strategically placed around the globe, so as to provide worldwide high QoE services interconnected by a flexgrid-based network. Each DC has access to some amount of energy coming from green sources (solar energy), which can cover a fraction of total energy consumption (*green coverage*), being the rest drawn from the grid. We study two approaches to orchestrate such DC federation to provide committed QoE while minimizing OPEX: distributed and centralized. In the distributed approach, DCs schedule VM placement so as to minimize an estimation of the energy cost plus communication costs while ensuring QoE. In the centralized approach, a centralized orchestrator computes the global optima from placing VM to take full advantage from green energy availability in the federated DCs.

A first optimization to reduce energy expenditures is to perform consolidation, placing VMs so as to load servers as much as possible and switching off those servers that become unused. To further reduce energy consumption, consolidation can be performed by taking into account clusters structure, and switching on/off clusters as single units. Those servers in switched on clusters without assigned load remain active and ready to accommodate spikes in demand.

In addition, DC federations can perform elastic operations, migrating VMs among DCs aiming at minimizing operational costs by taking advantage of available green energy in some DCs and off-peak cheap brown energy in others DCs while ensuring the desired QoE level. We use latency experienced by the users of a service as a measure of QoE level.

We face then the ELFADO problem, which orchestrates federated DCs providing optimal VMs placement so as to minimize operational costs. We assume that operational costs are dominated by energy and communications costs, so we focus on specifically minimizing those costs.

Two approaches can be devised to orchestrate federated DCs (Fig. 4-1):

- Distributed (Fig. 4-1a), where scheduling algorithms running inside DC resource managers compute periodically the optimal placement for the VMs currently placed in the local DC.

- Centralized (Fig. 4-1b), where a *federation orchestrator* computes periodically the global optimal placement for all the VMs in the federated DCs and communicates that computation to each DC resource manager.

In both approaches, local resource managers interface the rest of DCs to coordinate VM migration and an intelligent control plane controlling the interconnection network to request optical DC2DC connections' set-up and tear down.

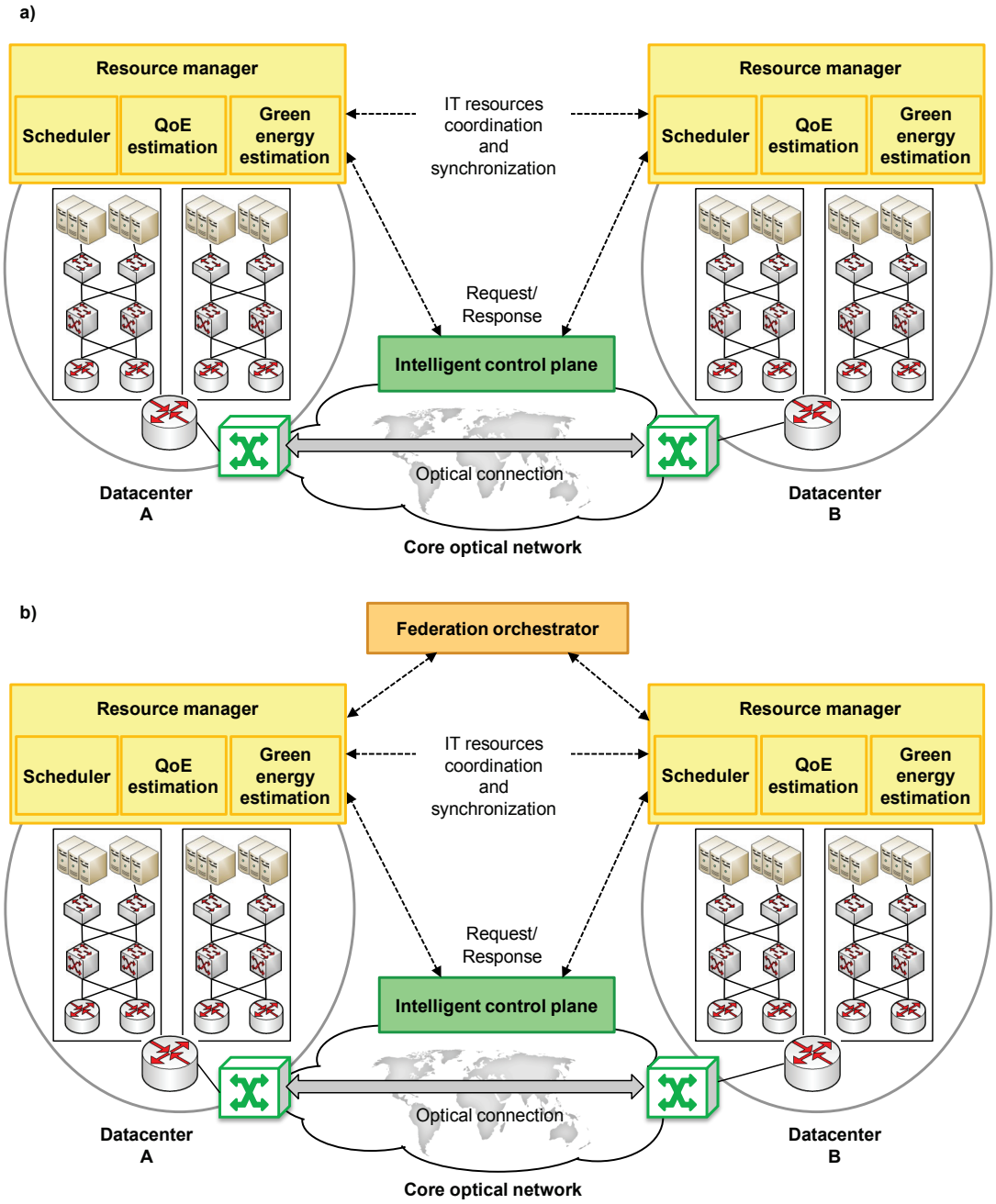

Fig. 4-1 Distributed (a) and centralized (b) federated DCs orchestration.

To solve the ELFADO problem some data must be available, such as an estimation of QoE perceived by the users, the amount of green energy available in each DC and the cost of brown energy, among others. QoE can be estimated by a specialized module inside each resource manager [VERIZON]. The cost of brown energy comes from the contract each DC has with the local power supply company, which varies with the time of day. Finally, the amount of green energy that will be likely available in the next period can be predicted using historical data and weather forecast [Sh10]. Each local resource manager can flood all that data to the rest of resource managers in remote DCs.

For illustrative purposes, Fig. 4-2 plots unit brown energy cost, c_d, and normalized availability of green energy, δ_d, for DC d as a function of the time of day. Brown energy cost varies with the time showing on-peak and off-peak periods, where energy during on-peak is approximately 40% more expensive than during off-peak periods. Regarding green energy availability, large variations during the day can be expected. In the view of Fig. 4-2, it is clear that some advantage can be taken from orchestrating the federated DCs, moving VMs to place them in the most advantageous DC.

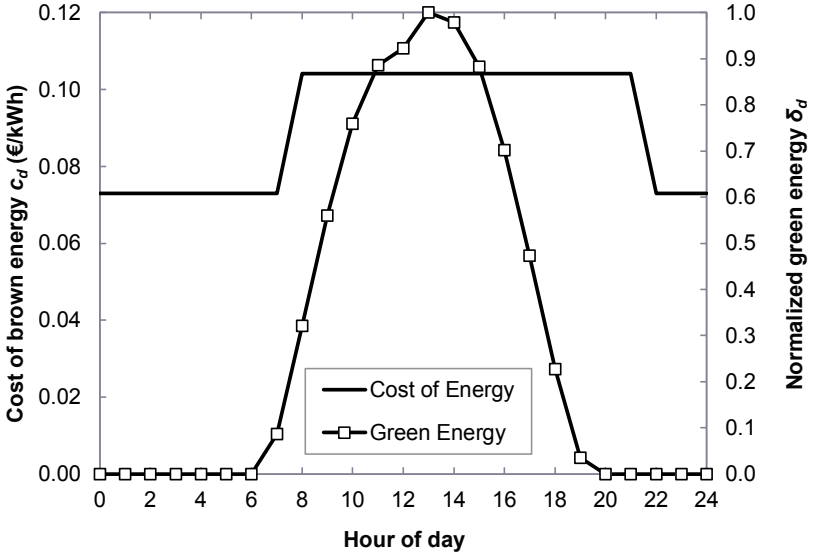

Fig. 4-2 Unit cost of brown energy and normalized availability of green energy against the time of day.

Let us assume that DCs are dimensioned to cover some proportion β_d of the total energy consumption for the maximum dimensioning. Then, green coverage in DC d, a_d, can be estimated as, $a_d(t) = \beta_d \cdot \delta_d(t)$, and the amount of green energy available can be estimated as $g_d(t) = a_d(t) \cdot Energy\_MaxDimensioning$, where *Energy_MaxDimensioning* represents the amount of energy consumed for the maximum dimensioning.

In the distributed approach, local DCs do not know the amount of VMs that will be placed in each DC in the next period, since that decision is to be taken by each DC resource manager in the current period. Therefore, the amount of VMs that can take advantage of green energy availability in each DC in the next period cannot be computed. To overcome that problem, estimation on the unitary energy cost in each DC should be made. We use equation (4.1), i.e., the cost of the energy in each DC ($\hat{c}_d$) is estimated by decrementing the cost of brown energy with the expected green coverage value. As an example, the estimated cost of the energy is 0.0729 €/kWh at 2 a.m. and 0 €/kWh at 1 p.m. (assuming $\beta_d = 1$).

$$\hat{c}_d = (1 - \alpha_d) \cdot c_d \tag{4.1}$$

In general, however, green energy covers only partially, even in the generation peak, total energy consumption, thus $\beta_d < 1$. Therefore, if several DCs take the decision of migrating local VMs to one remote DC in the hope of reducing costs, it may happen that some brown energy need to be drawn from the grid if not enough green energy is available, which may result in higher energy cost in addition to some communication cost.

In contrast, the amount of VMs to be placed in each DC in the next period is known in the centralized approach since the placing decision is taken in the centralized federation orchestrator. Therefore, one can expect that better VM placements can be done in the centralized approach, which might result in further cost savings.

Next section formally states the ELFADO problem and presents MILP models and heuristic algorithms for solving efficiently both the distributed and the centralized approaches.

4.2 Mathematical formulation

4.2.1 ELFADO Problem statement

The ELFADO problem can be formally stated as follows:

Given:

- a set of federated DCs D,
- the set of optical connections E that can be established between pairs of DCs,
- a set of VMs $V(d)$ in each DC d,
- a set of client locations L, where n_l is the number of users in location l to be served in the next period,

- PUE_d, brown energy cost c_d, and green coverage level a_d in DC d for the next period,
- the data volume k_v and the number of cores $cores_v$ of each VM v,
- energy consumption of each server as a function of the load k, $w_{server}(k)=P_{server}(k)\cdot 1\text{h}$,
- the performance p_{ld} perceived in location l when served from a VM placed in DC d,
- a threshold th_v for the performance required at any time for accessing the service in VM v.

Output: DCs where each VM will be placed the next time period.

Objective: minimize energy and communications cost for the next time period ensuring the performance objective for each service.

As previously stated, the ELFADO problem can be solved assuming either a distributed or centralized approach. In the subsequent subsections, we present mathematical programming formulations for each of the approaches. In addition, in view of the stringent time requirements, their exact solving becomes impractical and, as a result, heuristic algorithms are needed so as to provide good near optimal solutions in the time periods required for on-line DC operation.

4.2.2 Mathematical model

The following sets and parameters have been defined:

Sets:

D	Set of federated DCs, index d.
E	Set of optical connections that can be established, index e.
$E(d_1)$	Set of optical connections between d_1 and any other DC.
V	Set of VMs, index v.
$V(d_1)$	Set of VMs in DC d_1.
L	Set of client locations, index l.

Users and performance:

p_{ld}	Performance perceived in location l when accessing DC d.
n_l	Number of users in location l.
th_v	The threshold performance to be guaranteed for v.

DC architecture and VMs:

M Maximum number of clusters per DC.

n_{server} Number of cores per server.

k_v Size in bytes of VM v.

n_v Number of cores needed by VM v.

Energy:

α_d Green energy cover in DC d.

g_d Amount of green energy available in DC d.

PUE_d PUE for DC d.

w_v Energy consumption of VM v. It can be computed assuming that the server where it is placed is fully loaded, so $w_v = w_{server\_max}/n_v$.

c_d Brown energy cost per kWh in DC d.

Connections:

c_e Cost per Gb transmitted through connection e.

k_e Maximum amount of bytes to transfer without exceeding the maximum capacity assigned in connection e. k_e includes the needed overhead from TCP/IP downwards to the optical domain.

Additionally, the decision variables are:

x_{vd} Binary, 1 if VM v is placed in DC d; 0 otherwise.

y_d Real positive, energy consumption in DC d.

z_e Integer positive, bytes to transfer through optical connection e.

The MILP formulation for the ELFADO problem assuming the distributed approach is as follows. It is worth highlighting that this problem is solved by each of the DCs separately; in the model, d_1 identifies the local DC.

$$\left(\textit{Distributed ELFADO}\right) \quad \text{minimize} \quad \sum_{d \in D} \left(1-\alpha_d\right) \cdot c_d \cdot y_d + \sum_{e \in E(d_1)} 8 \cdot c_e \cdot z_e \tag{4.2}$$

subject to:

$$\frac{1}{\sum_{l \in L} n_l} \cdot \sum_{l \in L} \sum_{d \in D} n_l \cdot p_{ld} \cdot x_{vd} \leq th_v \quad \forall v \in V(d_1) \tag{4.3}$$

$$\sum_{d \in D} x_{vd} = 1 \quad \forall v \in V(d_1) \tag{4.4}$$

$$y_d = PUE_d \cdot \sum_{v \in V(d_1)} w_v \cdot x_{vd} \quad \forall d \in D \tag{4.5}$$

$$z_{e=(d_1,d_2)} = \sum_{v \in V(d_1)} k_v \cdot x_{vd_2} \quad \forall d_2 \in D \setminus \{d_1\} \tag{4.6}$$

$$z_e \leq k_e \quad \forall e \in E(d_1) \tag{4.7}$$

The objective function in (4.2) minimizes the total cost for the VMs in a given DC d_1, which consists on the estimated energy costs resulting from the computed placement of those VMs plus the communication costs for the VMs that are moved to remote DCs.

Constraint (4.3) guarantees that each VM is assigned to a DC if the on-average performance perceived by the users is above the given threshold. Constraint (4.4) ensures that each VM is assigned to one DC. Constraint (4.5) computes the energy consumption in each DC as a result of moving VM from the local DC. Constraint (4.6) computes the amount of data to be transfer from the local to each remote DC. Finally, constraint (4.7) assures that the capacity of each optical connection from the local DC is not exceeded.

The MILP formulation for the centralized one is presented next. Although the model is similar to the distributed approach, this problem computes a global solution for all the DCs and as a result, the total amount of VMs that will be placed in the next period in each DC can be computed. Therefore, the centralized ELFADO computes the cost of the energy in each DC given the amount of green energy available.

Two additional decision variables are defined:

γ_d Positive integer with the number of servers operating with some load in DC d.

ρ_d Positive integer with the number of clusters switched on in DC d.

$$(Centralized\ ELFADO) \quad \text{minimize} \quad \sum_{d \in D} c_d \cdot y_d + \sum_{e \in E} 8 \cdot c_e \cdot z_e \tag{4.8}$$

subject to:

$$\frac{1}{\sum_{l \in L} n_l} \cdot \sum_{l \in L} \sum_{d \in D} n_l \cdot p_{ld} \cdot x_{vd} \leq th_v \quad \forall v \in V \tag{4.9}$$

$$\sum_{d \in D} x_{vd} = 1 \quad \forall v \in V \tag{4.10}$$

$$\gamma_d \geq \frac{1}{n_{server}} \cdot \sum_{v \in V} n_v \cdot x_{vd} \quad \forall d \in D \tag{4.11}$$

$$\rho_d \geq \frac{4}{M^2} \cdot \gamma_d \quad \forall d \in D \tag{4.12}$$

$$y_d \geq PUE_d \cdot \left(\begin{array}{c} \frac{M^2}{4} \cdot w_{core} + \frac{M}{2} \cdot \left(w_{agg} + w_{edge} \right) \cdot \rho_d + \\ w_{server-\max} \cdot \gamma_d + w_{server-idle} \cdot \left(\frac{M^2}{4} \cdot \rho_d - \gamma_d \right) \end{array} \right) - g_d \quad \forall d \in D \tag{4.13}$$

$$z_{e=(d_1,d_2)} = \sum_{v \in V(d_1)} k_v \cdot x_{vd2} \quad \forall d_1, d_2 \in D, d_1 \neq d_2 \tag{4.14}$$

$$z_e \leq k_e \quad \forall e \in E \tag{4.15}$$

The objective function (4.8) minimizes the total cost for all DCs in the federation, which consists on the energy costs plus the communication costs for the VMs that are moved between DCs.

Constraint (4.9) guarantees that each VM is assigned to a DC if the on-average performance perceived by the users is above the given threshold. Constraint (4.10) ensures that each VM is assigned to one DC. Constraint (4.11) computes, for each DC, the amount of servers where some VM is going to be placed, whereas (4.12) computes the number of clusters that will be switched on. Constraint (4.13) computes the brown energy consumption in each DC as the difference between the effective energy consumption, computed as equations (2.1) and (2.2), and the amount of green energy available in the next period in each DC. Note that $w_{(\cdot)}=P_{(\cdot)} \cdot 1\text{h}$. Constraint (4.14) computes the amount of data to be transfer from each DC to some other remote DC. Finally, constraint (4.15) assures that the capacity of each optical connection is not exceeded.

4.2.3 Complexity analysis

The ELFADO problem is *NP-hard* since it is based on the well-known capacitated plant location problem, which has been proved to be *NP-hard* [Di02]. Regarding problem sizes, the number of variables and constraints for each approach are detailed in Table 4-1. Additionally, an estimation of problems' size is calculated for the scenario presented in 4.4.1.

Table 4-1 Size of the ELFADO problem

	Constraints	**Variables**
Distributed	$O(\|V\|+\|D\|)$ (10^4)	$O(\|V\| \cdot \|D\|)$ (10^5)
Centralized	$O(\|V\|+\|D\|^2)$ (10^5)	$O(\|V\| \cdot \|D\|)$ (10^5)

Although the size of the MILP models is limited, the ELFADO problem must be solved in short times in realistic scenarios (in the order of few seconds). In our experiments described in Section 4.5, we used commercial solvers such as CPLEX to solve each approach. The distributed approach took tens of minutes on average to be solved; more than 1 hour in the worst case, whereas the centralized approach took more than one hour on average. As a consequence, in the next section we propose heuristic algorithms that provide much better trade-off between optimality and complexity to produce solutions in practical computations times, short enough to be used for schedule real federated DCs.

4.3 Heuristic algorithms

The heuristic algorithm for the distributed approach (Table 4-2) schedules the set of VMs in the local DC. For each VM, all feasible, in terms of performance (p_{vd}), placements are found and the cost for that placement is computed (lines 2-9). If the placement is in the local DC, only energy costs are considered, whereas if it is in a remote DC communication costs are also included. Note that energy costs are estimated using the green energy cover to decrement the cost of the energy in the considered DC. The list of feasible placements is ordered as a function of the cost (line 10). Each VM is placed afterwards in the cheapest DC provided that the amount of data to be transferred through the optical connection does not exceed the maximum available, in case of a remote placement (lines 11-17). The final solution is eventually returned (line 18).

The heuristic algorithm for the centralized approach (Table 4-3) schedules the set of VMs in all the federated DCs. The proposed heuristic focuses on taking advantage of all the available green energy, only considering the cost of brown energy and communications when no more green energy is available. The perceived performance of each VM in its current placement is computed; those infeasible placements (when the perceived performance is under the threshold) are added to set U whereas those which are feasible are added to set F (lines 2-7). Next, the remaining green energy in each DC is computed, considering the available green energy and the energy consumption of those feasible placements (line 8). The set R stores those DCs with remaining green energy available.

Table 4-2 Heuristic for the distributed ELFADO

```
INPUT: d1, V(d1), D
OUTPUT: Sol
 1:  Sol ← ∅
 2:  for each v ∈ V(d1) do
 3:    for each d ∈ D do
 4:      if p_vd ≤ th_v then
 5:        if d ≠ d1 then
 6:          let e = (d1, d)
 7:          C[v].list ← {d, e, (1-α_d)·c_d·w_v + c_e·k_v }
 8:        else
 9:          C[v].list ← {d, ∅, (1-α_d)·c_d·w_v}
10:    sort(C[v].list, Ascending)
11:  for each v ∈ V(d1) do
12:      for i = 1..C[v].list.length do
13:        {d, e} ← C[v].list(i)
14:        if e ≠ ∅ && z_e + k_v > k_e then continue
15:        if e ≠ ∅ then z_e ← z_e + k_v
16:        Sol ← Sol ∪ {(v, d)}
17:        break
18:  return Sol
```

The remaining green energy in the DCs (if any) is used to place infeasible placements in set *U*; the cheapest feasible placement if found for each VM in *U* provided that the energy consumption of that VM can take advantage of remaining green energy (lines 12-15). If a feasible placement is finally found, the remaining green energy for the selected DC is updated (line 16) and if no green energy remains available, that DC is eventually removed from set *R*. The same process of maximizing available green energy is performed for the feasible placements in set *F* (lines 19-25).

Every remaining not yet considered, feasible or unfeasible, placement is stored in set *F* to be jointly considered (line 26) and an algorithm similar to the one for the distributed approach is then followed (lines 27-42). The only difference is that the cost of new placements is computed considering that all the energy will come from brown sources (lines 32 and 34). Finally, the solution for all the DCs is returned.

The performance of each of the proposed heuristic algorithms was compared against the corresponding MILP model. In all the experiments performed, the heuristics were able to provide a much better trade-off between optimality and computation time; in all tests the optimal solution was found within running times of hundreds of milliseconds, in contrast to tens of minutes (for the distributed) and even hours (for the centralized) needed to find the optimal solution with the MILP models. Thus, we use the heuristics to solve the instances in the scenario presented in the next section.

Table 4-3 Heuristic for the centralized ELFADO

INPUT: V, D
OUTPUT: Sol

```
 1: Initialize Sol ← ∅; U ← ∅; F ← ∅; R ← ∅
 2: for each d ∈ D do
 3:   U_d ← ∅; F_d ← ∅
 4:   for each v ∈ V(d) do
 5:     if p_vd > th_v then
 6:       U_d ← U_d ∪ {(v, d)}
 7:     else F_d ← F_d ∪ {(v, d)}
 8:   r_d ← g_d - computeEnergy(F_d)
 9:   U ← U ∪ U_d; F ← F ∪ F_d
10:   if r_d > 0 then
11:       R ← {(d, r_d)}
12: if R ≠ ∅ then
13:   for each (v, d_1) ∈ U do
14:     find (d_2, r_d2) ∈ R feasible for v such that r_d > w_v
        with min comm. cost
15:     Sol ← Sol ∪ {(v, d_2)}
16:     r_d2 ← r_d2 - PUE_d2·w_v
17:     if r_d2 <= 0 then
18:       R ← R \ {(d_2, r_d2)}
19: if R ≠ ∅ then
20:   for each (v, d_1) ∈ F do
21:     find (d_2, r_d2) ∈ R feasible for v such that r_d > w_v
        with min comm. cost
22:     Sol ← Sol ∪ {(v, d_2)}
23:     r_d2 ← r_d2 - PUE_d2·w_v
24:     if r_d2 <= 0 then
25:       R ← R \ {(d_2, r_d2)}
26: F ← F ∪ U
27: for each {v, d_1} ∈ F do
28:   for each d_2 ∈ D do
29:     if p_vd2 ≤ th_v then
30:       if d_2 ≠ d_1 then
31:         let e = (d_1, d_2)
32:         C[v].list ← (d_2, e, c_d2·w_v + c_e·k_v)
33:       else
34:         C[v].list ← (d_2, e, c_d2·w_v)
35:   sort(C[v].list, Ascending)
36: for each (v, d_1) ∈ F do
37:   for i = 1..C[v].list.length do
38:     (d_2, e) ← C[v].list(i)
39:     if e ≠ ∅ && z_e + k_v > k_e then continue
40:     if e ≠ ∅ then z_e ← z_e + k_v
41:     Sol ← Sol ∪ {(v, d_2)}
42:     break
43: return Sol
```

4.4 Performance evaluation

In this section, we present a scenario considered in our experiments and we show the results of solving the ELFADO problem considering a realistic instance; we evaluate the impact in the cost when distributed and centralized approaches are used for scheduling VM placement compared to a fixed placement, where no scheduling is done.

4.4.1 Scenario

We implemented the proposed heuristic algorithms for the distributed and centralized ELFADO approaches on a scheduler in the OpenNebula cloud management middleware. For comparison, a *fixed* approach, where the total workload is evenly distributed among the federated DCs, was also implemented.

We consider a global 11-location topology depicted in Fig. 4-3. Each location collects user traffic towards the set of federated DCs, which consists of five DCs strategically located in Taiwan, India, Spain, and Illinois and California in the USA. A global telecom operator provides optical connectivity among DCs, which is based upon the flexgrid technology. The number of users in each location was computed considering Wikipedia's audience by regions [METAWIKI] that was scaled and distributed among the different locations in each region. Latency between location pairs was computed according to [VERIZON].

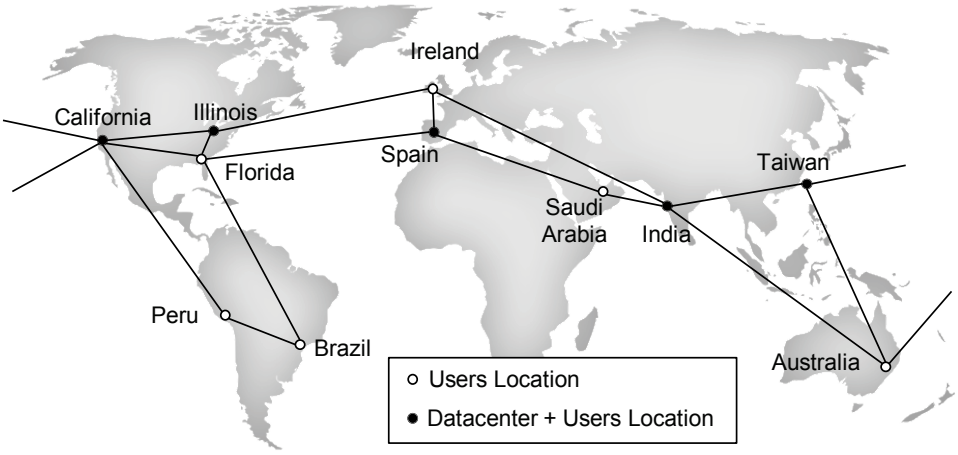

Fig. 4-3 Considered scenario: federated DCs, locations and inter-DC network.

Table 4-4 briefly presents the value considered for some representative energy parameters. Daily PUE values were computed according to [Go12] using data obtained from [USEIA]. Green energy coverage was obtained from [USEIA], [USDOE] and [Ki04] and brown energy cost for each DC was estimated from their respective local electric company rates (e.g., [EEP], [USAEP]). Servers in DCs are

assumed to be HP ProLiant DL580 G3, equipped with four processors, 2 cores per processor, with $P_{server\text{-}idle}$ = 520W and $P_{server\text{-}max}$ = 833W.

Table 4-4 Value of energy parameters

DC	c_d (on/off peak) (€/kWh)	β_d	PUE (max/ avg)
Taiwan	0.0700 / 0.0490	0.5	1.671 / 1.632
India	0.0774 / 0.0542	0.9	1.694 / 1.694
Spain	0.1042 / 0.0729	0.9	1.670 / 1.457
Illinois	0.0735 / 0.0515	0.2	1.512 / 1.368
California	0.0988 / 0.0692	0.5	1.385 / 1.303

In line with [Fa08], DCs are dimensioned assuming a fat-tree topology with a maximum of M = 48 clusters with two levels of switches and $M^2/4$ = 576 servers each. The number of VMs was set to 35,000, with individual image size of 5 GB; we assume that each VM runs in one single core. An integer number of clusters is always switched on, so as to support the load assigned to the DC; those servers without assigned load remain active and ready to accommodate spikes in demand. Green cover was set to ensure, at the highest green energy generation time, a proportion of energy β_d when all VMs run in DC d.

We consider a different type of switches, and thus a different power consumption value, for each layer of the intra-DC architecture. We selected the Huawei CloudEngine switches series; Table 4-5 specifies model, switching capacity, and power consumption for each considered model.

Table 4-5 Characteristics of Huawei CloudEngine switches

Layer	Model	Switching capacity	Power consumption
Core	12812	48 Tb/s	P_{core} = 16,200 W
Aggregation	6800	1.28 Tb/s	P_{agg} = 270 W
Edge	5800	336 Gb/s	P_{edge} = 150 W

Finally, we consider that each DC is connected to the flexgrid inter-DC network through a router equipped with 100 Gb/s BVTs. Therefore, the actual capacity of optical connections is limited to that value. To compute the real throughput, we consider headers for the different protocols, i.e., TCP, IP, and GbE. The maximum

amount of bytes to transfer, k_e, was computed to guarantee that VM migration is performed in less than 40 minutes.

4.4.2 Illustrative results

Fig. 4-4 (top) displays the availability of green energy as a function of the time (GMT) at each DC, $a_d(t)$, for a typical spring day, whereas the two bottommost graphs in Fig. 4-4 illustrate the behavior of the distributed (middle) and centralized (bottom) ELFADO approaches.

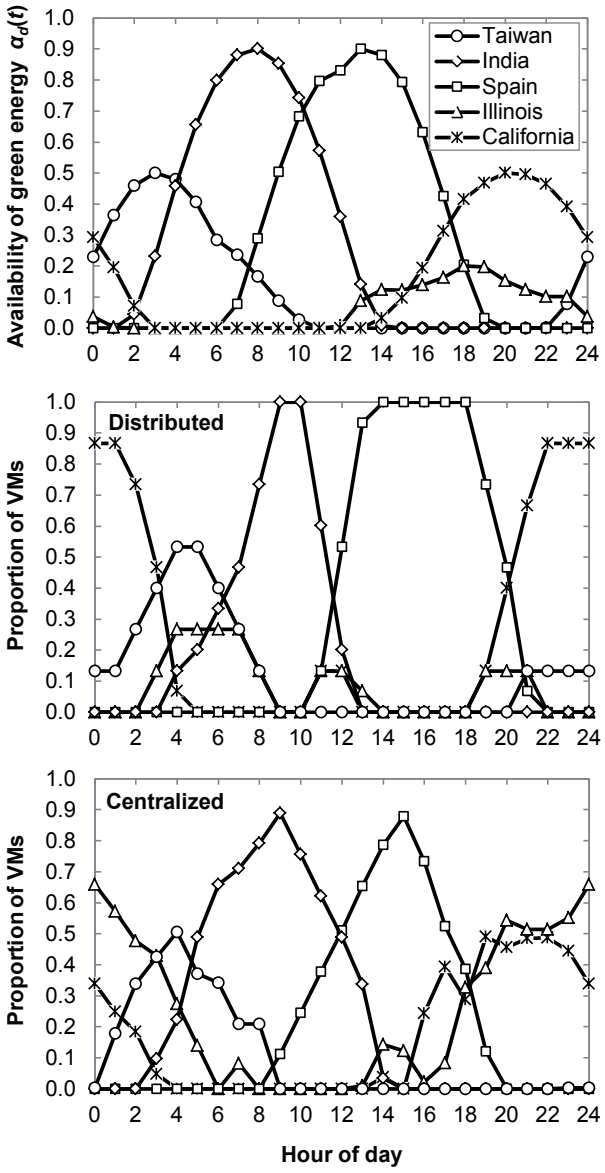

Fig. 4-4 Availability of green energy vs. time in all DCs (top). Percentage of VMs in each DC when the distributed (middle) and the centralized (bottom) approaches are applied.

The distributed approach places VMs in DCs where the cost of energy (plus communications) is expected to be minimum in the next period; equation (4.1) is used for that energy cost estimation. However, in view of Fig. 4-4 (middle), it is clear that equation (4.1) does not provide a clear picture, since all VMs are placed in India and Spain during the day periods where more green energy is available in those locations, thus exceeding green energy availability and paying a higher cost. In contrast, DC in Illinois seems to be underutilized.

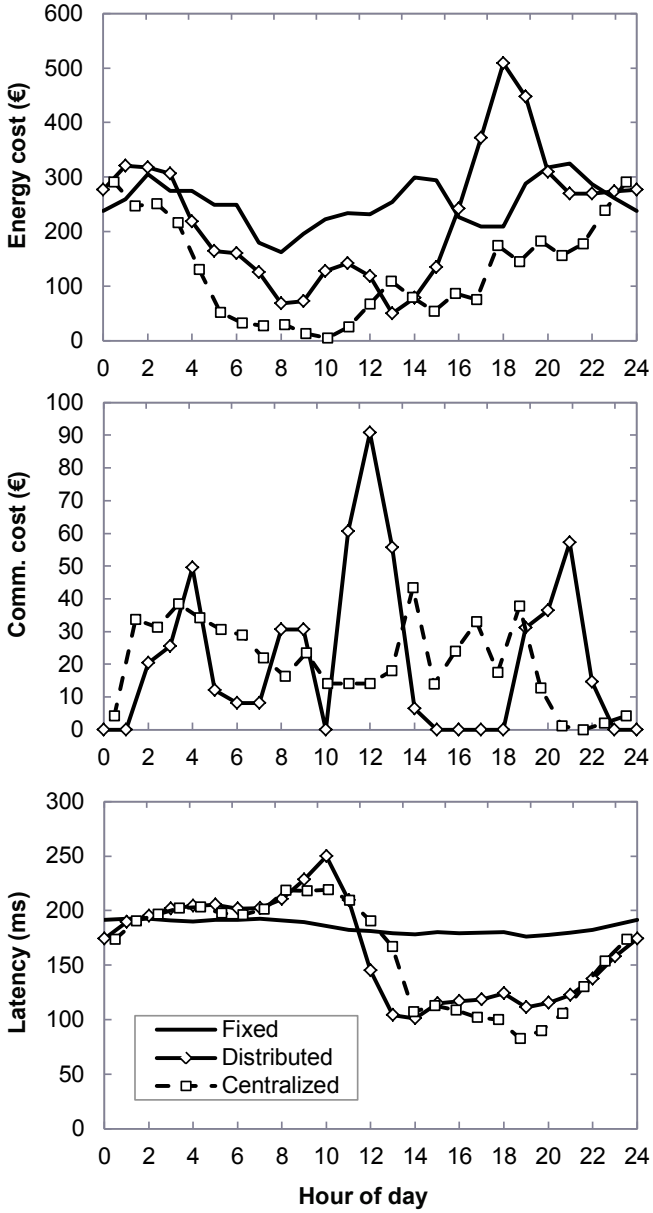

Fig. 4-5 Energy (top) and communication (middle) cost per hour against time. Latency vs. time (bottom).

Interestingly, the centralized approach reduces the percentage of VMs in those DCs with higher green coverage, to place only the amount of VMs (translated into

powered-on clusters and servers) that the available green energy can support and placing the rest considering brown energy (and communications) costs. In fact, the DC in Illinois is more used in the centralized approach as a consequence of its cheaper brown energy cost compared to that of California.

Fig. 4-5 presents costs and performance as a function of the time for all three approaches; cost per transmitted bit was set to 1e-9 €/Gb·km. Energy costs per hour plots in Fig. 4-5 (top) show a remarkable reduction in energy costs when ELFADO (either distributed or centralized) is implemented, with respect to the *fixed* approach. Daily comparison presented in Table 4-6 shows savings of 11% for the distributed and over 52% for the centralized approach. Hourly plot for the distributed approach clearly highlight how by placing VMs in DCs where the estimated energy is cheaper results in a high amount brown energy being drawn from the grid at a more expensive price. In contrast, the centralized approach leverages green energy arriving to virtually zero energy cost in some periods.

Regarding communications (Fig. 4-5 (middle)), the distributed approach shows a more intensive use, presenting three peaks, exactly when the DC in Illinois is used to compensate energy costs between green energy availability peaks in the rest of DCs. However, although the centralized approach is less communications intensive, the total daily communications costs are only under 6% cheaper compared to the distributed approach, as shown in Table 4-6.

Table 4-6 Comparison of daily costs and performance

Approach	Energy cost	Comm. cost	Total cost	Average Latency
Fixed	6,048 €	--	6,048 €	185.2 ms
Distributed	5,374 € (11.1%)	537 €	5,912 € (2.3%)	164.2 ms (11.3%)
Centralized	2,867 € (52.6%)	508 € (5.8%)	3,376 € (44.2%)	161.5 ms (12.8%)

Aggregated daily costs are detailed in Table 4-6 for all three approaches. As shown, the distributed approach saves only 2% of total cost when compared to the *fixed* approach. Although, that percentage represents more than 100 € per day, it is just a small portion of the savings obtained by the centralized approach, which are as high as just over 44% (more than 2.6 k€ per day).

Regarding performance (latency), both the distributed and the centralized approach provide figures more than 10% lower than that of the fixed approach as shown in Table 4-6. Hourly plots presented in Fig. 4-5 (bottom) show that latency is slightly higher during some morning periods under both, the distributed and the centralized ELFADO, with respect to that of the *fixed*; afternoon, however, both approaches reduce latency extraordinary since VMs are placed closer to users.

The results presented in Fig. 4-5 were obtained by fixing the value of th_v to 1.3·average(*latency_fixed*) (specified in Table 4-6), so as to allow obtaining worse hourly performance values in the hope of obtaining better daily ones. Fig. 4-6 (left) gives insight of the sensitivity of costs to the value of that threshold. Fixed costs are also plotted as a reference. Costs in the centralized approach show that even for very restrictive thresholds, noticeable cost savings can be obtained. In addition, when the threshold is set to the average latency in the *fixed* approach or above, obtained costs are almost constant. In contrast, the distributed approach proves to be more sensible to that threshold, reaching a minimum in terms of costs when the threshold value is 30% over the average latency in the *fixed* approach.

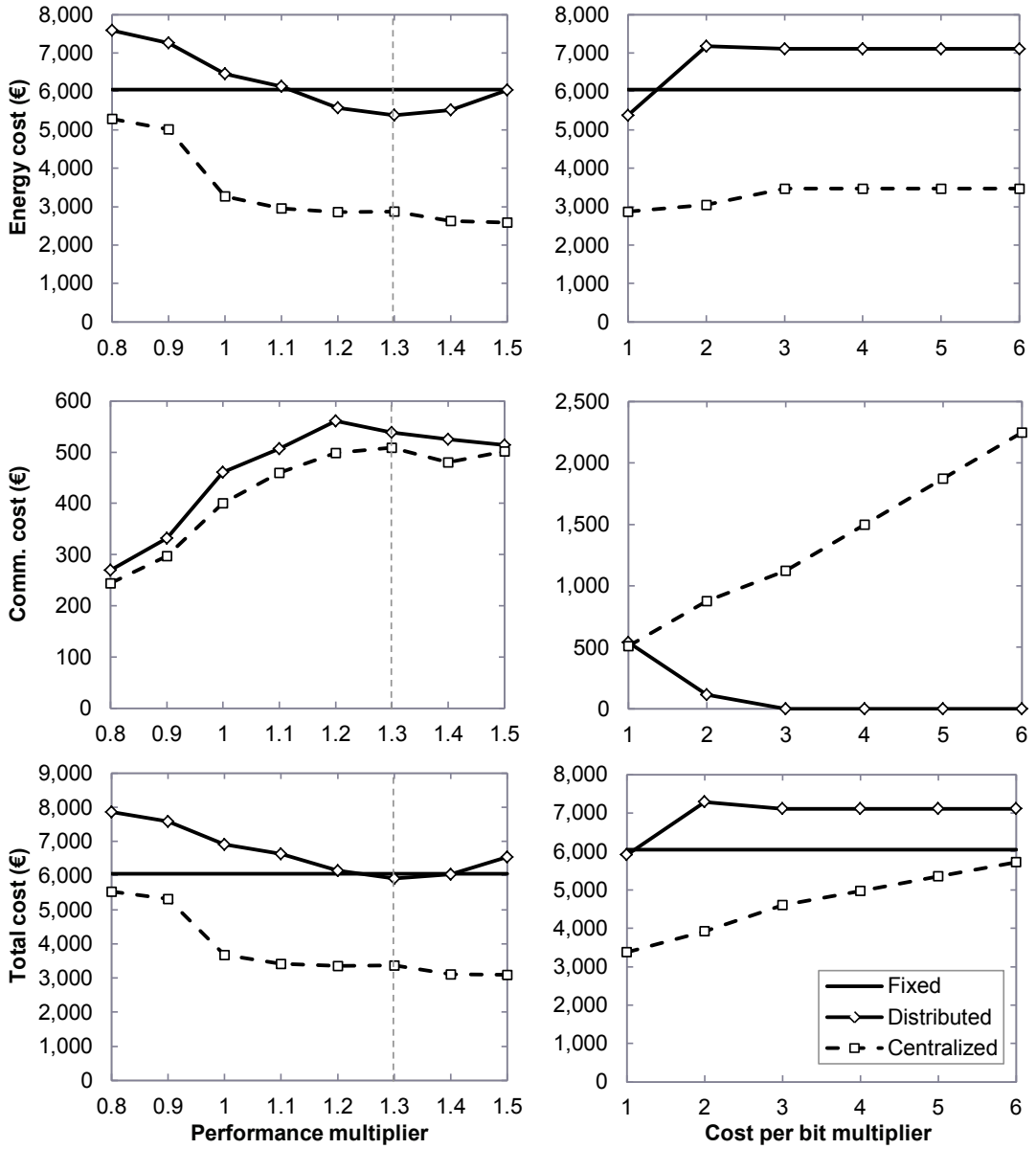

Fig. 4-6 Cost per day against performance threshold (left) and cost per day against cost per bit (right).

Finally, Fig. 4-6 (right) illustrates the influence of the cost per bit to transfer VMs from one DC to another. As before, fixed costs are plotted for reference. Energy costs in the distributed approach increase sharply when the cost per bit doubles, almost preventing from moving VMs, as clearly shown Fig. 4-6 (right, middle). Nonetheless, energy costs are almost stable in the centralized approach. Recall that the proposed heuristic focuses on green energy availability as the first indicator for placing VMs. In fact, communication cost increase linearly with the increment in the cost per bit. However, it is not until the cost per bit increases more than 6 times when the centralized approach cost equals that of the *fixed* approach.

4.5 Conclusions

The enormous energy consumption of DCs translates into high OPEX for DC operators. Although the use of green energy allows reducing energy bills, its availability is reduced depending on the hour of the day, weather and season, etc. Federating DCs can be a way for independent DC operators to, not only increase their revenue, but also reduce OPEX. Aiming at optimizing costs whilst ensuring the desired QoE for users, this chapter described and formally stated the ELFADO problem to orchestrate federated DCs, placing workloads in the most convenient DC.

Two approaches to solve the ELFADO problem were compared, distributed and centralized, where mathematical formulations as well as heuristic algorithms for scheduling VM placement were proposed. The distributed approach is based on running scheduling algorithms inside DC resource managers to compute periodically the optimal placement for the VMs currently in the local DC. VMs are placed in DCs where the cost (energy and communications) is expected to be minimum for the next period. In this approach, energy costs are estimated since the total amount of VMs to be placed in each DC is computed in a distributed manner. Therefore, the available green energy could not be enough to cover the whole energy consumption in each DC. In contrast, the centralized approach, proposes a *federation orchestrator* to compute the global optimal placement for all the VMs in the federated DCs. VMs are placed in DCs so as to take full advantage of green energy availability. This is possible as a result of computing the placement of all VMs in the proposed federation orchestrator at the same time.

The obtained results showed that both ELFADO approaches improve QoE by reducing average latency by more than 10% with respect to a *fixed* approach where no scheduling is performed. Regarding costs, the distributed approach can save up to 11% of costs with respect to the *fixed* approach. However, when communication costs are considered, total cost savings reduce to only 2%. The centralized approach showed remarkable energy cost savings circa 52%, which result in 44% when communication costs are taking into account. Additionally, it was shown that the

centralized approach provides costs savings even when the cost per bit increases 6 times.

As a final conclusion, elastic operations in DC federations require huge capacity and on-demand connectivity from the interconnection networks.

Next chapter continues evaluating the influence of the interconnection network in telecom cloud scenarios.

Chapter 5

Telecom cloud-based services

Several IP-based services may take advantage of the telecom cloud infrastructure in national or regional-wide scale. Although common connectivity requirements may be found, some services on the telecom cloud require service-specific SLA parameters, such as stringent QoS and bitrate guarantees in addition to huge capacity and bandwidth-on-demand.

Among different IP-based services, in this chapter we focus on *i*) live-TV distribution, which could be deployed in telecom cloud infrastructure in the short or the medium term, and *ii*) future 5G networks, which are currently under research. Specifically, a hierarchical distributed telecom cloud architecture for live-TV distribution exploiting flexgrid networking and SBVTs is proposed and its scalability is compared to that of a centralized architecture. Regarding 5G networks, the impact of the C-RAN centralization level on CAPEX and OPEX is studied in telecom cloud-based scenarios, where COs can host virtualized BBU pools. To that end, we present the C-RAN CAPEX minimization (CRAM) problem and an ILP to model it. In both live-TV and C-RAN, results show remarkable network costs savings when a lower level of centralization is considered compared to the maximum centralization level.

Finally, from the studies carried out, connectivity requirements for DC interconnection to support services deployed in the telecom cloud are devised and summarized.

5.1 Live-TV distribution on telecom cloud

Video signal processing is needed to adapt a full-quality signal to the quality that better fits end-users' players. Taking advantage of the cloud infrastructure that

many telecom operators have recently deployed, video signal processing can be performed in commodity hardware inside DCs and distributed directly towards end-users. When the telecom cloud consists of one single large DC, a large number of small flows need to be conveyed to the metro networks, which are commonly used to aggregate users' traffic. On the contrary, if several small DCs are placed closer to end-users to perform video signal processing, uncompressed UHD video signals need to be conveyed from the signal source to each of the small DCs over the core network.

In this section, we study the scalability of both telecom cloud architectures for live-TV signal processing and video distribution. To that end, we assume that the core network is based on the flexgrid technology and that SBVTs as well as fixed transponders (FT) can be installed.

5.1.1 Scalability of telecom cloud architectures

Fig. 5-1 illustrates the centralized and distributed architectures. In the centralized architecture, depicted in Fig. 5-1a, one single large DC receives an uncompressed video stream from the production facilities. Video processing and distribution in the required quality for each end-user is performed in that DC. Each compressed video stream is conveyed over the core network to the metro switch the end-user is connected to. That fact derives in a large number of aggregated flows to be transported from the DC location to different metro segments (represented by Layer 2 (L2) switches in this section). To that end, a large switch needs to be deployed connecting the DC to the flexgrid core network.

In the distributed architecture (Fig. 5-1b), the uncompressed video stream is received in a primary large DC and is forwarded to a secondary DCs placed closer to the end-users, where video processing and distribution to the end-user is performed. Compressed video streams are then aggregated into a single flow and conveyed to the corresponding metro L2 switch, placed in the same location as the secondary DC or in a remote one.

Fig. 5-2 depicts the configuration of the primary DC (a), metro locations (b), and secondary DCs (c). A core switch needs to be installed close to each DC to perform flow switching and aggregation, adapting input flows to the core flexgrid interconnection network. We consider that SBVTs can be installed in those switches to interface OXCs in the core network. On the contrary, FTs need to be installed to interface DCs and metro switches. It is worth noting that to properly dimension a switch, two parameters need to be considered: *i*) switching capacity, and *ii*) number of card slots, each with a number of transponders. It is clear that both core switch dimensioning and number of SBVTs and FTs are different in the two considered architectures. This fact motivates the study in this section.

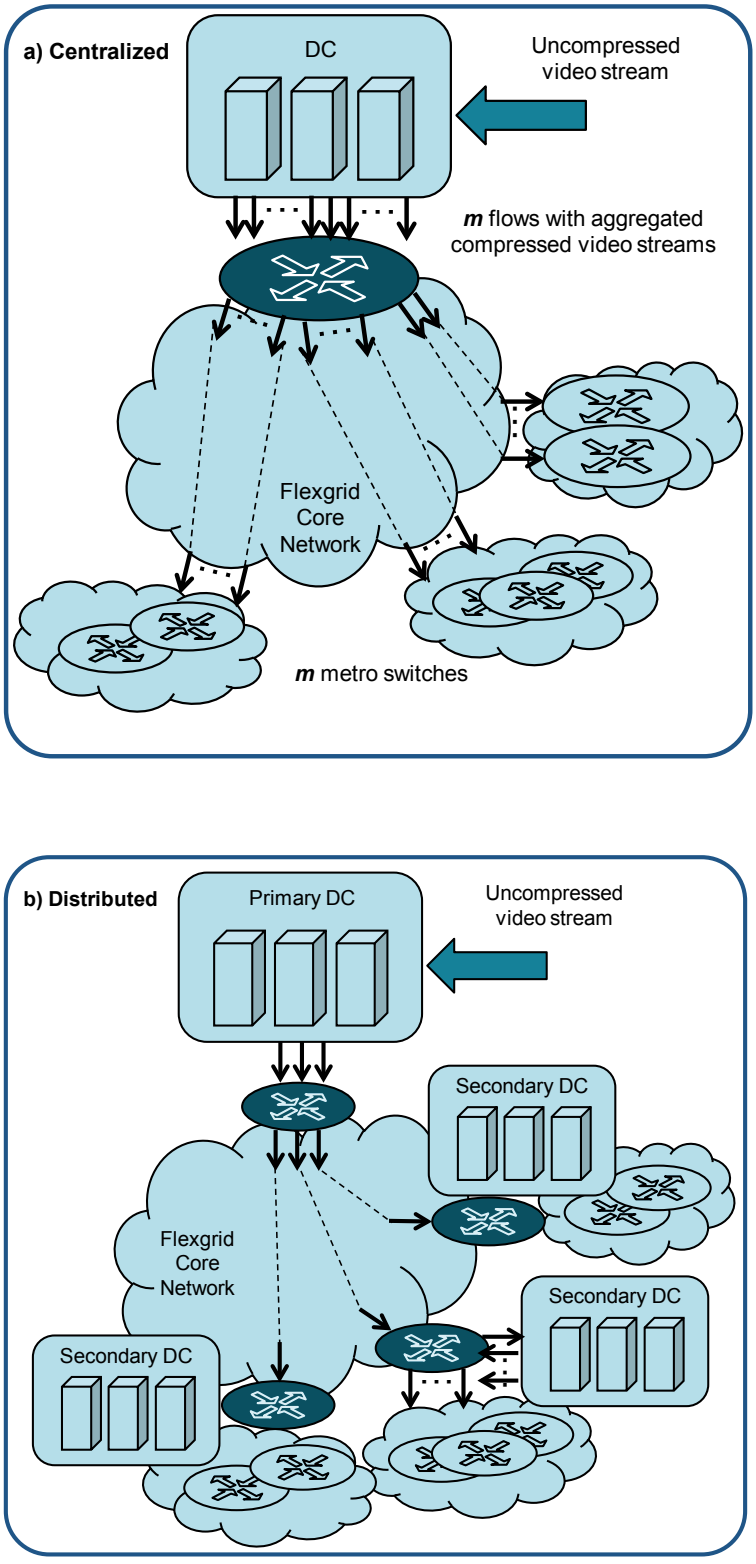

Fig. 5-1 Centralized (a) and distributed (b) architectures.

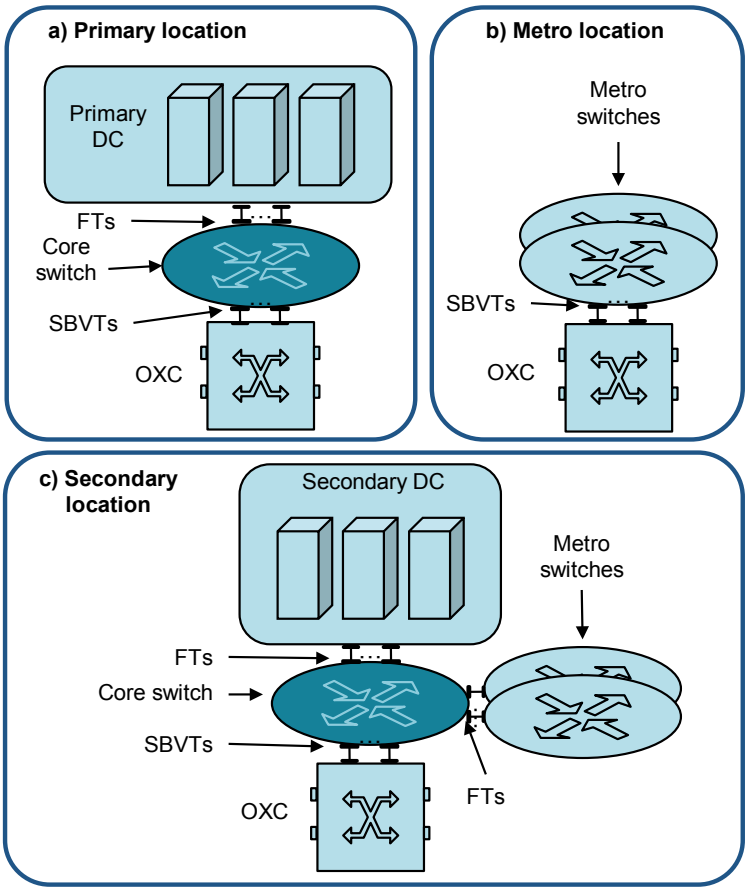

Fig. 5-2 Architecture details.

5.1.2 Planning procedure

To compare the scalability and network CAPEX of both architectures, we propose an optimization problem for the distributed architecture targeting at finding the most cost efficient architecture, given a number n of secondary DCs. The problem can be stated as follows:

Given:

- a primary location p,
- a set M of metro locations each containing a set of metro switches,
- a subset $D \subseteq M$ of locations that can host a secondary DC,
- the topology G of the optical network interconnecting locations in D,
- the cost structure for core switches, FTs, and SBVTs,
- an uncompressed video stream to be distributed from p to every secondary DC (n) and a set of aggregated video streams to be distributed from every secondary DC to the corresponding metro switches.

Output:

- the set $D^* \subseteq D$ of locations, where secondary DCs are placed,
- the configuration of every core switch in terms of capacity, number of card slots, and number and type of transponders to be installed,
- the set of optical connections to be set up in the optical network.

Objective: minimize CAPEX when n secondary DCs are installed.

Aiming at solving this problem, we have developed an iterative procedure, which randomly generates secondary DCs placement (set D') and computes the associated solution by using the constructive algorithm in Table 5-1. Several iterations are performed until some stopping criteria is met (e.g., maximum number of iterations).

The cost computation algorithm firstly finds the optical connections to distribute the uncompressed video signal and the aggregated video streams (lines 1-5). Next, core switches are dimensioned according to the above connections, i.e., the sets of required FTs and SBVTs are computed and a switch with enough switching capacity and number of card slots is installed (lines 6-8). Finally, the total CAPEX is returned.

Table 5-1 Constructive algorithm

```
INPUT: M, D', G, cost structure
OUTPUT: cost
1:  for each d ∈ D' do
2:    Find a connection from p to d on G
3:  for each m ∈ M do
4:    if m ∈ D' then assign all metro switches in m to the
      local secondary DC
5:    else Find a connection from metro switches in m to the
      nearest d∈ D' on G
6:  for each d ∈ D' do
7:    Compute # FTs and SBVTs
8:    Find the cheapest feasible switch
9:  return Compute network CAPEX using cost structure
```

5.1.3 Illustrative results

The scalability and network CAPEX of both architectures have been evaluated using a network topology based on the Telefonica's national network, consisting of 5 regional 30-node optical networks connected through a 21-node core optical network and assuming a realistic number of users (7-8 millions). Adoption scenarios for video-technologies are based on the traffic share for SD, HD, and UHD forecasted in [VNI14]. Table 5-2 summarizes representative values of those scenarios per year considering connection speeds recommended by Netflix for each technology.

Table 5-2 Technology adoption scenarios for different years

		2015	2016	2017	2018
Traffic share	**SD (3 Mb/s)**	28.6%	22.0%	13.6%	7.0%
	HD (5 Mb/s)	65.7%	72.0%	70.5%	71.0%
	4K (25 Mb/s)	5.7%	6.0%	15.9%	22.0%
# users per metro switch		22,900	21,900	19,300	17,400
# metro switches		336	357	399	441

We assume a scenario where 8 TV channels are distributed, i.e., 8 x 12 Gb/s uncompressed 4K UHD TV video streams are conveyed using 100 Gb/s optical connections. In addition, 100 Gb/s connections aggregating SD, HD, and 4K UHD compressed video streams have been considered. Metro switches are connected to OXCs at regional level; the number of switches increases with the video technology adoption scenarios. 4x100 Gb/s and 1x400 Gb/s line-cards and 100 Gb/s FTs and 400 Gb/s SBVTs, capable of sourcing four 100 Gb/s optical connections, are considered to be installed in the core switches; the cost model in [Ra13] for switches and FTs was used, and a SBVT cost 2.5 times that of the 100 Gb/s FTs was assumed, in line with [VeGo14].

Results for the centralized architecture are shown in Table 5-3, where the values of four parameters for the single core L2 switch to be equipped are presented. Both, L2 switch's capacity and number of transponders increase with the increasing number of metro switches required, being the capacity that needs to be installed in the stringent scenario as huge as 89.6 Tb/s.

Table 5-3 Representative values for the centralized architecture

Parameter	2015	2016	2017	2018
Capacity (Tb/s)	70.4	76.8	83.2	89.6
Used cap %	47.7%	46.5%	48.0%	49.2%
# SBVTs	84	90	100	111
# FTs	336	357	399	441

Table 5-4 details the results for the distributed architecture, where the number of secondary DCs, *n*, ranges from 5 to 15. As expected, the capacity of every core L2 switch to be installed in each secondary DC location is much lower than in the centralized architecture, being as low as 6.4 Tb/s. Fig. 5-3 plots the required

capacity of individual switches against the number of secondary DCs considered for each year. It is clear that the capacity of every L2 switch in each secondary DC decreases with n. Regarding transponders, remarkable reductions in the SBVT total number are observed in Table 5-4. To compute the number of transponders, those to be equipped in the switch in the primary location need to be accounted. For instance, $13 \cdot 5 + \lceil 5/4 \rceil = 67$ SBVTs are needed for $n = 5$, whereas $2 \cdot 15 + \lceil 15/4 \rceil = 34$ SBVTs are needed for $n = 15$ in 2015. However, FT total number increases when n increases, as it will be shown later. To compare the centralized and the distributed architectures, we got the solution in the distributed architecture minimizing the total CAPEX; these solutions were for $n = 18$ from 2015 to 2017 and $n = 19$ for 2018. Fig. 5-4a plots the total number of transponders to be equipped in the centralized and the distributed architectures. The amount of FTs is slightly higher in the distributed architecture; the difference ranges from 24 to 34. However, the number of SBVTs to be installed in the centralized architecture is always higher than in the distributed, being the higher difference as remarkable as 87.

Table 5-4 Representative values for the distributed architecture

n	Parameter	2015	2016	2017	2018
5	**Capacity (Tb/s)**	12.8	12.8	19.2	19.2
	Used cap %	53.3%	56.6%	42.1%	46.5%
	# SBVTs	13	14	16	17
	# FTs	68	72	80	89
10	**Capacity (Tb/s)**	6.4	6.4	6.4	12.8
	Used cap %	54.1%	57.3%	63.9%	35.2%
	# SBVTs	5	5	6	6
	# FTs	34	36	40	45
15	**Capacity (Tb/s)**	6.4	6.4	6.4	6.4
	Used cap %	36.6%	38.8%	43.1%	47.5%
	# SBVTs	2	2	2	3
	# FTs	23	24	27	30

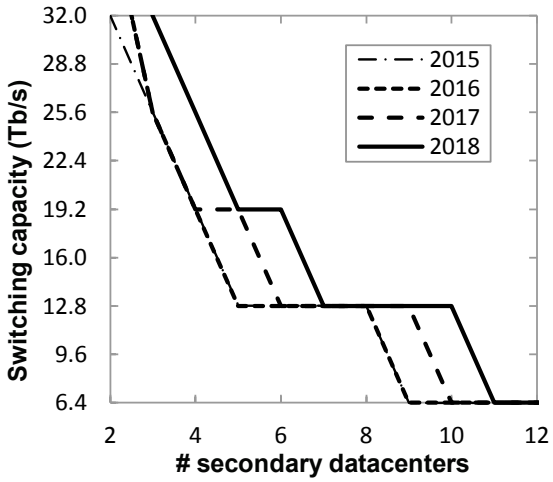

Fig. 5-3 Average switching capacity.

Finally, Fig. 5-4b plots the total CAPEX for both architectures in monetary units (m.u.). Total CAPEX is lower for the distributed architecture under the evaluated scenarios. As soon as HD and 4K UHD streams to be distributed starts representing a significant portion of the total traffic, and thus, more metro switches are required, savings increase up to 32% in the stringent scenario. This is as a result of the combined effect of the increasing switching capacity of the single switch and from the ever increasing number of SBVTs to be installed in the centralized architecture.

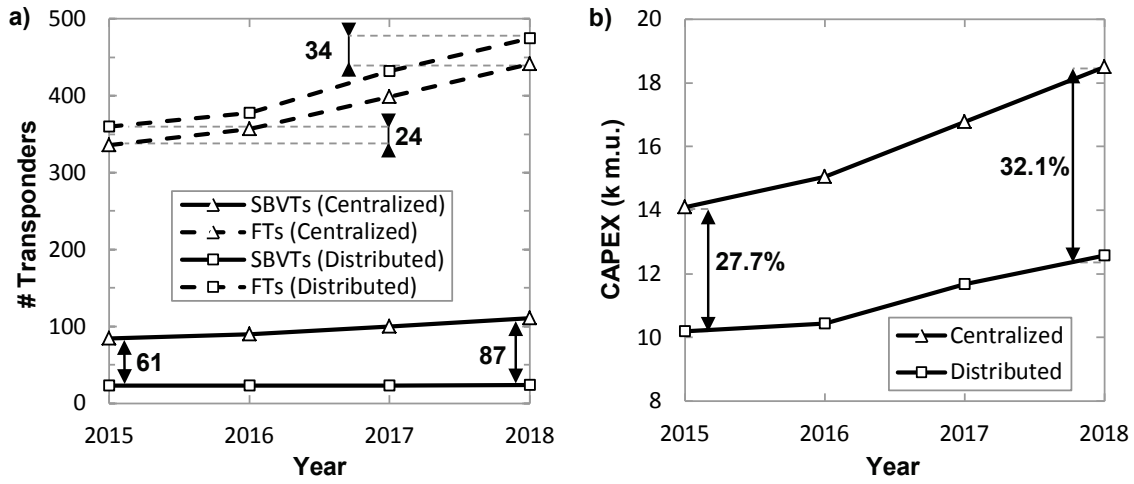

Fig. 5-4 Total number of FTs and SBVTs installed (a) and network CAPEX (b).

5.2 Study of the centralization level of C-RAN

In this section, we propose a mathematical model to study the impact of the centralization level in C-RAN architectures supported by optical networks. To that end, different centralization levels are compared in terms of CAPEX and OPEX. Specifically, the contributions of this section are as follows:

- An ILP model for dimensioning locations hosting virtualized BBU pools (i.e., central offices) to minimize CAPEX in terms of VMs and network equipment, while taking into account the different interfaces needed.
- From the resulting CO design, impact of centralization level is also studied from the OPEX perspective, in terms of network equipment power consumption.

The discussion is supported by representative results from realistic scenarios.

5.2.1 C-RAN architecture model

We consider a reference scenario based on the LTE and LTE-A technologies, where a set of geographically distributed RRHs covers certain regions and virtualized BBU pools are hosted in main COs, whereas the peering point is located in a core CO, which hosts, among others, the Mobility Management Entity (MME) and the Serving Gateway (S-GW); functions that in turn could be virtualized according to [VeCo15].

To provide the required capacity to support load fluctuations in different areas along the day, some of those RRHs can be activated or deactivated. Let us assume that activation (deactivation) of those RRHs can be done through the corresponding entity in charge of the control and management of the C-RAN. RRHs are connected to end-points through fiber links. To support CPRI links, connections from end-points to COs can be effectively implemented and dynamically modified, allowing a given RRH to be assigned to different virtualized BBU pools along the time. Moreover, to support handover and tight coordination schemes, among others, coordination among active and neighboring RRH needs to be considered; thus, X2 interfaces between virtualized BBUs in remote virtualized BBU pools are required. It is worth noting that, due to delay limitations required in X2 interfaces, not all BBUs in virtualized BBU pools in distant COs might be accessible. Finally, S1 links towards the aforementioned core CO (hosting MME and S-GW) need also to be established over the backhaul network; we assume an IP/MPLS-based network.

Fig. 5-5 depicts an example of the reference scenario where a set of RRHs corresponding to macro base stations (MBSs) covers large areas and a set of small cells' RRHs covers smaller areas aiming at offloading cell load during peak hours. The next subsection faces the problem of minimizing CAPEX costs to equip main COs while satisfying demand at any time for all cells; CPRI, S1, and X2 interfaces

requirements and limitations, such as capacity and maximum delay constraints, are considered.

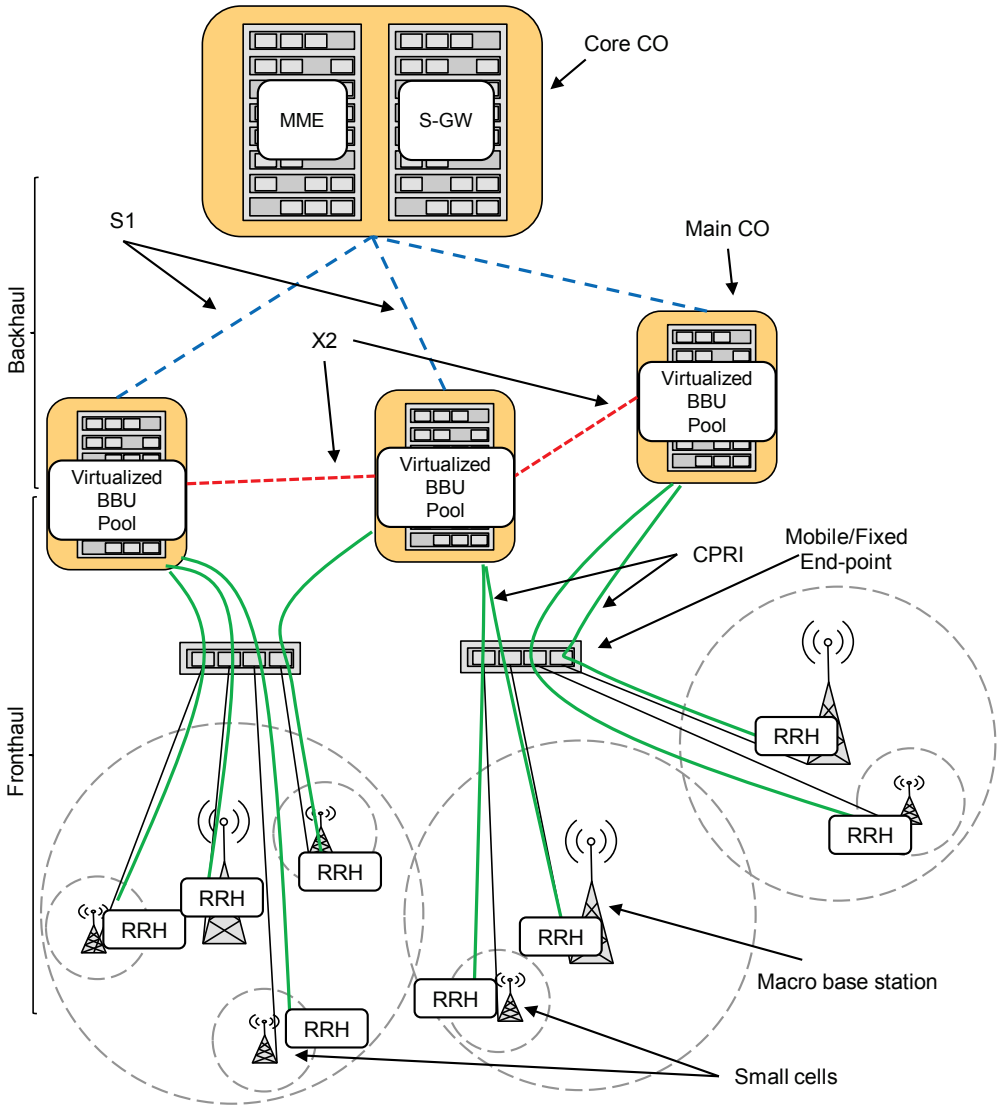

Fig. 5-5 Example of C-RAN architecture.

5.2.2 The C-RAN CAPEX Minimization (CRAM) problem

The CRAM problem can be formally stated as follows:

Given:

- A set of geographically distributed RRHs H; representing $N(h)$ the subset of RRHs neighboring RRH h; i.e., near RRHs operating at the same frequency band and requiring X2 interface links between them to interconnect their respective BBUs, and representing $H(t)$ the subset of H with the RRHs to be activated at time t.

- The tuple $<\alpha_h, \beta_h, \gamma_h>$ representing the required capacity by RRH h for CPRI, S1, and X2 interfaces, respectively, in the case it is active. Since required capacity is constant and depends on the configuration, it can be pre-computed in advance.
- A set V of VMs' configurations with capabilities for BBU pools virtualization; each VM configuration v is defined by its cost κ_v and the number of BBUs it can virtualize λ_v; let us assume that one BBU can serve one RRH.
- A set of transponders P; each transponder p is defined by its cost κ_p and capacity φ_p; since grey or colored transponders may be considered to support different interfaces, the parameters δ_p^{CPRI}, δ_p^{S1}, δ_p^{X2} indicate if p can support CPRI, S1 or X2 interface links, respectively.
- A set of line cards C; each line card c can support one type of transponder and it is defined by its cost κ_c, and the number of ports to plug-in transponders ξ_{cp}.
- A set of IP/MPLS routing equipment E; each router e is defined by its cost κ_e, its switching capacity o_e, and the number of available slots ρ_e to plug-in line cards; the parameter η_{ec} indicates whether equipment e can support card c.
- A set O of main COs; each main CO can be equipped with a predefined configuration of VMs and with an IP/MPLS router.
- $O(h)$ represents the subset of main COs that can be reached by RRH h without exceeding delay imposed by CPRI requirements.
- $U(o)$ represents the subset of main COs that can be reached from main CO o without violating X2 delay constraints.
- A core CO with functions for MME, S-GW, etc.

Output: the VMs' configurations and routing equipment, lines cards and transponders to install in each main CO.

Objective: minimize the cost of VMs' configurations, routing equipment, line cards, and transponders used.

5.2.3 Mathematical formulation

The following sets, subsets, and parameters have been defined:

Sets and subsets:

H	Set of RRHs.
O	Set of main COs.

V Set of VMs' configurations.

E Set of routing equipment.

T Set of day hours.

P Set of transponders.

C Set of line card types.

$H(t)$ Subset of H consisting of RRHs active at time t.

$N(h)$ Subset of H consisting of RRHs neighboring h.

$O(h)$ Subset of O consisting of main COs that can be accessed by RRH h without exceeding the CPRI delay constraint.

$U(o)$ Subset of O consisting of main COs that can be reached from main CO o without exceeding the X2 delay constraint.

Parameters:

λ_v Number of VMs in VMs's configuration v.

a_h Capacity required in CPRI link by RRH h when active.

β_h Capacity required in S1 interface link by RRH h when active.

γ_h Capacity required in X2 interface link by RRH h when active.

φ_p Capacity of transponder p.

δ_p^{CPRI} 1 if transponder p can support CPRI links; 0 otherwise.

δ_p^{S1} 1 if transponder p can support S1 interface links; 0 otherwise.

δ_p^{X2} 1 if transponder p can support X2 interface links; 0 otherwise.

ξ_{cp} Number of ports in line card type c to support transponder p; 0 if line card type c does not support transponder p.

σ_e Available capacity in routing equipment e.

ρ_e Number of available slots in equipment e.

η_{ec} 1 if equipment e can support line card type c; 0 otherwise.

κ_v Cost of VM configuration v.

κ_p Cost of transponder p.

κ_c Cost of line card type c.

κ_e Cost of equipment e.

$bigM$ Large positive constant.

Decision variables:

x_{ov} Binary. 1 if CO o is equipped with VM configuration v; 0 otherwise.

y_{oe} Binary. 1 if CO o is equipped with equipment e; 0 otherwise.

l_{oc} Integer. Number of cards of type c to install in o.

a_{op} Integer. Number of transponders p in CO o.

z_{hot} Binary. 1 if RHH h is assigned to CO o at time t; 0 otherwise.

$w_{hoo't}$ Integer. Number of X2 interface links required between COs o and o' by RRH h at time t.

q_{hotp} Binary. 1 if transponder p is equipped in main CO o to support CPRI links at time t for RRH h; 0 otherwise.

m_{otp} Integer. Number of transponders p to install in main CO o to support S1 interface links at time t.

$n_{oo'tp}$ Integer. Number of transponders p to install in CO o to support X2 interface links at time t to reach CO o'.

The problem can be formulated as follows:

$$\textit{Minimize} \quad \sum_{o\in O}\sum_{v\in V}\kappa_v \cdot x_{ov} + \sum_{o\in O}\sum_{e\in E}\kappa_e \cdot y_{oe} + \sum_{o\in O}\sum_{c\in C}\kappa_c \cdot l_{oc} + \sum_{o\in O}\sum_{p\in P}\kappa_p \cdot a_{op} \tag{5.1}$$

subject to:

$$\sum_{v\in V} x_{ov} \le 1 \quad \forall o \in O \tag{5.2}$$

$$\sum_{e\in E} y_{oe} = \sum_{v\in V} x_{ov} \quad \forall o \in O \tag{5.3}$$

$$\sum_{v\in V} \lambda_v \cdot x_{ov} \ge \sum_{h\in H(t)} z_{hot} \quad \forall t \in T, o \in O \tag{5.4}$$

$$\sum_{o\in O(h)} z_{hot} = 1 \quad \forall t \in T, h \in H(t) \tag{5.5}$$

$$w_{hoo't} \ge \sum_{h'\in N(h)} z_{h'o't} - (1 - z_{hot}) \cdot bigM \quad \forall t \in T, h \in H(t), o \in O(h), o' \in U(o) \tag{5.6}$$

$$\sum_{o'\in O\setminus U(o)} \sum_{h'\in N(h)} z_{h'o't} \le (1 - z_{hot}) \cdot bigM \quad \forall t \in T, h \in H(t), o \in O \tag{5.7}$$

$$\sum_{p\in P} \varphi_p \cdot \delta_p^{CPRI} \cdot q_{hotp} \ge \alpha_h \cdot z_{hot} \quad \forall t \in T, h \in H(t), o \in O(h) \tag{5.8}$$

$$\sum_{o\in O}\sum_{p\in P} q_{hotp} = 1 \quad \forall t \in T, h \in H(t) \tag{5.9}$$

$$\sum_{p\in P} \varphi_p \cdot \delta_p^{S1} \cdot m_{otp} \geq \sum_{h\in H} \beta_h \cdot z_{hot} \quad \forall t \in T, o \in O \tag{5.10}$$

$$\sum_{p\in P} \varphi_p \cdot \delta_p^{X2} \cdot n_{oo'tp} \geq \sum_{h\in H} \gamma_h \cdot w_{hoo't} \quad \forall t \in T, o \in O, o' \in U(o) \tag{5.11}$$

$$n_{oo'tp} = n_{o'otp} \quad \forall t \in T, o \in O, o' \in U(o), p \in P \tag{5.12}$$

$$a_{op} \geq \sum_{h\in H} q_{hotp} + m_{otp} + \sum_{o'\in U(o)} n_{oo'tp} \quad \forall t \in T, o \in O, p \in P \tag{5.13}$$

$$\sum_{c\in C} \xi_{cp} \cdot l_{oc} \geq a_{op} \quad \forall o \in O, p \in P \tag{5.14}$$

$$\sum_{e\in E} \rho_e \cdot y_{oe} \geq \sum_{c\in C} l_{oc} \quad \forall o \in O \tag{5.15}$$

$$\sum_{e\in E} \sigma_e \cdot y_{oe} \geq \sum_{p\in P} \alpha_p \cdot a_{op} \quad \forall o \in O \tag{5.16}$$

$$l_{oc} \leq \rho_e \cdot \eta_{ec} + (1 - y_{oe}) \cdot bigM \quad \forall o \in O, e \in E, c \in C \tag{5.17}$$

The objective function (5.1) minimizes the cost of the VM configurations, routing equipment, line cards, and transponders to install in main COs.

Constraint (5.2) guarantees that at most one VM configuration is assigned to a main CO. Constraint (5.3) ensures that a main CO is equipped with routing equipment if and only if it is equipped with a VM configuration. Constraint (5.4) ensures that a VM configuration selected in each main CO has enough VMs to satisfy BBU virtualization for the RRHs assigned to it. Constraint (5.5) ensure that RRHs are assigned to one and only one accessible main CO at each time when they are active. Equation (5.6) accounts whether X2 interface link between main COs *o* and *o'* is required for RRH *h* at time *t*. Constraint (5.7) guarantees that, if RRH *h* is assigned to main CO *o*, their neighboring RRHs are not assigned to COs that cannot be accessed from main CO *o*; i.e., to guarantee that X2 interface links would not exceed the delay constraint. Constraints (5.8) and (5.9) guarantee that transponder *p* selected for CPRI link of active RRH *h* at time *t* has enough capacity and that one and only one transponder is selected (note that CPRI links are not multiplexed). Constraint (5.10) ensures that capacity of transponders selected in main CO *o* for S1 interface links is enough to satisfy the total S1 interfaces' capacity required in *o* at each time. Similarly, constraint (5.11) ensures that

capacity of transponders selected for X2 interface links between main COs is enough to satisfy the required capacity for X2 interfaces in *o* at each time. Constraint (5.12) ensures the same transponders' configuration is selected for X2 interfaces between main COs *o* and *o'*. Constraint (5.13) accounts the number of transponders of each type to install in main CO *o* to guarantee the required connections at any time. Constraint (5.14) ensures that the cards to install in each main CO can support the selected transponders. Constraints (5.15) and (5.16) guarantee that the selected switching equipment has enough slots and capacity, respectively. Finally, constraint (5.17) ensures that if routing equipment *e* is assigned to main CO *o*, and it does not support line card *c*, that line card is not installed in *o*.

Considering the particular case where the exact number of main COs to equip is given, the parameter Φ representing the number of main COs to equip is defined and the model extended with the following constraints:

$$\sum_{t\in T}\sum_{h\in H(T)} z_{hot} \geq \sum_{e\in E} y_{oe} \quad \forall o \in O \tag{5.18}$$

$$\sum_{e\in E}\sum_{o\in O} y_{oe} = \Phi \tag{5.19}$$

Constraint (5.18) ensures that only main COs that host BBUs assigned to active RRHs at some time are equipped, whereas constraint (5.19) ensures that Φ main COs are equipped.

5.2.4 Illustrative results

A. Scenario

For evaluation purposes, we consider a scenario where 49 RRHs, e.g., representing MBSs, are geographically distributed. The outmost cells cover regions where the traffic load varies according to a business load profile and the central ones vary according to a residential profile similarly as described in [CMR], e.g., representing an urban area surrounded by industrial zones; Fig. 5-6a depicts the reference scenario.

In addition, a set of RRHs, e.g., corresponding to small cells, are also geographically distributed for offloading purposes resulting in a scenario with 195 RRHs; it is worth highlighting that not all of them will be active simultaneously, since the TPs vary along the daytime. Nonetheless, MBS' RRHs are considered always active to guarantee coverage even in off-peak hours; whereas small cells' RRHs are progressively activated (deactivated) as load increases (decreases). Fig. 5-6b illustrates the number of active small cells' RRHs per MBS required for the two profiles against hour of day. A set of main COs that can be selected to host virtualized BBU pools is considered. Their location is illustrated in Fig. 5-6a. We

target a maximum 150 µs RTT delay between RRHs and BBUs and, as a consequence, no single main CO can be accessed by all RRHs in the evaluated scenarios. One sector is considered in each cell.

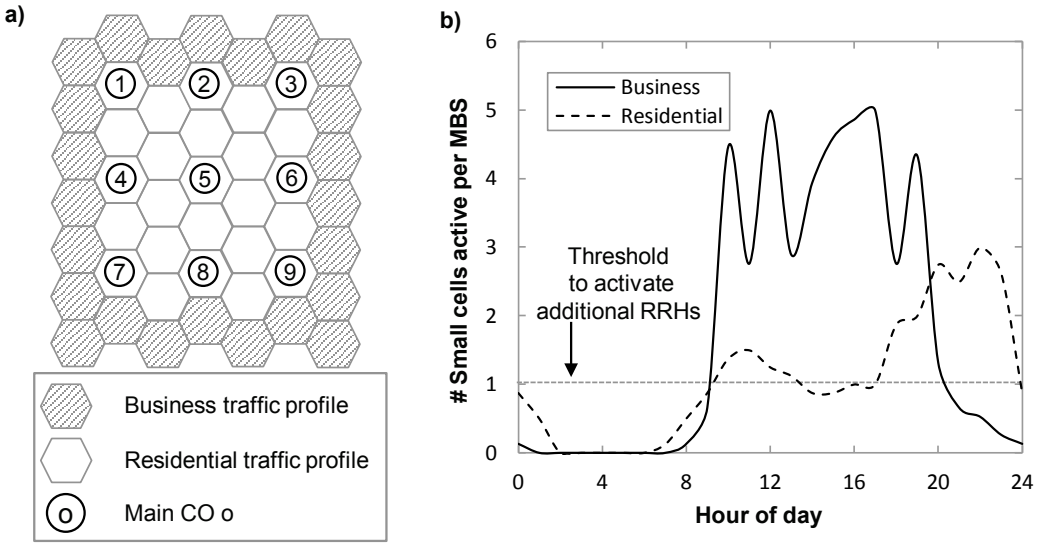

Fig. 5-6 Cells and main CO placement (a) and number of small cells active, per MBS, against the daytime (b).

B. CAPEX Study

CAPEX is studied from the network equipment perspective (IP/MPLS routers, line cards, and transponders to install in main COs) and different centralization levels compared. The network equipment's cost is based on the cost model in [Ra13]. Virtualized BBU pools' cost is not considered. The ILP model proposed for the CRAM problem was implemented and several instances were solved using CPLEX.

To study the impact of the centralization level in CAPEX, firstly we consider different LTE-A configurations in the previously described scenario and solve problem instances for representative daytime hours; i.e., peak and off-peak hours.

Fig. 5-7 shows the network equipment cost evolution against the number of main COs to equip at peak hours in business (12 p.m.) and residential areas (10 p.m.) and two configurations: *i*) LTE-A 4x4 Multiple Input Multiple Output (MIMO) [Ge07] with 40 MHz spectrum width (Fig. 5-7a and b), requiring CPRI links' capacity close to 10 Gb/s and S1 and X2 links' capacity close to 600 Mb/s and 230 Mb/s, respectively, and *ii*) LTE-A 4x4 MIMO with 100 MHz spectrum width (Fig. 5-7c and d), requiring CPRI links' capacity close to 25 Gb/s and S1 and X2 links' capacity close to 1.5 Gb/s and 550 Mb/s, respectively. As expected, the maximum centralization level requires two main COs in any case. Indeed, for the 40 MHz configuration (Fig. 5-7a and b), equipping the same two COs at any time with the cheapest equipment configuration, results in the minimum cost solution.

Interestingly, as soon as CPRI links' capacity increases (Fig. 5-7c and d), e.g., due to a configuration upgrade from 40 MHz to 100 MHz, the number of main COs to equip with minimum cost moves away from the fully centralized solution at peak hours. Results for off-peak hours showed that the fully centralized case, two COs, satisfies the demand at that time and with the minimum cost. As it can be seen in Fig. 5-7c and d, equipping more than 7 and 4 COs at 12 p.m. and 10 p.m., respectively, increases the cost.

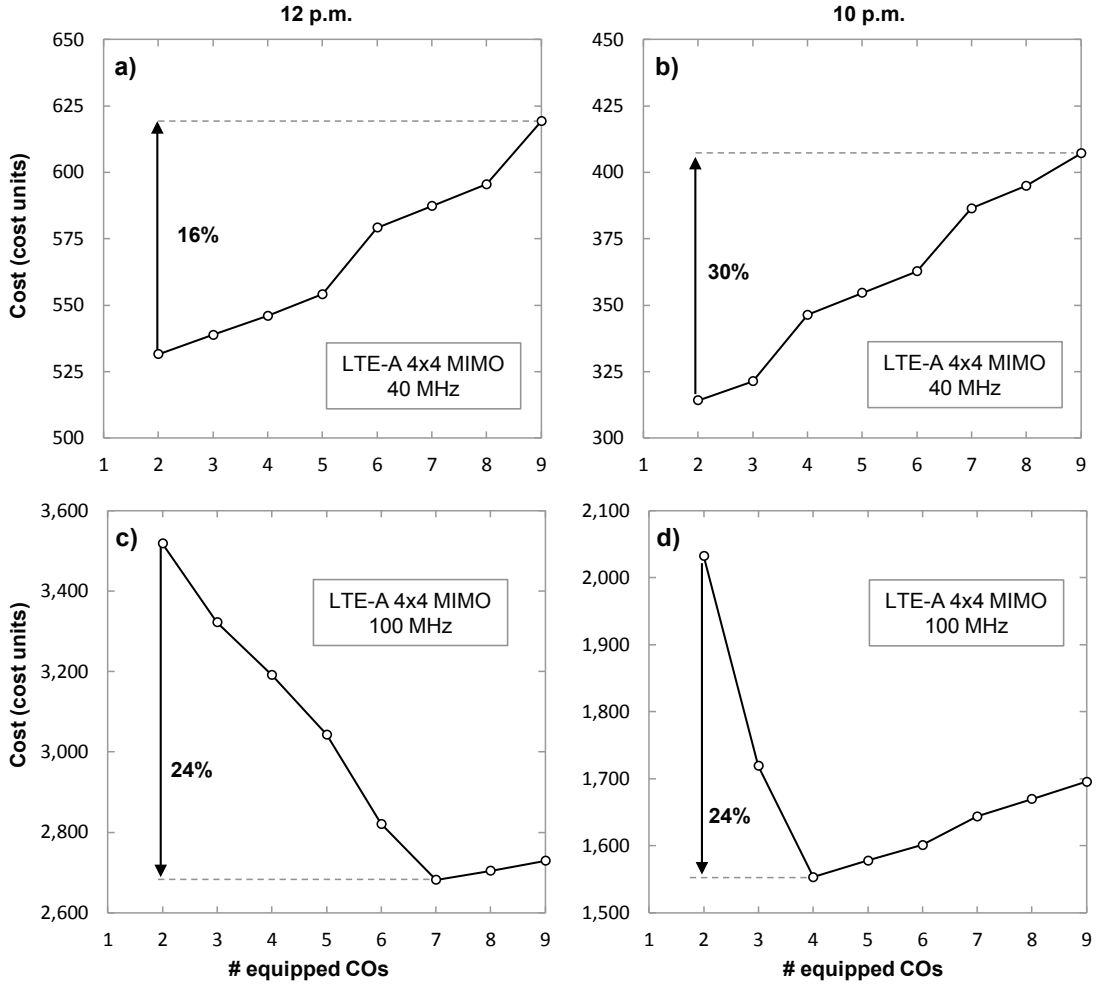

Fig. 5-7 Cost evolution against the number of main COs to equip for two different LTE-A configurations at 12 p.m. and 10 p.m.

Considering the 100 MHz configuration and aiming at dimensioning our scenario, we restricted the set of COs that can be selected to seven main COs, corresponding to the ones that need to be equipped to satisfy demand at peak hours and that can be selected to satisfy demand at any time. The problem was solved for each hour separately and the minimum cost solutions obtained were saved. Then, each main CO was dimensioned with the minimum equipment to satisfy demand at any daytime hour. Although the proposed mathematical model can solve the problem

considering all daytime hours jointly, splitting the problem into different instances per each hour allows solving it in reasonable times, while obtaining good enough solutions as it will be seen in the next paragraphs. Results showed that by equipping seven main COs with the smallest IP/MPLS routers, demand is satisfied at any time. More specifically, required equipment in each main CO resulted in 2,867.6 cost units in terms of CAPEX.

Similarly, we dimensioned the same configuration scenario considering the fully centralized approach, where only two COs can be equipped, and a theoretical fully distributed approach, where 49 main COs are equipped, each to serve a single MBS' RRH and its small cells' RRHs. From the results, the fully centralized approach required a huge capacity router (6.72 Tb/s and 48 slots) and a small one (2.24 Tb/s and 16 slots), whereas the fully distributed required 49 routers of the smallest capacity (1.40 Tb/s and 10 slots). CAPEX value obtained for the fully centralized approach was 3,518.4 cost units, whereas for the fully distributed one, was 4,694.9 cost units. The solution obtained when 7 COs were equipped, represents CAPEX savings as high as 18% and 39% compared to the cases where 2 and 49 COs were equipped respectively.

Focusing on the main COs to equip hour by hour, Fig. 5-8 illustrates that during off-peak hours only two COs need to be equipped whereas for peak hours more main COs need to be equipped. An elastic CO network equipment use is envisioned.

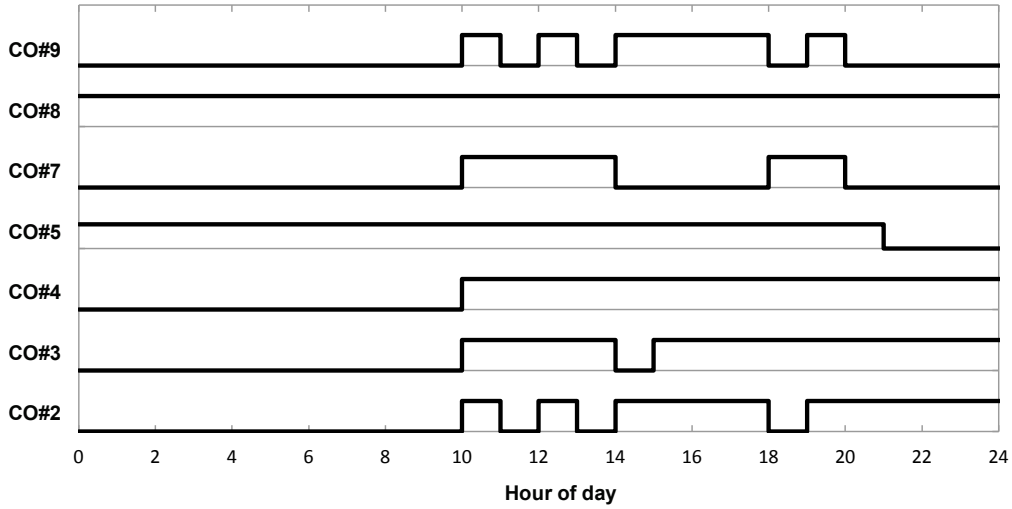

Fig. 5-8 Main COs to equip against daytime hours.

C. OPEX Study

For completeness, we also study the impact of the centralization level taking into account the power consumption of the equipment along the daytime.

In line with [He12], we assume that the power consumption of routers can be approximated as the summation of the consumption of the basic node, the slots

cards, and the port cards. In addition, we consider a fixed component of power consumption in IP/MPLS routers related to the basic node and its slots power requirements and a variable contribution from the line cards and transponders in use, assuming that they only consume when they are in use.

Fig. 5-9 represents the power consumption of transponders (Fig. 5-9a) and total power consumption considering the whole equipment in all main COs (Fig. 5-9b) against daytime hours. As expected, since the fully centralized approach is the one requiring the lowest number of transponders to be installed, their contribution to the power consumption is also the lowest. On the contrary, the distributed approach is the one requiring more transponders, since each main CO requires the necessary equipment not only for the CPRI interfaces, but also for the X2 and S1 interfaces. The solution requiring 7 COs to be equipped, results in a slightly increment of 5% in terms of transponders' power consumption compared to the centralized approach, and savings near 37% compared to the distributed one. Notwithstanding, the contribution to power consumption from routers and line cards needs to be considered to evaluate OPEX.

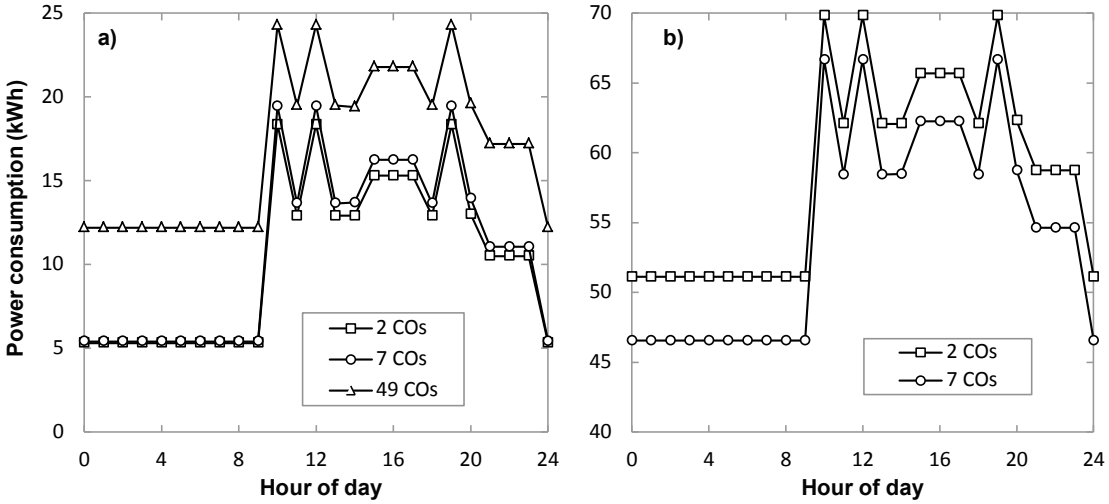

Fig. 5-9 Power consumption of transponders (a) and total equipment (b).

As described in the CAPEX study, because of the equipment selection for CAPEX minimization, the centralized approach requires a huge capacity router plus a small one, and the distributed approach requires 49 routers of the smallest capacity. For the centralized approach, it is clear that the large power consumption of the huge router will impact the total power consumption, even though, the lowest number of transponders is required. As shown in Fig. 5-9b, the centralization level requiring 7 COs presents lower total power consumption than the fully centralized approach; savings close to 7% are observed. Values for the fully distributed approach are not depicted in Fig. 5-9b, since computing only the fixed contribution from the 49 routers of the smallest capacity is as high as 270

kWh (49 x 5.51 kWh). The 7 COs solution shows savings close to 82% compared to the fully distributed approach.

Finally, as showed in Fig. 5-9a, it is clear that for any of the approaches considered, equipment usage follows curves along daytime similarly as traffic load figures shown in Fig. 5-6.

5.3 Connectivity requirements

From Chapter 4 and the current chapter, connectivity requirements for DC interconnection in DC federations to support elastic operations and to support services on the telecom cloud are devised and summarized in Table 5-5.

Since huge amounts of data need to be conveyed during some periods of time, networks supporting capacities in the order of hundreds of Gb/s are required. Moreover, since communications costs need to be considered when performing elastic operations in DC federations, flexible-grid network and an intelligent control plane are needed to support different capacities required along the time and to accept connection requests on-demand, respectively. In addition, mechanisms to improve network resources availability, when they are dynamically required, need to be explored.

Table 5-5 Connectivity requirements for DC interconnection

		Services on telecom cloud	
	DC federations	**Live-TV distribution**	**C-RAN**
Huge capacity	✓	✓	✓
Bandwidth-on-demand	✓	✓	✓
High network resources availability	✓	✓	✓
Stringent QoS		✓	✓
Bitrate guarantees			✓

Finally, based on the studies carried out in this chapter, in addition to bandwidth-on-demand and huge capacity, service-specific SLA parameters need to be considered when orchestrating services in the telecom cloud. Delay, as a measure of QoS, can be required both in live-TV and C-RAN scenarios, whereas bitrate guarantees may be required to avoid service interruption in the mobile network; e.g., an eventual failure of a link supporting the S1 interface would then be

translated into a reduction in the offered capacity to the users but would avoid service interruption.

5.4 Conclusions

In this chapter, scalability of centralized and distributed telecom cloud network architectures and impact of the centralization level on network costs to support services in the telecom cloud have been studied. To that end, two different optical network-supported scenarios have been considered: one based on live-TV distribution; the other based on C-RAN. Results showed that relaxing the centralization level in telecom cloud-based services, results in remarkable network CAPEX savings.

Specifically, results showed that a distributed architecture scales the best for live-TV distribution compared to a fully centralized approach. Moreover, the number of SBVTs to be installed is noticeable lower, which results in a lower total network CAPEX for the distributed architecture. CAPEX savings near 30% were observed in the scenarios considered, representing the evolution of SD, HD, and 4K UHD penetration along different years.

Regarding the impact of the centralization level of C-RAN in the network costs, results showed that, in the evaluated scenarios, although the maximum centralization level results in the minimum CAPEX solution for certain LTE-A configuration, as soon as higher capacities are required in different LTE-A interfaces (e.g., due to a configuration upgrade) lower levels of centralization result in CAPEX savings up to 18% compared to the fully centralized approach. Savings as high as 37% were observed compared to a fully distributed approach, For completeness, OPEX was also studied from the solutions obtained after solving the CRAM problem. OPEX savings near 7% and up to 82% were shown for the solution requiring a low level of centralization compared to the fully centralized and fully distributed approaches, respectively.

Finally, we summarized connectivity requirements for DC interconnection considering the studies carried out in this and the previous chapter. Although common connectivity requirements such as huge capacity, bandwidth-on-demand, and high network resources availability are devised, service-specific SLA parameters lead to particular connectivity requirements in addition to the previously mentioned; e.g., stringent QoS and bitrate guarantees.

The next chapters propose different connectivity schemes, algorithms, and architectures aiming at satisfying the aforementioned connectivity requirements.

Chapter 6

Dynamic and elastic connections

In the previous chapters, we studied two different scenarios for interconnection networks and concluded that bandwidth-on-demand is required. In this chapter, we study different types of dynamic connectivity in optical networks.

In fact, current inter-DC connections are configured as static *big fat pipes*, which entails large bitrate over-provisioning and high operational costs for DC operators. In addition, network operators cannot share such connections between customers, because DC traffic highly varies over time. To improve resource utilization and save costs, dynamic inter-DC connectivity is being currently targeted from the research and the standardization points of view.

To facilitate dynamic inter-DC connectivity and to convey large amounts of data, in this chapter, first, we propose a distributed *cross-stratum orchestrator*, CSO, to coordinate cloud and network and study two polling-based connectivity models: *dynamic* and *dynamic elastic*. The study shows that dynamic connectivity is not enough to guarantee elastic DC operations and might lead to poor performance provided that not enough over-provisioning of network resources is performed; whereas dynamic elastic connections can alleviate it to some extent. Next, we explore the elastic SA capability of EON and, in this context, we study the effectiveness of three alternative SA policies, namely *Fixed*, *Semi-Elastic*, and *Elastic*. For each elastic SA policy, we develop a dedicated algorithm which is responsible for adaptation of spectrum allocated to lightpath connections in response to traffic changes.

6.1 Dynamic connectivity models

In contrast to *static* connectivity, dynamic connectivity allows DCs to manage optical connections to remote DCs, requesting connections as they are really needed to perform data transferences and releasing them when all data has been transferred. After the CSO requests a connection and negotiates its capacity as a function of the current network availability, the resulting bandwidth can be used by scheduling algorithms to organize transferences. Fig. 6-1 represents the assigned bitrate to a connection along the time for the static and the dynamic connectivity models to transfer data between times t_1 and t_3. It is clear that, although in both examples the network usage along the time is the same, the network resources assigned are over-dimensioned in the static model, since resources to provide a 40 Gb/s connection are always allocated, even though they are not used. On the contrary, when dynamic connectivity is considered, network resources are allocated when they are needed, from t_1 to t_3.

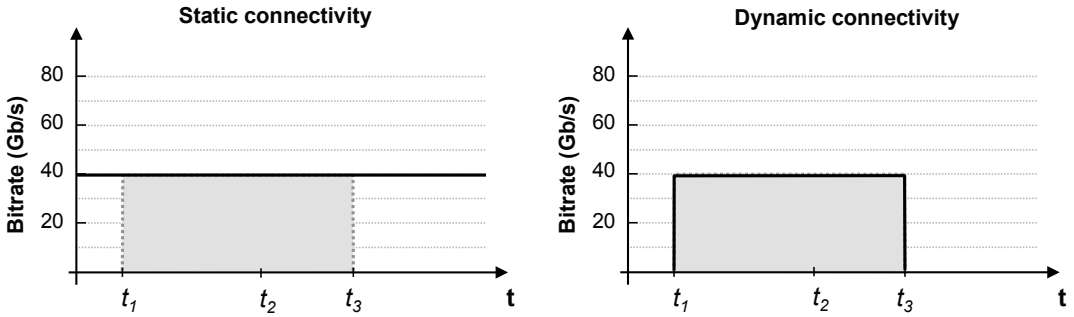

Fig. 6-1 Example of connection's bitrate against time for the static and dynamic connectivity models.

Nonetheless, the availability of resources is not guaranteed and lack of network resources at requesting time may result in long transference times and even in transference period overlapping. Note that connection's bitrate cannot be re-negotiated and remains constant along connection's holding time. To reduce the impact of the unavailability of required connectivity resources, we propose to use elasticity that the flexgrid technology provides, allowing to increase the amount of spectral resources assigned to each connection, and thus its bitrate. This adaptation is performed if there are not enough resources at request time, requesting for more bandwidth at any time after the connection has been set-up. We name this type of connectivity as dynamic elastic. Fig. 6-2 represents the connection's bitrate along the time according to the dynamic and the dynamic elastic connectivity models. Since resources allocated to a connection based on the dynamic elastic model may be increased along the time, and bitrate increased accordingly, data transference would then finish earlier and resources would be released sooner.

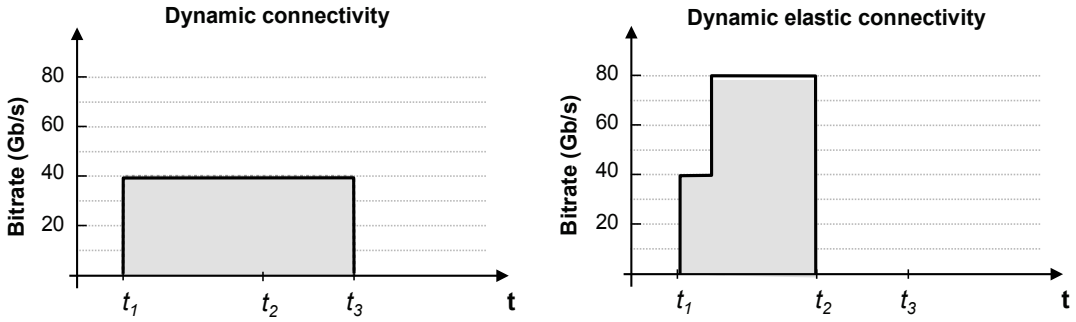

Fig. 6-2 Example of connection's bitrate against time for the static and dynamic connectivity models.

6.2 Using connectivity models for elastic operations

Aiming at satisfying connectivity requirements related to bandwidth-on-demand, in this chapter, we study two different polling-based connectivity models (dynamic and dynamic elastic) and evaluate their performance in comparison with the commonly implemented *static* model. To that end, cloud-ready transport network is assumed and it is used to interconnect DCs in different locations and to provide bandwidth-on-demand. To allow dynamic connection requests, different approaches can be considered as shown in Section 6.1. Among different candidates to implement the intelligent control plane, we select ABNO.

Moreover, from the DC perspective, to implement the above-mentioned dynamic connectivity models, we propose a distributed CSO to coordinate cloud and network. Our CSO interfaces ABNO. CSO can manage inter-DCs connectivity not only dynamically, but also elastically and request ABNO to modify the amount of bandwidth of already established optical connections.

To the best of our knowledge, elasticity is not currently supported by any other interface other than ABNO's northbound interface. In addition to the cloud and network management for local DC resources, our CSO implements new components to facilitate inter-DC operations:

- an IT resources coordination and synchronization module in charge of coordinating VMs migrations and databases (DB) synchronizations among DCs,
- a connection manager and a virtual topology DB.

The set of VMs to migrate and the set of DBs to synchronize are computed by the scheduler inside the DC resource manager. Once those sets are computed, the CSO module coordinates intra- and inter-DC networks to perform the transferences.

In line with the architecture considered in Chapter 4, Fig. 6-3 illustrates the architecture assumed in this section, with ABNO interfacing CSO.

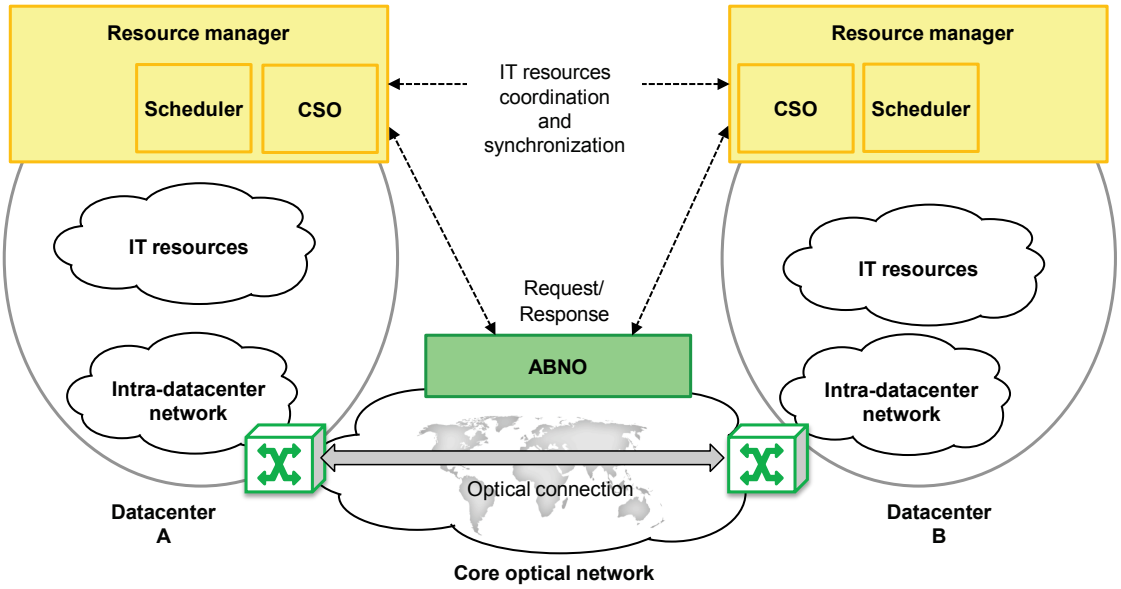

Fig. 6-3 Architecture considered with cross-stratum orchestrator.

The sequence in Fig. 6-4 for both dynamic connectivity models illustrates messages exchanged between DC resource managers and ABNO to set-up and tear-down an optical connection. In Fig. 6-4, once the algorithms in the DC resource manager have computed a transference to be performed, DC resource manager requests an 80 Gb/s optical connection to a remote DC. The policy agent inside ABNO verifies local policies and performs internal operations (in our implementation, it forwards the message to the PCE) to find a route and spectrum resources for the request. Assuming that not enough resources are available for the bitrate requested, an algorithm inside ABNO finds the maximum bitrate and, at time t_0, sends a response to the originating DC resource manager with that information. Upon the reception of the maximum available bitrate, 40 Gb/s in the example, DC resource manager re-computes the transference, reduces the amount of data to transfer, and requests a connection with the available bitrate. When the transference ends (t_1), DC resource manager sends a message to the network control plane to tear-down the connection and the used resources are released so they can be assigned to any other connection.

In the dynamic elastic connectivity model, DC resource manager is able to request increments in the bitrate of already established connections. In the example in Fig. 6-4, after the connection has been established in t_0 with half of the initially requested bitrate, DC resource manager sends periodical retrials to increment its bitrate. In the example, some network resources have been released after the connection has been established and, after a request is received in t_2, they can be assigned to increment the bitrate of the already established connection; the

assigned bitrate increases then to 80 Gb/s, which reduces the total transference time. Note that this is beneficial for both DC federation, since better performance can be achieved, and the network operator, since unused resources are immediately occupied.

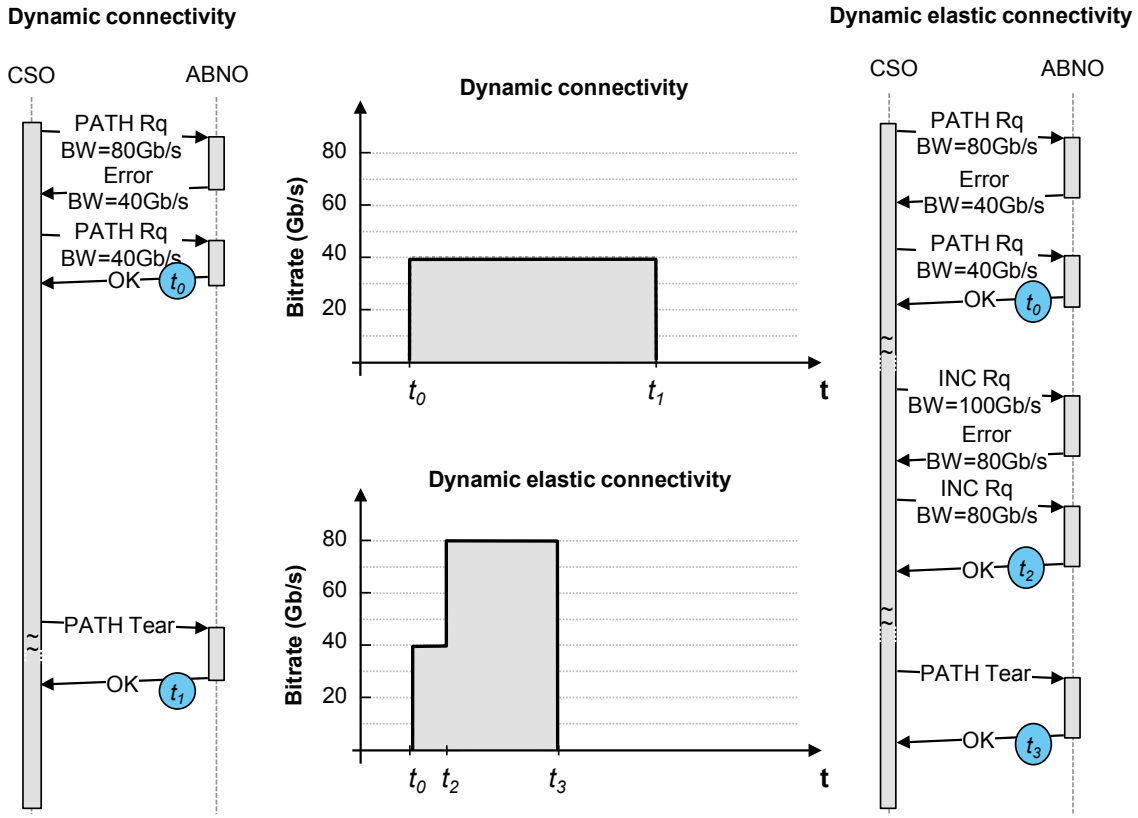

Fig. 6-4 Dynamic and dynamic elastic connectivity models.

6.2.1 Performance evaluation

For evaluation purposes, we implemented the CSO on a scheduler in the OpenNebula cloud management middleware, whereas the flexgrid network and the ABNO architecture were implemented as an ad-hoc event-driven simulator in OMNeT++. An XML-based protocol was developed to communicate CSO and ABNO. In the scheduler inside cloud management, a *follow-the-work* strategy was implemented for VM migration, where the VMs are moved to DCs closer to the users, reducing the user-to-service latency. The DB synchronization policy tries to update the differential images between services running in all the DCs; in the case that an image cannot be synchronized in time, the next update will attempt to synchronize the whole DB image, increasing the DC2DC traffic overhead. Regarding PCE, the algorithms described in [Ca12] and [Kl13] for RSA and elastic operations, respectively, were implemented.

We consider the global 11-node topology depicted in Fig. 6-5 where locations are used as source for U2DC traffic. DCs are placed in four locations: Illinois, Spain, India, and Taiwan. DC2DC and U2DC traffic compete for resources in the physical network.

Users' traffic connection requests arrive following a Poisson process and are sequentially served without prior knowledge of future incoming connection requests. The destination DC of each U2DC connection is the closest DC to the user's location in case of requesting a distributed service or to the DC containing the VM in case of singular services. To represent user's activity along a day, the bitrate actually demanded by each U2DC connection request is proportional to the number of active users at the requesting time, e.g., several Gb/s during office hours and few Mb/s during nights. The holding time of connections is exponentially distributed with the mean value equal to 2 hours. Different values of offered network load can be considered by changing the arrival rate while keeping the mean holding time constant. Specifically, we considered two offered load for the U2DC traffic:

- a) low load unleashing U2DC blocking probability < 0.2%;
- b) high load unleashing blocking probability < 0.6%.

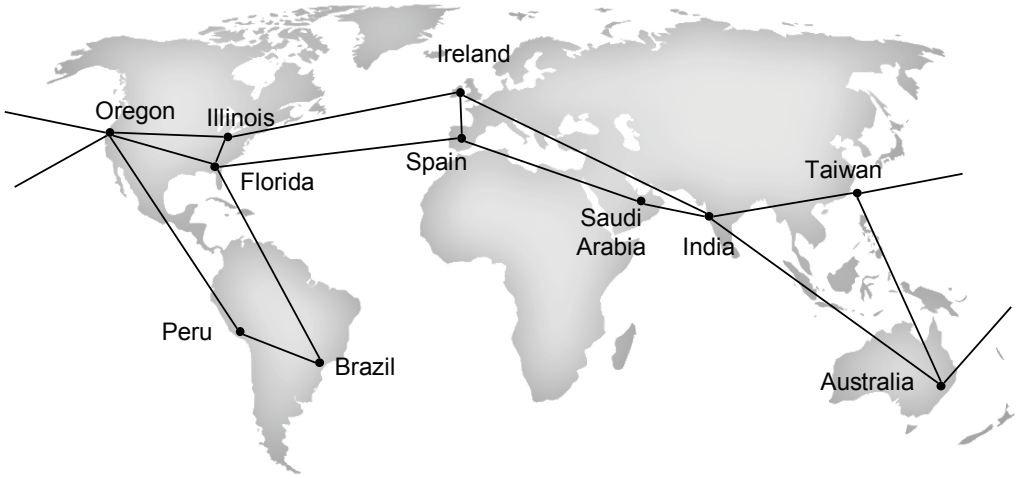

Fig. 6-5 Network scenario

To conduct all the simulations, optical spectrum width was set to 4 THz, spectral granularity to 6.25 GHz, and the capacity of transponders interfacing the flexgrid network to 1 Tb/s. As for the DC federation, the number of VMs was set to 35,000 with an image size of 5 GB each, while the number of DBs was set to 300,000 distributed among DCs, each with a size of 5 GB, half the size of Wikipedia [WIKISIZE]; at each scheduling round we assume that only 450 MB change. In addition, protocol stack overheads were considered: raw data to be transferred is transported in TCP packets, which are encapsulated into 1,442-byte IPv4 packets; MPLS and gigabit Ethernet headers are added afterwards.

We set a scheduling round each hour, so the objective is to perform the required VMs migrations and DB synchronizations within each time period. In fact, the shorter the transfer time, the better the offered network service, by having earlier the VMs in their proper locations and the DBs up-to-date. Fig. 6-6 and Fig. 6-7 present results for the static connectivity. In Fig. 6-6a, the bitrate of static connections is plotted as a function of the average time to complete transferences so as to ensure all transferences are performed within 1 hour, for both VMs migrations and DB synchronizations. It is clear that the larger the bitrate of the connections, the shorter the transferences. To minimize the bitrate over-provisioned for the static connections, let us assume an objective of 30 minutes on average for the transferences. Thus, connections of 200 Gb/s and 150 Gb/s are needed for DB synchronization and VM migration, respectively; although one connection would be enough in this case, two different connections are established for the sake of differentiating usage. Using those bitrates, *time-to-transfer* VMs and DBs as a function of the time along one day in the connection between DCs in Spain and Illinois are plotted in Fig. 6-6b. As shown, time-to-transfer is always lower than 60 minutes, although with peaks of 50 minutes and above.

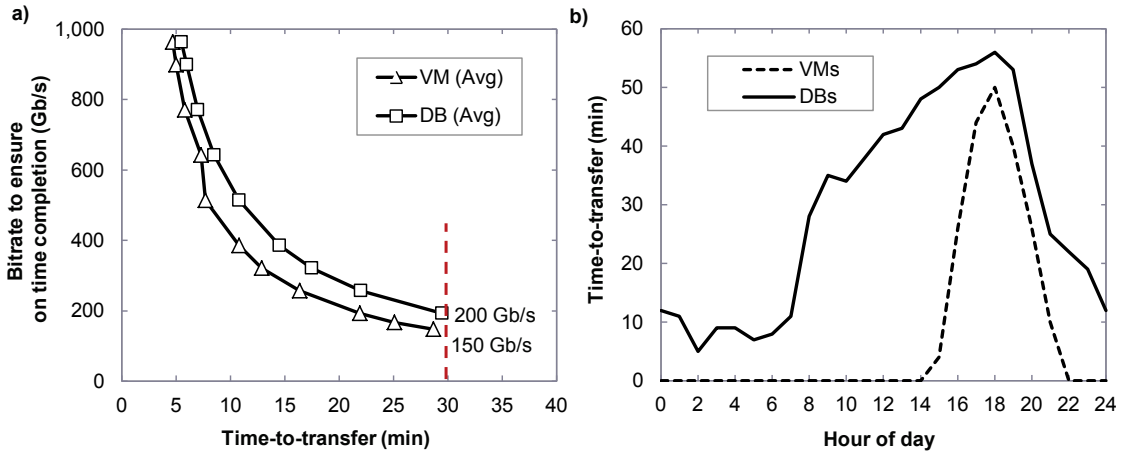

Fig. 6-6 Connections dimensioning for static connectivity.

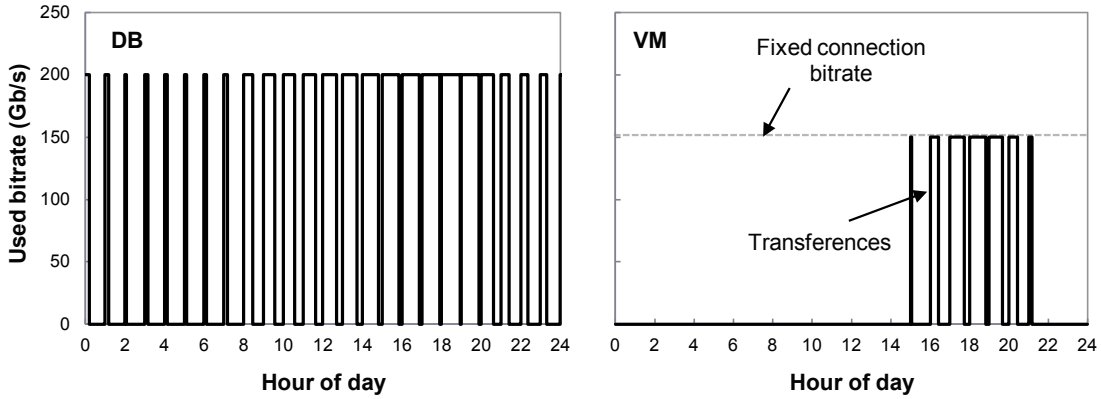

Fig. 6-7 Static connectivity.

The used bitrate to synchronize DBs and get VMs migrated is also illustrated in Fig. 6-7. Note that since the bitrate of the optical connections has been fixed to 200 Gb/s for DBs and to 150 Gb/s for VMs, different time-to-transfers are obtained in line with the time-to-transfer plots.

Fig. 6-8 and Fig. 6-9 show the used bitrate for the dynamic and the dynamic elastic connectivity models. Two different cases are considered regarding U2DC traffic: low (Fig. 6-8) and high (Fig. 6-9). Recall that U2DC and DC2DC traffics compete for network resources.

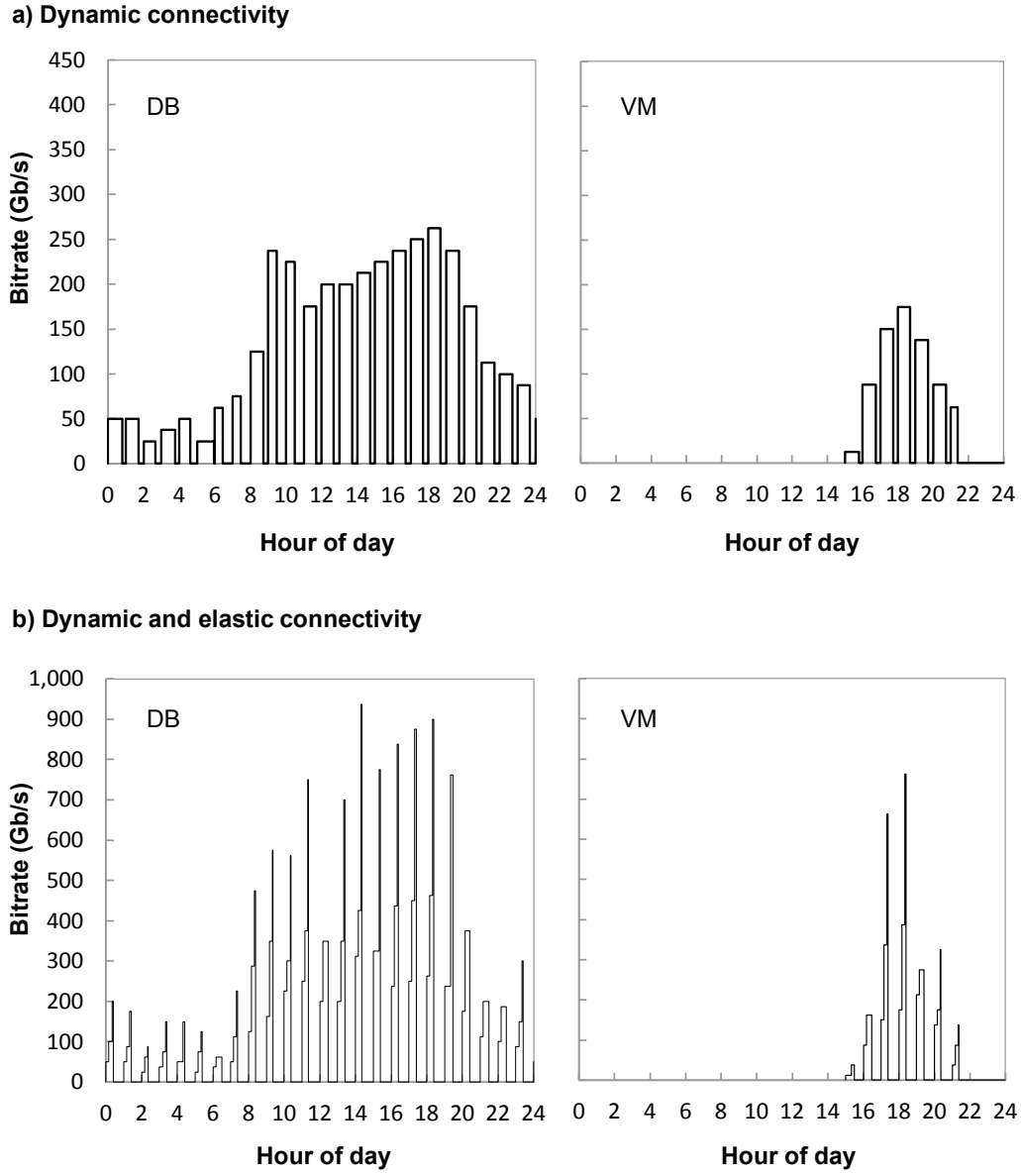

Fig. 6-8 Dynamic connectivity. Low U2DC traffic.

In the dynamic model under low U2DC, time-to-transfer is kept clearly under 60 minutes for both DBs and VMs. Note however that connection's bitrate varies with

the amount of data to transfer; connections bitrate for DBs are as low as 50 Gb/s and as high as 250 Gb/s, in contrast to the constant 200 Gb/s bitrate used in the static model. However, as soon as the U2DC traffic increases, the amount of needed bitrate for the dynamic connections might not be available at requesting time, so this model is not able to perform data transfers within 1 hour leading to period overlapping, as shown in Fig. 6-9; this fact yields poor performance as some DBs become degraded and users perceive an increased latency.

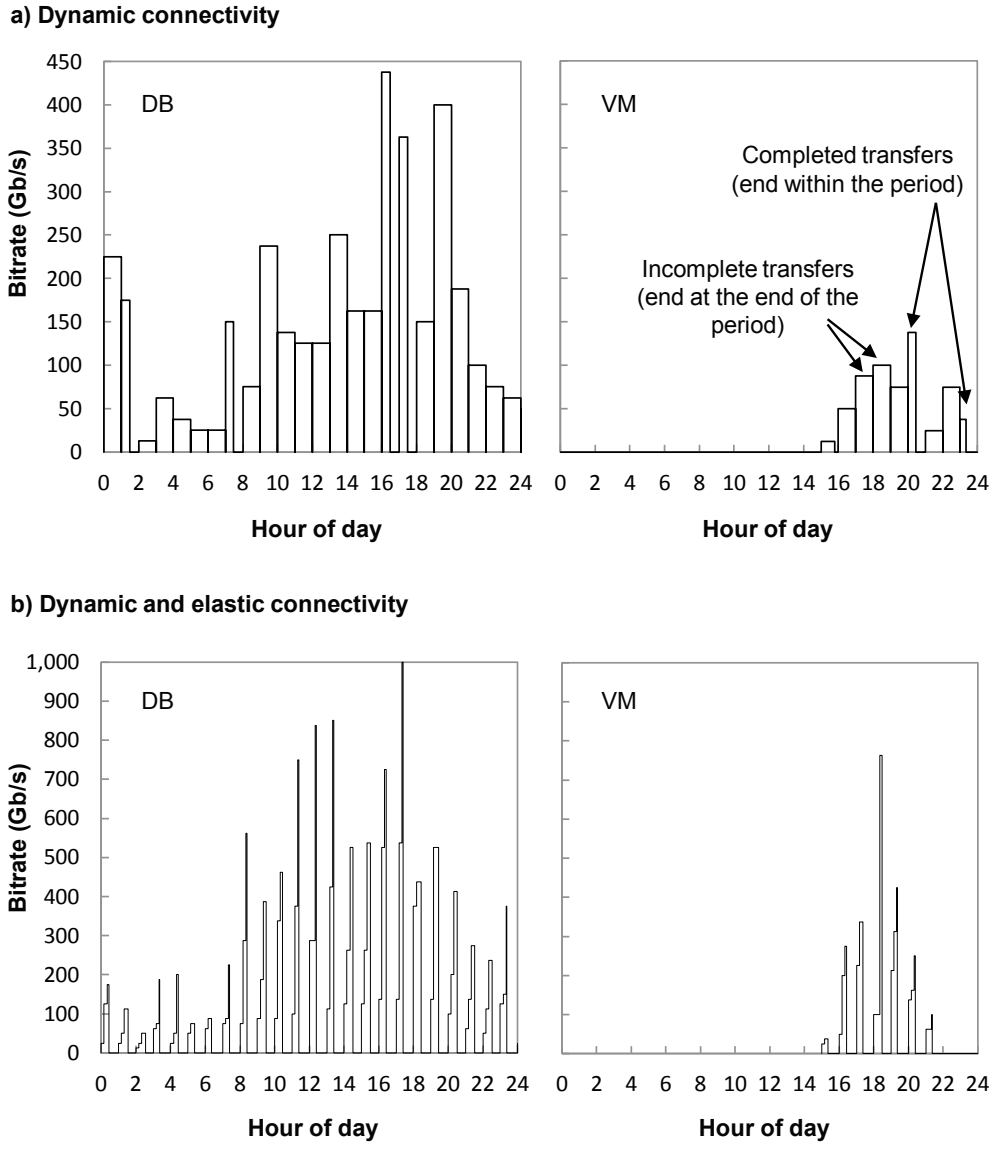

Fig. 6-9 Dynamic connectivity. High U2DC traffic.

The dynamic elastic model is able to keep time-to-transfer under 60 minutes for both low and high U2DC traffic. As shown, the initial bitrate for the connections is comparable to that of the connections for the dynamic model. However, since CSO can perform elastic bitrate operations on the established optical connections, it is

able to increase connections bitrate even under high U2DC traffic. In fact, as detailed in Table 6-1, transfer times are clearly lower than that obtained with the dynamic and even with the static connectivity models.

The cost of elasticity is at the control plane, since the amount of messages that need to be processed slightly increases (Table 6-2); 40.8 request messages per hour to increase connections' capacity in addition to 36.8 messages per hour for connections' setup and teardown. More than half of requests (53.8%) were successful resulting in expanded optical connections; moreover, each connection got 1.67 increments on average during each period. It is worth highlighting that although the remaining 46.2% requests were not successful, the corresponding connections maintained their prior assigned capacity.

Finally, it is worth pointing out that the bitrate savings obtained by implementing any of the proposed dynamic connectivity models is as high as 60% with respect to the static model (Table 6-1). Note that in the static model 12 connections for DB synchronization and 4 connections for VM migration need to be permanently established.

Table 6-1. Connectivity models comparison.

		Time-to-transfer (minutes)		
		Static	**Dynamic**	**Elastic**
Time-to-transfer (minutes)	**DB synchronization (Max. / Avg.)**	54.0 / 28.5	58.0 / 39.9	49.0 / 24.7
	VM migration (Max. / Avg.)	50.0 / 28.7	48.0 / 39.9	40.0 / 24.4
Bitrate savings		-	59.1%	57.9%

Table 6-2. Elastic operations per period.

Total signaling	**% success**	**# per connection**
40.8	53.8%	1.67

6.3 Adaptive spectrum allocation

Although optical networks with ABNO can manage dynamic connection requests, the network ability to deal with elastic operations to support the dynamic elastic connectivity model needs to be studied. To that end, in this section, we study how

connections requiring dynamic SA can be managed in EONs when their required bitrate varies once in operation.

In our study, we assume that the profile of the client flows arriving to the aggregation router is known. A TP is characterized by a guaranteed bitrate and a range of variation, given for a considered set of time periods. Therefore, lightpaths can be pre-planned, using for example the models in [Kl13], and established on the network. We assume that the lightpath capacity is limited. Therefore, several lightpath connections may exist between a pair of aggregation networks.

Once in operation, however, the established lightpaths must elastically adapt their capacity so as to convey as much bitrate as possible from the demanded by the aggregation networks. In particular, each change in aggregated traffic resulting in a change in the amount of optical spectrum of a lightpath (i.e., spectrum expansion/reduction) requires a request to be sent from the corresponding CSO to the ABNO controlling the flexgrid core network to find the appropriate SA for that lightpath. In response to this request, a dynamic SA algorithm implemented in ABNO is in charge of adapting the SA in accordance to the SA policy which is used in the network.

6.3.1 Spectrum allocation policies

Based on the static approach presented in [Kl13], the SA policies showed in Fig. 6-10 are considered for time-varying traffic demands in dynamic scenarios.

The SA policies put the following restrictions on the assigned CF and the allocated spectrum width, in particular:

- *Fixed* (Fig. 6-10a): both the assigned CF and spectrum width do not change in time. At each time period, demands may utilize either whole or only a fraction of the allocated spectrum to convey the bitrate requested for that period.
- *Semi-Elastic* (Fig. 6-10b): the assigned CF is fixed but the allocated spectrum may vary. Here, spectrum increments/decrements are achieved by allocating/releasing frequency slices symmetrically, i.e., at each end of the already allocated spectrum while keeping invariant the CF. The frequency slices can be shared between neighboring demands, but used by, at most, one demand in a time interval.
- *Elastic* (Fig. 6-10c): asymmetric spectrum expansion/reduction (with respect to the already allocated spectrum) is allowed and it can lead to short shifting of the central frequency. Still, the relative position of the lightpaths in the spectrum remains invariable, i.e., no reallocation in the spectrum is performed.

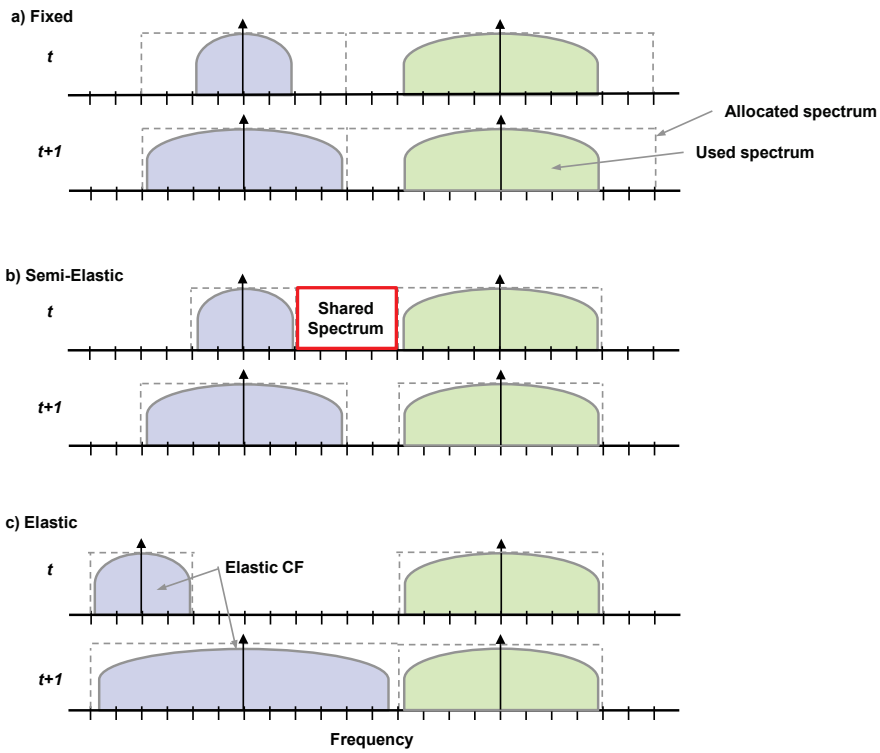

Fig. 6-10 Three SA policies for time-varying traffic in a flexgrid network. Two time intervals are observed: t before and t+1 after spectrum adaptation has been performed.

6.3.2 Problem statement

The problem of dynamic lightpath adaptation can be formally stated as:

Given:

- a core network topology represented by a graph $G(N, E)$, being N the set of BV-OXC nodes and E the set of bidirectional fiber links connecting two BV-OXC nodes; each link consists of two unidirectional optical fibers;
- a set S of available slices of a given spectral width for every link in E;
- a set L of lightpaths already established on the network; each lightpath l is defined by the tuple $<R_l, f_l, s_l>$, where the ordered set $R_l \subseteq E$ represents its physical route, f_l its central frequency, and s_l the amount of frequency slices;
- a lightpath $p \in L$ for which spectrum adaptation request arrives and the required number of frequency slices, $(s_p)^{req}$.

Output: the new values for the SA of the given lightpath p: $<R_p, f_p, (s_p)'>$ and $<R_p, (f_p)', (s_p)'>$, respectively, if the *Semi-Elastic* and *Elastic* policies are used.

Objective: maximize the amount of bitrate served.

6.3.3 Spectrum adaptation algorithms

For the *Fixed* SA policy, the allocated spectrum does not change in time. Therefore any fraction of traffic that exceeds the capacity of the established lightpath is lost. Regarding the *Semi-Elastic* and *Elastic* policies, the corresponding lightpath adaptation algorithms are presented in Table 6-3 and Table 6-4, respectively. In the following, we discuss the details of these algorithms.

- *Semi-Elastic* algorithm: the elastic operation is requested for a lightpath p and the required amount of frequency slices to be allocated, maintaining f_p invariant, is given. Since flexgrid is implemented, $(s_p)^{req}$ must be an even number. If elastic spectrum reduction is requested, the tuple for lightpath p is changed to $<R_p, f_p, (s_p)^{req}>$ (lines 1-2). In the opposite, when an elastic expansion is requested, the set of spectrum adjacent lightpaths at each of the spectrum sides is found by iterating on each of the links of the route of p (lines 4-7). The greatest value of available spectrum without CF shifting, s_{max}, is subsequently computed and the value of spectrum slices actually assigned to p, $(s_p)'$, is computed as the minimum between s_{max} and the requested one (lines 8-9). The tuple representing lightpath p is now $<R_p, f_p, (s_p)'>$.
- *Elastic* algorithm: the CF of p can be changed, so the difference with the *Semi-Elastic* algorithm explained above is related to that issue. Now, the value of s_{max} is only constrained by the amount of slices available between the closest spectrum adjacent paths. Then, s_{max} is the sum of the minimum available slices along the links in the left side and the minimum available slices in the right side of the allocated spectrum (line 9). Finally, the returned value $(f_p)'$ is obtained by computing the new CF value so as to minimize CF shifting (line 11).

Table 6-3 Semi-elastic SA algorithm

INPUT: $G(N,E)$, S, L, p, $(s_p)^{req}$
OUTPUT: $(s_p)'$

1: **if** $(s_p)^{req} \leq s_p$ **then**
2: $(s_p)' \leftarrow (s_p)^{req}$
3: **else**
4: $L^+ \leftarrow \emptyset$, $L^- \leftarrow \emptyset$
5: **for each** $e \in R_p$ **do**
6: $L^- \leftarrow L^- \cup \{l \in L: e \in R_l, \text{adjacents}(l, p), f_l < f_p\}$
7: $L^+ \leftarrow L^+ \cup \{l \in L: e \in R_l, \text{adjacents}(l, p), f_l > f_p\}$
8: $s_{max} \leftarrow 2 \cdot \min\{\min\{f_p - f_l - s_l, l \in L^-\}, \min\{f_l - f_p - s_l, l \in L^+\}\}$
9: $(s_p)' \leftarrow \min\{s_{max}, (s_p)^{req}\}$
10: **return** $(s_p)'$

Table 6-4 Elastic SA algorithm

INPUT: $G(N,E)$, S, L, p, $(s_p)^{req}$
OUTPUT: $(f_p)'$, $(s_p)'$

```
if (s_p)^req ≤ s_p then
  (s_p)' ← (s_p)^req
  (f_p)' ← f_p
else
  L^+ ← ∅, L^- ← ∅
  for each e ∈ R_p do
    L^- ← L^- ∪ {l ∈ L: e ∈ R_l, adjacents(l, p), f_l < f_p}
    L^+ ← L^+ ∪ {l ∈ L: e ∈ R_l, adjacents(l, p), f_l > f_p}
  s_max ← min{f_p − f_l − s_l, l ∈ L^-} + min{f_l − f_p − s_l, l ∈ L^+}
  (s_p)' ← min{s_max, (s_p)^req}
  (f_p)' ← findSA_MinCFShifting(p, (s_p)', L^+, L^-)
return (f_p)', (s_p)'
```

6.3.4 Illustrative results

A. Scenario

Since dynamic connection requests are currently supported by ABNO, in this study, we focus on the elastic capabilities supported in the flexgrid network to facilitate the dynamic elastic connectivity model.

We consider three scenarios representing different levels of traffic aggregation consisting in three different core networks: TEL21, TEL60, and TEL100, with 21, 60, and 100 nodes, respectively. These networks are based on the Telefonica (TEL) network presented in Fig. 6-11. The main characteristics of the networks are presented in Table 6-5.

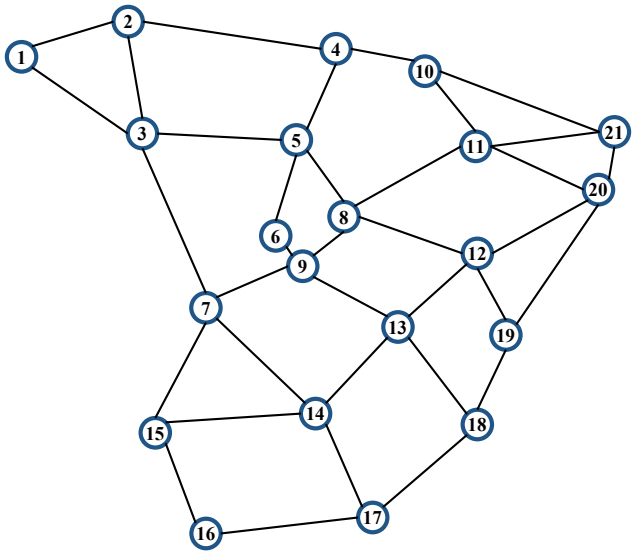

Fig. 6-11 TEL21 network topology.

For the sake of fairness in further comparisons, each core network was designed to provide approximately the same blocking performance at a given total traffic load. To this aim, the models and methods proposed in [VeKl12.1] and [RuVe13] to design core networks under dynamic traffic assumption were adapted and used to fit our pre-planning needs. Note that, although the relation between traffic load and unserved bitrate (UB) is kept pretty similar in all defined scenarios, the number of core node strongly affects the level of traffic aggregation. Namely, since the analyzed core networks cover the same geographical area and the offered traffic is the same for each network, the lower number of core nodes, the larger aggregation networks as well as the higher the flow grooming on lightpaths.

For the preplanning phase, the flows uniformly distributed among core node pairs were generated following the time-variant bandwidth distribution used in [VeKl12.2]. In particular, three different TPs, defined by their h^{min} and h^{max} bitrate values, were used to compute bitrate fluctuations of flows in time. Here, a granularity of 1h was considered. Each flow belongs to one of those TPs (with the same probability) and bitrate fluctuations are randomly chosen following a uniform distribution within an interval defined by h^{min} and h^{max}. The considered TPs with their bandwidth intervals are TP1 = [10,100], TP2 = [40,200], and TP3 = [100,400]. All traffic flows between a pair of core nodes are transported using the minimum number of lightpaths. Hence, more than one lightpath between each pair of nodes might be needed. Furthermore, the lightpaths' maximum capacity was limited to 100 Gb/s and to 400 Gb/s for the completeness of our study and the QPSK modulation format was assumed.

The preplanning problem of finding the initial route and SA for each pair of generated traffic matrix and core network so as to minimize the amount of blocked bitrate was solved using the algorithms proposed in [Kl13] for each of the proposed elastic policies. The solutions of the preplanning phase, i.e., the routing and the initial SAs of established lightpaths, were used as input data for our elastic network simulator.

Table 6-5 Details of network topologies

	Nodes	**Links**	**Nodal degree**
TEL21	21	35	3.33
TEL60	60	90	3.0
TEL100	100	125	2.5

For evaluation purposes we developed an ad-hoc event-driven simulator in OMNET++ running the algorithms presented in Section 6.3.3. Similarly as for the planning phase, a set of flows following the TP defined above were configured. For each flow, the requested bitrate varies randomly ten times per hour, i.e., a finer

and randomly-generated granularity both in time and bitrate is considered in the simulator. Hence, bitrate variations appear randomly in the system where a grooming module ensures that all the bitrate can be served. If either more resources are needed for a lightpath or resources can be released, the grooming module of the aggregation network requests the core network to perform an elastic operation on that lightpath. To this end, algorithms in Table 6-3 and Table 6-4 are executed provided that either the *Semi-Elastic* or the *Elastic* policies are used; no lightpath adaptation algorithm is run when the *Fixed* policy is considered.

In our simulations, the optical spectrum width was set to 1.2 THz and the slice width was fixed to 3.125 GHz (spectral granularity of 6.25 GHz).

B. Efficiency of SA policies

Firstly, we analyze the efficiency of SA policies in the considered core network scenarios assuming different lightpath capacity limits (100 Gb/s and 400 Gb/s). We recall that for the largest core network (i.e., TEL100) we have the lowest level of traffic aggregation and, as a consequence, the highest variability of traffic offered to the core.

Fig. 6-12 illustrates the accumulated percentage of UB as a function of the offered traffic load represented by the average load for each core network and lightpath capacity limit.

We observe that when the aggregation level is high (TEL21) and lightpath capacity is limited to 100 Gb/s, all the SA policies offer similar performance. The rationale behind that is in the low traffic variability in this scenario and, therefore, the network does not benefit from adaptive SA. Furthermore, since the volume of aggregated traffic is large and the lightpath capacity is low, a large number of lightpath connections are established between each pair of aggregation networks where a relatively large fraction of these lightpath is always saturated. Hence, elasticity is not required in that scenario. As soon the aggregation level is gradually increased in TEL60 and TEL100, the benefits of elasticity also increase, although they are rather limited.

In contrast, the advantages of elasticity clearly appear when the capacity of the lightpaths is increased to 400 Gb/s. Starting from the limited benefits observed when the aggregation level is high (TEL21), UB remarkably decreases when that level is reduced in TEL60 and especially in TEL100.

Regarding SA policies, in all the analyzed scenarios, the *Elastic* SA policy outperforms the *Semi-Elastic* SA policy which, on the other hand, performs better than *Fixed* SA.

In Table 6-6, the performance gain of elastic SA policies with respect to the fixed SA for the network load which results in 1% of UB is reported in detail. Interestingly, *Elastic* SA allows serving up to 21% of traffic more than *Fixed* SA in

the TEL100 scenario with 400 Gb/s lightpath capacity limit. In the same scenario, the gain when using the *Semi-Elastic* SA policy is just over 9%.

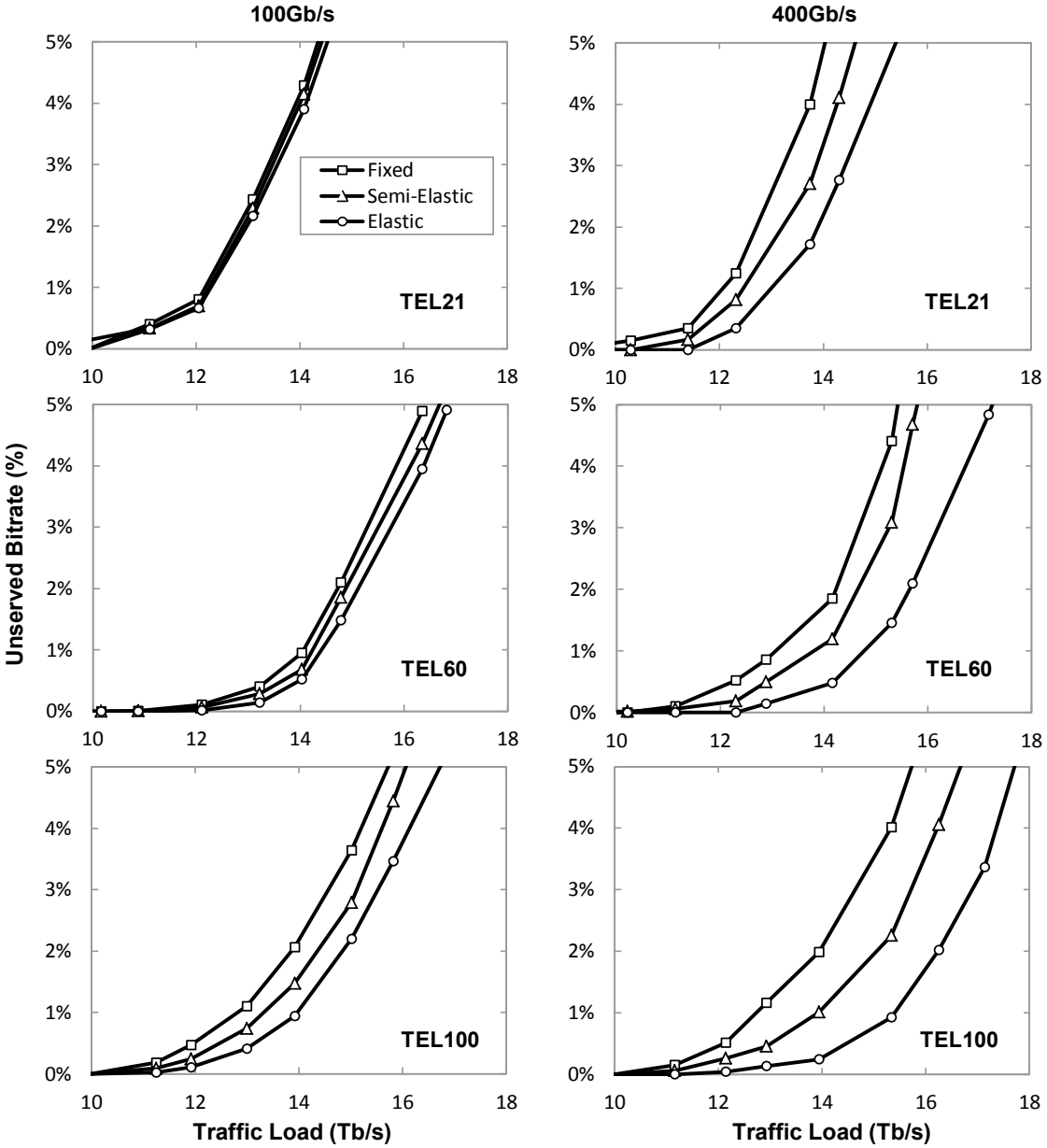

Fig. 6-12 Percentage of unserved bitrate against the traffic load for TEL21 (top), TEL60 (middle) and TEL100 (bottom) networks when the capacity of the lightpaths is limited to 100 Gb/s (left) and 400 Gb/s (right).

Table 6-6 Gain of adaptive SA policies vs. fixed SA at 1% of unserved bitrate

Max. Capacity	SA Policy	TEL21	TEL60	TEL100
400 Gb/s	**Elastic**	7.73%	12.99%	20.99%
	Semi-Elastic	3.25%	5.67%	9.34%
100 Gb/s	**Elastic**	0.92%	2.47%	8.83%
	Semi-Elastic	0.60%	1.24%	3.85%

C. Study of lightpath variability

It is interesting analyzing the relationship between traffic aggregation level and the variability of each lightpath along the time. To this end, the set of operating lightpaths is divided into three subsets, named L1 to L3, depending on the degree of bitrate variations in time:

- L1 contains all those lightpaths whose bitrate variations are almost unappreciable, and thus, the amount of spectrum resources is constant. This is mainly the case of full-loaded lightpaths grooming several demands, where statistical multiplexing keeps constant the total bitrate at maximum level.
- L2 includes lightpaths where bitrate variations are dramatic within some range of minimum and maximum values. While aggregated traffic can reach the maximum bitrate at a certain time, a minimum amount of traffic is always present.
- L3 contains those lightpaths where the highest relative bitrate fluctuations are observed, changing instantly from not being used to a peak where several spectrum slices are required.

In Table 6-7, the distribution of lightpaths belonging to each of the above defined subsets is detailed. These results are consistent with the discussion carried out in the analysis of the efficiency of SA policies and similar arguments can be used.

Firstly, there is a higher percentage of lightpaths of type L1 in the scenario with low lightpath capacity limit (100 Gb/s) compared to the scenario where 400 Gb/s lightpaths are allowed. This fact considerable reduces the amount of lightpaths needing elasticity, and thus, the performance of the fixed policy is close to the elastic SA policies. That percentage decreases if the aggregation level is decreased (see TEL100 vs. TEL21).

Secondly, when 400 Gb/s lightpaths can be used, virtually all lightpaths belong to set L2, being just a small fraction of lightpath in set L3. Hence, lightpath capacity

is always subject to changes, there are no saturated lightpaths, and only some lightpaths experience high capacity variations. On the other hand, the traffic fluctuations of lightpaths when the limit of 100 Gb/s is applied are much higher (see the percentage of lightpaths in set L3); especially, in the TEL100 scenario in which traffic aggregation is low and traffic variability is higher.

Table 6-7 Distribution of lightpaths according to the level of variability

Max. capacity	Lightpath type	TEL21	TEL60	TEL100
100 Gb/s	**L1**	19.0%	15.2%	14.3%
	L2	34.9%	41.3%	42.9%
	L3	46.2%	43.5%	42.9%
400 Gb/s	**L1**	0.0%	0.0%	0.0%
	L2	94.8%	98.0%	100.0%
	L3	5.2%	2.0%	0.0%

6.4 Conclusions

To satisfy connectivity requirements for dynamic inter-DC connectivity and to convey large amounts of data, in this chapter, we proposed a distributed CSO, which can send connection requests to ABNO, and two connectivity models: dynamic and dynamic elastic. A static approach, where inter-DC connections are over-provisioned to satisfy demand at any time, and the two dynamic connectivity models were compared.

Even with dynamic connectivity, some resource over-provisioning still need to be done to guarantee that resources are available when needed. Aiming at improving the performance of dynamic connectivity, we proposed to add elastic operations on already established connections to allow DC resource managers doing retries when not enough resources are available at connection's set-up time.

Illustrative results showed that dynamic elastic connectivity improves the performance of the dynamic one in scenarios where the physical optical network is shared by several services, so competence to use network resources arises. Finally, it was shown that dynamic connectivity could entail bitrate savings as high as 60% with respect to static connectivity. This fact becomes remarkable for DC operators willing to reduce OPEX and paves the way to devise scheduling algorithms that might use cheaper inter-DC connectivity to improve service performance, reduce OPEX, e.g., by minimizing energy consumption.

Then, because of the performance improvement showed when the dynamic elastic connectivity model is considered and since connection requests can be sent to ABNO dynamically, we focused on the study of the network ability to manage elastic operations once in operation aiming at satisfying dynamic elastic connectivity requirements. To that end, we focused on dynamic adaptation of lightpath connections, by means of elastic SA, in a flexgrid-based elastic optical network with time-varying traffic demands. In details, we addressed a scenario in which an EON core network connects a number of IP/MPLS metro area networks performing traffic aggregation. Although a large core network connecting a large set of small aggregation networks is a cost-effective solution, if the core network is extended towards the edges and the number of aggregation networks is increased, then the level of traffic aggregation is decreased and the variability of traffic to be carried over the core network is higher.

To deal with that variability, we have proposed to make use of elastic SA, which translates to the adaptation of spectrum allocated to lightpath connection in response to changes in traffic demands. We have analyzed two SA policies, namely, a symmetric SA policy (referred to as *Semi-Elastic* SA) and asymmetric SA policy (referred to as *Elastic* SA), and compared their performance against a *Fixed* SA policy, which does not allow for spectrum changes. For each elastic SA policy, we developed a dedicated lightpath adaptation algorithm. The evaluation has been performed in a simulation environment assuming different network scenarios, which are characterized by different levels of traffic aggregation and lightpath capacity limits. We have shown that the effectiveness of lightpath adaptation in dealing with time-varying traffic highly depends on both the aggregation level and the maximum lightpath capacity. In a network with low traffic aggregation, the best performance gap of about 21% (vs. the *Fixed* SA policy) has been achieved for *Elastic* SA operating with high lightpath capacity limits.

It was shown that when core networks are extended towards the edges, the capacity of the lightpaths could be limited to deal with longer distances. In such scenarios, the effectiveness of elasticity is rather limited. Therefore, as a final remark, those networks where the majority of lightpaths are some few hundred kilometers long (e.g., national European core networks) can currently take advantage of elasticity. On the opposite, additional research to extend the reach of 400 Gb/s (or even higher) lightpaths is needed so as to larger core networks can benefit also from elasticity.

In the next chapter, we continue studying connectivity requirements and propose a notification-based model against the polling-based, dynamic elastic, connectivity model.

Chapter 7

Polling-based and notification-based connectivity models

In this chapter, we extend the ELFADO problem presented in Chapter 4. Since green energy availability greatly depends on weather conditions, we present a statistical model to improve green solar energy availability estimation accuracy and we use it in a MILP formulation to model the Stochastic-ELFADO (STC-ELFADO) problem, computing optimal VM placement. As described in the previous chapter, dynamic and dynamic elastic connections can be requested to an intelligent control plane, e.g., ABNO. However, competence for network resources could lead to connections capacity being reduced or even blocked at requesting time, resulting in a poor cloud performance. Even though DC resource managers can either give connection request retries to increase the bitrate of already established connections or set-up new ones, network resource availability is not guaranteed at requesting time to satisfy the requested capacity, since both cloud and network strata are operated independently.

A *polling-based* and a *notification-based* models are compared in scenarios where the STC-ELFADO problem is solved. Noticeable costs savings in OPEX are observed when notification-based connectivity is considered.

7.1 Statistical model for green solar energy availability estimation

We assume a scenario where federated DCs interconnected by a flexgrid-based network are strategically placed around the globe to provide worldwide high QoE services. Each DC obtains its required energy from *green* and *brown* sources. In

Chapter 4, we formally stated the Elastic Operations in Federated DC for Performance and Cost Optimization, ELFADO, problem and studied two approaches to orchestrate a DC federation: distributed and centralized. We showed that the centralized approach takes full advantage of green energy availability in the federated DCs helping to minimize energy and communication costs (cost per GB transmitted through optical connections) while ensuring QoE. Thus, here we assume a centralized federation orchestrator computing periodically the global optimal placement for all the VMs in the federated DCs.

Because weather conditions severely impact green solar energy availability, in this section, we present a model to estimate the amount of green solar energy available in each location as a function of the specific time period and the weather conditions.

In line with Chapter 4, let us assume that green solar energy covers some proportion β_d of the maximum energy consumption of DC d (Max_Energy_d), i.e., in the scenario where all servers are at the highest load, all switches are active, and the *PUE* value is the highest one. The amount of green energy available in DC d at period t can be estimated as $g_d(t) = \beta_d \cdot \delta_d(t) \cdot Max\_Energy_d$, where δ_d(t) is the normalized availability of green energy as a function of the time of day in the location of DC d.

Therefore, once the VMs to be placed in each DC in the next period are known, the federation orchestrator can compute precisely the amount of workload in each DC, compute the green energy available and thus, the optimal VM placement.

Note that the accuracy of $\delta_d(t)$ highly impacts the accuracy of the green solar energy availability estimation $g_d(t)$. However, the value of $\delta_d(t)$ is related not only to the current season, but also to the weather conditions, which makes it hard to be aggregated into a single value. Plots in Fig. 7-1 illustrate this fact for a DC placed in Berlin (data obtained from [USEIA], [USDOE], [Ki04]). As observed, all the days in the same week behave similarly, i.e., peak values are at the same time and around a mean value (seasonal effect), although steep changes from one hour to the next and between two consecutive days (weather effect) can be clearly observed. In view of this, we decided characterizing the green energy availability as a function of three variables: the day of the year (index i), the hour of the day (index j), and the weather conditions (modeled as *weather levels*, index k). Therefore, we use $\delta_d(i,j,k)$ hereafter.

Fig. 7-2a shows observed δ_d values along a year at 2 p.m.; i.e., data pairs $\{i, \delta_d(i,2,*)\}$, where * represents any value of the variable (weather level in this case). Five weather levels with different markers are plotted; each level shows a similar shape but different scale. In light of Fig. 7-2a, it can be concluded that weather conditions can be likely predicted as weather levels. Hence, our proposed statistical model to estimate $\delta_d(i,j,k)$ consists of a set of curves, one for each weather level k, as a function of the day i. In addition, a set of coefficients to scale predictions for

every hour j are necessary. With this approach, multiple stationary effects (daily, yearly) can be easily managed.

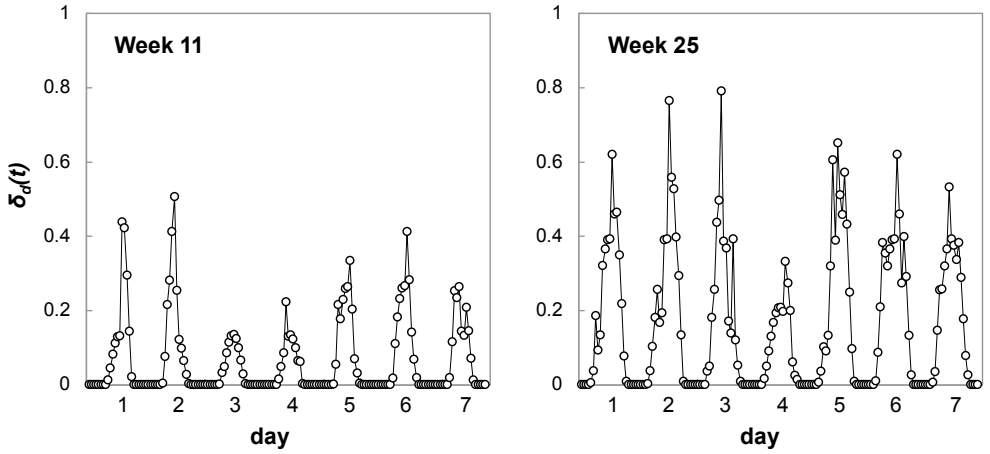

Fig. 7-1 $\delta_d(t)$ as a function of the day for two different weeks.

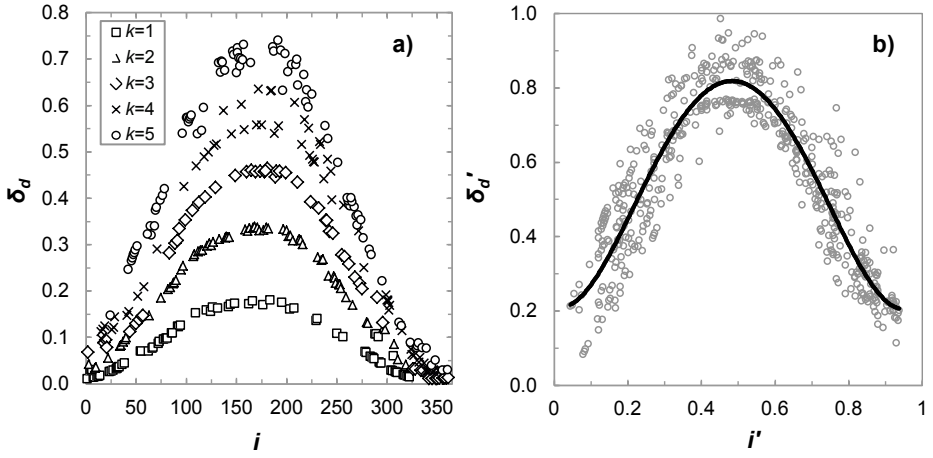

Fig. 7-2 δ_d as a function of the day (at 2 p.m.) (a) and δ_d' for k=3 (b).

Before applying curve fitting to each weather level, data must be properly normalized. To that end, a tuple with three coefficients is needed for every observation belonging to the same hour j (i.e., $\delta_d(*,j,*)$): $<firstDay_j, lastDay_j, peak_j>$, where the first two coefficients store the range of days where δ_d takes non-zero values and the last coefficient stores the maximum δ_d observed. The tuple for data in Fig. 7-2a is <1, 365, 0.747>. Note that those tuples are remarkably different for different hours, e.g., the tuple at 7 a.m. for the same location is <88, 294, 0.255>.

Next, data pairs $<i, \delta_d(i,*,*)>$ need to be normalized following equations (7.1) and (7.2), obtaining normalized data pairs $<i', \delta_d(i',*,*)'>$ in the continuous range [0, 1]. For illustrative purposes, let us normalize the observation in Fig. 7-2 for k = 3 and i=100, with δ_d=0.351. From the tuple of coefficients characterizing data in Fig. 7-2a and equations (7.1) and (7.2), i' = (100-1)/(365-1) = 0.27 and δ_d' = 0.351/0.747 =

0.47. Normalized data pairs for k=3 are represented in Fig. 7-2b, together with the curve that better fits the average trend. After a preliminary analysis, we concluded that a degree 4 polynomial is enough accurate for fitting. Therefore, the statistical model shown in (7.3) consists in a polynomial with 5 coefficients that receives as input the normalized day, the hour, and the weather level and returns the normalized green energy availability. Since the polynomial does not perfectly match all the observations, a Gaussian error ε centered in 0 with variance σ^2 is considered as part of the model.

$$i' = \frac{i - firstDay_j}{lastDay_j - firstDay_j} \tag{7.1}$$

$$\delta_d(i', j, *)' = \frac{\delta_d(i', j, *)}{peak_j} \tag{7.2}$$

$$\delta_d(i', j, k)' = \sum_{n=0..4} b_{nk} \cdot [i']^n + \varepsilon(0, \sigma^2) \tag{7.3}$$

Using the proposed statistical model, $\delta_d(i,j,k)$ for a given day i, hour j, and weather level k can be estimated as follows:

- the day is first normalized using (7.1); thus, obtaining *i';*
- $\delta_d(i',j,k)$' is estimated using (7.3);
- $\delta_d(i,j,k)$ is obtained from $\delta_d(i',j,k)$' using (7.2) properly reverted.

As a result, the probability density function of the green energy availability follows a Gaussian distribution with mean $\delta_d(i,j,k)$ and variance σ^2.

Note that the expected weather level is an input of the model that might be estimated by other means. When estimations are done between consecutive hours, it is clear that a time series model for predicting next weather level from past observations is a valuable option. However, in this work, we take advantage of a basic but time-dependent model consisting in a matrix of transition probabilities between levels in consecutive hours, estimated from the available data. The matrix stores the probability that weather changes from level k to level k' in the next hour j + 1; an example of transition probabilities matrix is shown in Section 7.4.

In the next section, the proposed methodology is used to generate instances for solving a new approach of the centralized ELFADO problem defined in Chapter 4 that we call stochastic ELFADO (STC-ELFADO).

7.2 Stochastic ELFADO formulation

For the sake of completeness, the ELFADO problem statement is reproduced in the following:

Given:

- a set of federated DCs D,
- the set of optical connections E that can be established between every pair of DCs,
- a set of VMs $V(d)$ in each DC d,
- a set of client locations L, where nl_l is the number of users in location l to be served in the next period,
- PUE_d, brown energy cost c_d, and green energy available in DC d for the next period,
- the data volume u_v and the number of cores nc_v of each VM v,
- energy consumption of each server as a function of the load and each switch,
- the QoE q_{ld} perceived in location l when served from a VM placed in DC d,
- a threshold th_v for the QoE required at any time for accessing the service in VM v.

Output: the DC where each VM will be placed in the next time period;

Objective: minimize energy and communications cost for the next time period ensuring the performance objective for each service.

We propose to solve the stochastic approach for the ELFADO problem by means of discrete probability scenarios (see [Sh09]). In view of the statistical model presented above, a scenario for each weather level k can be easily obtained: the green energy available is estimated from the polynomial model and its probability comes from the transition probability from the current to the next k level. Then, the brown energy needed at each scenario is different and the cost of each scenario is weighted in the objective function by the transition probability. In line with [Go12], we assume the cost of green energy is zero to prioritize it against brown energy.

In addition, we consider that each DC is connected to the flexgrid inter-DC network through a switch equipped with bandwidth variable transponders of a given capacity (e.g., 100 Gb/s). To limit the total time of the migration process (*maxMigrationTime*), we limit the amount of data that can be transferred by a given optical connection in *maxMigrationTime*, which depends on the effective connection throughput (i.e., capacity without headers).

The notation needed for the MILP formulation is as follows:

Sets:

D Set of federated DC, index d.

E Set of optical connections, index e.

V Set of VMs, index v.

$V(d)$ Set of VMs currently placed in DC d.

L Set of client locations, index l.

$K(d)$ Set of probability scenarios in DC d.

Users and performance:

q_{ld} QoE (in terms of delay in ms) experienced by users accessing DC d from location l.

nl_l Number of users in location l.

th_v Average QoE threshold to be guaranteed for those users accessing VM v.

DC architecture and VMs:

M Number of clusters per DC.

ns Number of cores per server.

u_v Size of VM v in GB.

nc_v Number of cores needed by VM v.

Connections:

u_e Max volume of data (GB) that can be conveyed through connection e, computed as: $\text{throughput}_e \cdot maxMigrationTime$.

c_e Cost per GB transmitted through connection e.

Energy:

PUE_d PUE for DC d.

c_d Brown energy cost per kWh in DC d.

$w(\cdot)$ Energy consumption of element $(\cdot)$, $w(\cdot) = P(\cdot) \cdot 1\text{h}$.

g_{dk} Amount of green energy available in DC d under probability scenario k.

p_{dk} Probability of scenario k in DC d.

The decision variables are:

x_{vd} Binary, 1 if VM v is placed in DC d; 0 otherwise.

y_{dk} Positive real, energy consumption in DC d in probability scenario k.

z_e Positive integer, GB to be transferred through optical connection e.

γ_d Positive integer with the number of servers operating with some load in DC d.

ρ_d Positive integer with the number of clusters switched on in DC d.

b_d Positive real, total energy consumption in DC d.

Finally, the formulation of the stochastic ELFADO based on introducing discrete probability scenarios, is as follows:

$$\left(\text{STC} - \text{ELFADO}\right) \text{minimize} \sum_{d \in D} c_d \cdot \sum_{k \in K(d)} p_{dk} \cdot y_{dk} + \sum_{e \in E} c_e \cdot z_e \tag{7.4}$$

Subject to:

$$\frac{1}{\sum_{l \in L} nl_l} \cdot \sum_{l \in L} \sum_{d \in D} nl_l \cdot q_{ld} \cdot x_{vd} \leq th_v \quad \forall v \in V \tag{7.5}$$

$$\sum_{d \in D} x_{vd} = 1 \quad \forall v \in V \tag{7.6}$$

$$\gamma_d \geq \frac{1}{ns} \cdot \sum_{v \in V} nc_v \cdot x_{vd} \quad \forall d \in D \tag{7.7}$$

$$\rho_d \geq \frac{4}{M^2} \cdot \gamma_d \quad \forall d \in D \tag{7.8}$$

$$b_d = PUE_d \cdot \left(\begin{array}{c} \frac{M^2}{4} \cdot w_{core} + \frac{M}{2} \cdot \left(w_{agg} + w_{edge}\right) \cdot \rho_d + \\ w_{server-max} \cdot \gamma_d + w_{server-idle} \cdot \left(\frac{M^2}{4} \cdot \rho_d - \gamma_d \right) \end{array} \right) \quad \forall d \in D \tag{7.9}$$

$$y_{dk} \geq b_d - g_{dk} \quad \forall d \in D, k \in K(d) \tag{7.10}$$

$$z_{e=(d_1,d_2)} = \sum_{v \in V(d_1)} u_v \cdot x_{vd_2} \quad \forall d_1, d_2 \in D, d_1 \neq d_2 \tag{7.11}$$

$$z_e \leq u_e \quad \forall e \in E \tag{7.12}$$

The objective function (7.4) minimizes the total cost for every DC in the federation, which consists of the energy costs (weighted by the probability of each scenario) and the communication costs for the VMs that are moved between DCs. Constraint (7.5) guarantees that each VM is assigned to a DC if the on-average QoE perceived by the users does not exceed the given threshold. Constraint (7.6) ensures that each VM is assigned to exactly one DC. Constraint (7.7) computes, for each DC, the amount of servers where some VM is to be placed, whereas (7.8) stores the number of clusters that will be switched on. Constraint (7.9) computes the total energy consumption in each DC (see power model in Section 2.1). Constraint (7.10) stores the brown energy consumption for each scenario in each DC computed as the difference between the effective energy consumption and the amount of green energy available in the next period in each DC. Constraint (7.11) computes the amount of data to be transferred from each DC to some other remote DC and (7.12) assures that the capacity of each optical connection is not exceeded.

7.3 DC and network orchestration

Scheduling algorithms, such as STC-ELFADO, run periodically (e.g., each hour) and compute the proper VM placement aiming to reduce energy expenditures, while ensuring the maximum migration time. It is clear that to minimize expenditures, a VM could be migrated every hour, which can be avoided by adding new constraints to the MILP formulation. Notwithstanding, the statistical model for green solar energy availability estimation, presented in Section 7.1, introduces short-term solar energy predictability that indirectly avoids abrupt changes in the prediction, preventing VMs to be continuously migrated.

VMs migrations are required to be performed within each period; more specifically, the migration process should be completed within *maxMigrationTime*. The live-migration process for each individual VM migration can be done in short times and thus, downtimes are virtually zero when both servers are in the same DC. However, when the migration process involves two different DCs, the downtime directly depends on the throughput of the path connecting the servers [Ha07] as a result of not only the delay, but also because some of the already migrated memory pages are modified in the current server (known as dirty pages). Therefore, from the DC federation viewpoint, the offered connectivity service is better when the VMs can be transferred fast to their proper locations. Nonetheless, the network operator wants to reuse its resources as much as possible, so it is required to assess a connectivity model which can fit with users and operators requirements.

To study the benefits from both viewpoints, we compare two connectivity models with elastic capabilities: *i*) polling-based, and *ii*) notification-based.

In the polling-based model, DC resource managers can either give connection request retries to increase the bitrate of already established connections or set-up

new ones, as shown in the dynamic elastic model. However, because both cloud and network strata are operated independently, network resource availability to satisfy the requested capacity cannot be guaranteed at requesting time. Therefore, we propose the notification-based model, where some network information is sent to DC resource managers; e.g., information on network resources availability on the route of a given connection. Notifications can be sent from the control plane in charge of the network to the resource managers after an event occurs; e.g., a connection tear-down. Fig. 7-3 illustrates connection operations request/response and notifications flow.

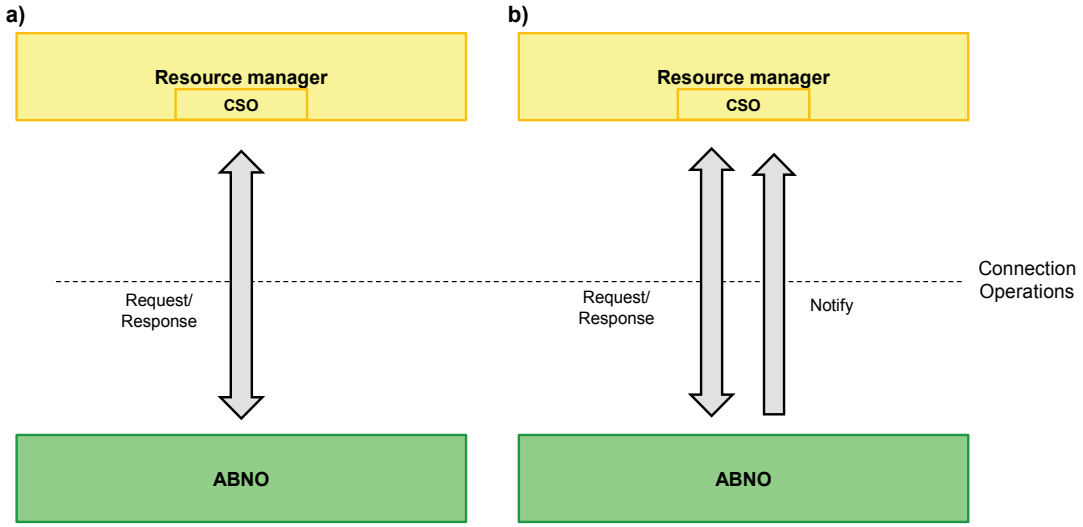

Fig. 7-3 Polling-based (a) and notification-based (b) models.

7.4 Illustrative results

In this section, we first validate the stochastic approach for the ELFADO problem comparing its performance against that of the deterministic version presented in Chapter 4. Next, we use STC-ELFADO on a worldwide scenario to compare the performance of using the *polling-based* and the notification-based connection models.

A. Scenario

For evaluation purposes, we developed resource managers in an OpenNebula-based cloud middleware emulator. The federation orchestrator solving the centralized STC-ELFADO model was implemented as a stand-alone module in Java. Federated DCs are connected to an ad-hoc event-driven simulator developed in OMNET++. The simulator implements the flexgrid network and includes the algorithm described in Chapter 6 for elastic SA.

B. Stochastic ELFADO validation

To compare the performance of the deterministic and the stochastic ELFADO approaches, we consider a European topology consisting of 10 DCs, each with a number of clusters M=10. For each location, a statistical model of the form presented in Section 7.1 is estimated from the data available in [USEIA], [USDOE], [Ki04]. Table 7-1 and Table 7-2 show the parameters used for data normalization and the coefficients of the polynomials, respectively, for a DC placed in Berlin. Note that the range between *firstDay* and *lastDay* as well as that for the *peak* increase for mid-day hours. In addition to this, Table 7-3 stores the weather transition probabilities between consecutive hours. Highest probabilities are observed in the diagonal and cells close to it, being this representative of a natural daily weather evolution, where changes during short time periods are usually small. The validation process of this model allows concluding that most of the 95% of the predictions provide residuals within a common accepted standard error of ±2 ($\sigma^2 \approx 0.002$). The model provides similar performance for all the considered locations.

Table 7-1 Normalization parameters

	j							
	5	**6**	**7**	**8**	**9**	**10**	**11**	**12**
$\boldsymbol{firstDay_j}$	139	109	88	56	10	1	1	1
$\boldsymbol{lastDay_j}$	216	256	294	328	365	365	365	365
$\boldsymbol{Peak_j}$	0.0106	0.0987	0.2548	0.421	0.5764	0.7094	0.7997	0.8516
	13	**14**	**15**	**16**	**17**	**18**	**19**	**20**
$\boldsymbol{firstDay_j}$	1	1	1	2	36	70	103	136
$\boldsymbol{lastDay_j}$	365	365	365	365	317	287	261	235
$\boldsymbol{peak_j}$	0.8556	0.8138	0.7283	0.6047	0.4564	0.2946	0.1362	0.0256

For evaluation, a total load of 5,000 VMs is distributed among DCs. Under the stochastic approach, one scenario per weather level k and DC were considered (50 scenarios in total), with expected g_{dk} and p_{dk} obtained from the proposed statistical model. Under the deterministic approach, only one scenario per DC with the expected average green energy availability was assumed.

Aiming at evaluating the quality of the obtained solutions, we computed the *Expected Value of Perfect Information* (*EVPI*), which assumes that the real amount

of green energy is perfectly known in advance, being this the benchmark for computing energy costs and savings.

Table 7-2 b_{nk} polynomial coefficients

	$n = 4$	$n = 3$	$n = 2$	$n = 1$	$n = 0$
$k = 1$	3.737	-7.443	3.759	-0.054	0.016
$k = 2$	7.175	-14.136	7.166	-0.208	0.075
$k = 3$	9.227	-17.895	8.661	0.001	0.124
$k = 4$	11.283	-21.782	10.319	0.247	0.133
$k = 5$	11.847	-21.920	8.497	1.823	-0.049

Table 7-3 Weather probability transitions (j→j+1)

	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$
$k = 1$	0.7298	0.1992	0.0498	0.0151	0.0060
$k = 2$	0.2427	0.4976	0.1445	0.0689	0.0463
$k = 3$	0.1069	0.3152	0.3261	0.1793	0.0725
$k = 4$	0.0712	0.1346	0.2000	0.3154	0.2788
$k = 5$	0.0054	0.0261	0.0534	0.1786	0.7364

We solved the ELFADO problem using CPLEX at every hour of 120 consecutive days of spring and summer time (2,880 problem instances in total). To estimate likely costs of each ELFADO solution, a simulation was run to obtain green energy availability from the probability distribution obtained with the $\delta_d(t)$ statistical model; thus, emulating a real-life behavior; costs were eventually computed from that simulated green energy values.

Fig. 7-4a shows the energy cost increment with respect to the cost of EVPI solutions as a function of daytime hours. As clearly observed, the main differences between strategies appear in central hours, when $\delta_d(t)$ takes the highest values and when fluctuations induced by weather variations are more evident. The stochastic approach obtains solutions much closer to EVPI compared to the deterministic scheme. Specifically, a maximum difference circa 5% with respect to EVPI is observed. The benefits of this stochastic approach can be also observed in Fig. 7-4b, which plots cumulative cost increment; a promising 30% of the total cost increment with respect to that provided by the deterministic approach is achieved by only

using few discrete probability scenarios. In view of this, it is clear that the stochastic approach for solving ELFADO provides high-quality solutions with a computational complexity similar to that of the deterministic approach.

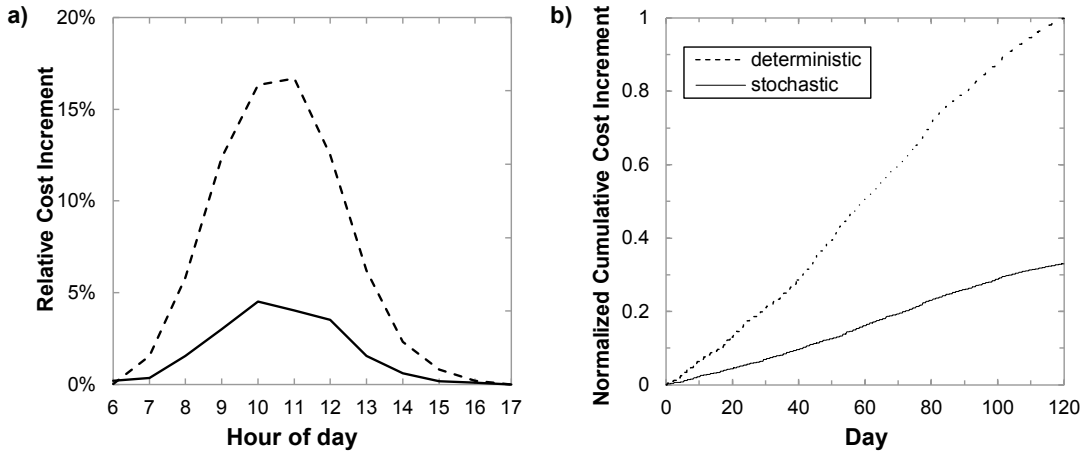

Fig. 7-4 Energy cost increment w.r.t. EVPI: relative per hour (a), cumulated per day (b).

C. Polling-based and notification-based performance evaluation

After the stochastic approach for ELFADO problem has been validated, we move to a global 11-node topology, where locations are used as source for U2DC traffic collecting user traffic towards the set of DCs. The set of DCs consists of five DCs strategically located in Taiwan, India, Spain, and Illinois and California in the USA (Fig. 7-5). A global telecom operator provides optical connectivity among DCs, which is based upon the flexgrid technology. The number of users in each location was computed considering Wikipedia's audience by regions that was scaled and distributed among the different locations in each region. Latency was computed according to Verizon's data [VERIZON].

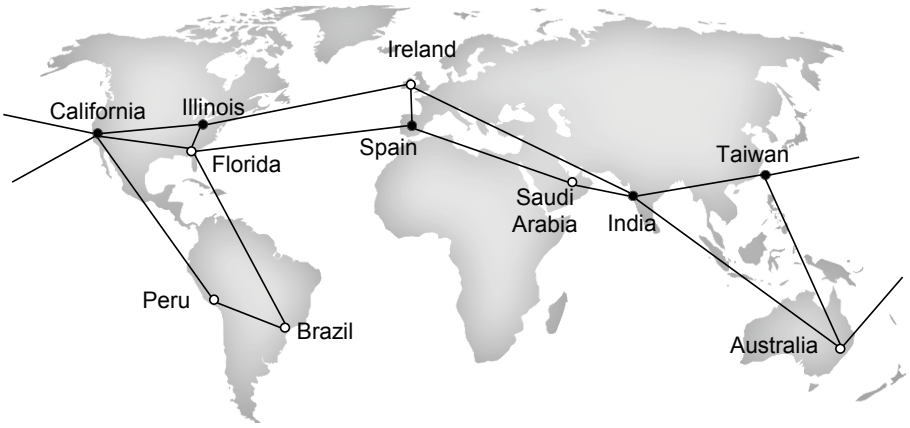

Fig. 7-5 Worldwide scenario considered in our experiments.

In line with [Fa08], DCs are dimensioned assuming a fat-tree topology with a maximum of $M = 48$ clusters with two levels of switches and $M^2/4 = 576$ servers each. The number of VMs was set to 35,000, with individual image size of 5 GB; we assume that each VM runs in one single core. Initially, all VMs are distributed among all DCs equally.

An integer number of clusters is always switched on, to support the load assigned to the DC; those servers without assigned load remain active and ready to accommodate spikes in demand. Green cover was set to ensure, at the highest green energy generation time, a proportion of energy β_d when all VMs run in DC d.

We consider that each DC is connected to the flexgrid inter-DC network through a switch equipped with 100 Gb/s bandwidth variable transponders. Therefore, the actual capacity of optical connections is limited to that value. To compute the real throughput, we consider headers for the different protocols, i.e., TCP, IP, and GbE. The capacity requested for optical connections was computed to guarantee that VM migration is performed in less than 40 minutes. Note that transponders capacity together with maximum scheduled transfer time limit the amount of VMs that can be moved between two DCs to about 4,620.

Finally, a dynamic network environment was simulated for the scenario under study, where background incoming connection requests arrive following a Poisson process and are sequentially served without prior knowledge of future incoming connection requests. Background traffic competes with the one generated by the federated DCs for network resources. The bitrate demanded by each background connection request was set to 100 Gb/s. The holding time of connections is exponentially distributed with the mean value equal to 2 hours. Source-destination pairs are randomly chosen with equal probability among all nodes. Different values of offered network load were considered by changing the arrival rate while keeping the mean holding time constant.

Fig. 7-6 plots daily energy (Fig. 7-6a) and communication (Fig. 7-6b) costs as a function of the normalized background traffic intensity. We observe a clear increasing trend when the background traffic increases, as a consequence of connections' initial capacity decreases from 55 Gb/s to only 12 Gb/s on average. To try to increase that limited initial connections' capacity, elastic capacity increments need to be requested. The results obtained when each connectivity model is applied are however different. Both models behave the same when the background traffic intensity is low or high, which is a consequence of the percentage of VMs that could not be migrated in the scheduled period (see Fig. 7-7a). When the intensity is low, there are enough resources in the network, so even in the case that elastic connection operations are requested, both models are able to perform scheduled VM migration in the required period. When the background intensity is high, connections requests are rejected or are established with a reduced capacity that is unlikely modified. As a result, a high percentage of scheduled VMs migrations could not be performed.

However, when the background load increases without exceeding 5% of total blocking probability, the behavior of the analyzed connectivity models is different; the proposed notification-based model provides a constant energy and communication costs until the normalized background load is greater than 0.4, in contrast to the remarkable cost increment provided using the *polling-based* model. In fact, costs savings as high as 20% and 45% in energy and communications, respectively, are obtained when the notification-based model is applied with respect to those of the *polling-based*. When the normalized background load increases from 0.4, the lack of resources starts affecting also the notification-based model and, although costs savings reach their maximum for a load of 0.5, energy and communication costs start increasing and relative savings decreasing.

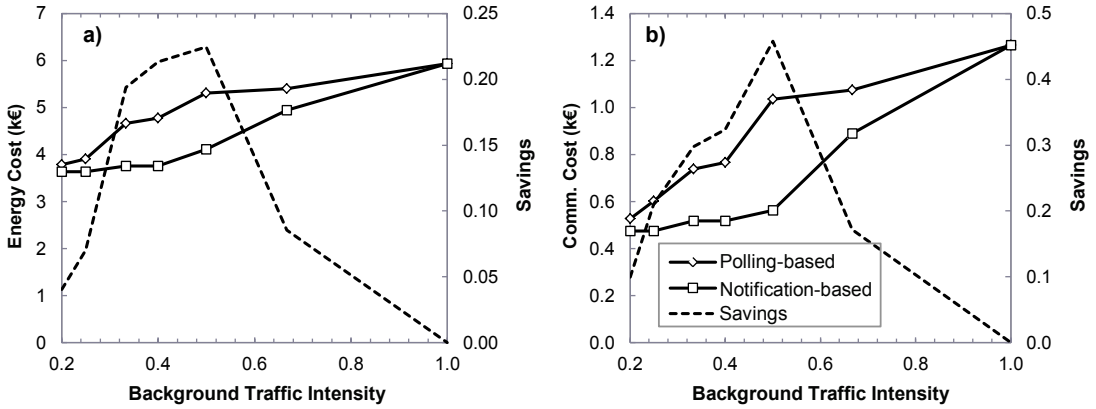

Fig. 7-6 Daily energy cost (a) and communications cost (b).

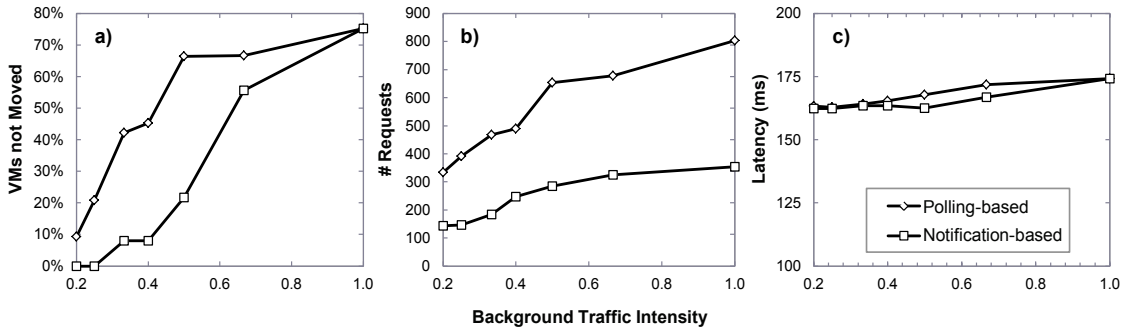

Fig. 7-7 Percentage of VMs not moved as first scheduled (a), number of connection requests (b), and latency experienced by users (c).

It is also interesting to see the total number of requests generated when each connectivity model is used. Fig. 7-7b plots the amount of requests for set-up, elastic capacity increment, and tear down that arrive to the ABNO. When the polling-based model is used, the number of requests is really high compared to that number under the notification-based model. However, since the requests are generated by the DC resource managers without any knowledge of the state of the resources, the majority of those requests are blocked as a result of lack of

resources. Such high utilization of the network resources is the target for the network operator. In contrast, in the notification-based model, elastic capacity increment requests are generated based on the information received when some resources in the route of established connections have been released and elastic capacity operation could be successfully applied. In this case, the amount of requests is much lower but many of them are successfully completed (although few can be also blocked). Regarding latency, both models are able to provide similar performance, as shown in Fig. 7-7c. This fact, however, is a result of the scheduler algorithm that focuses at guaranteeing the committed QoE.

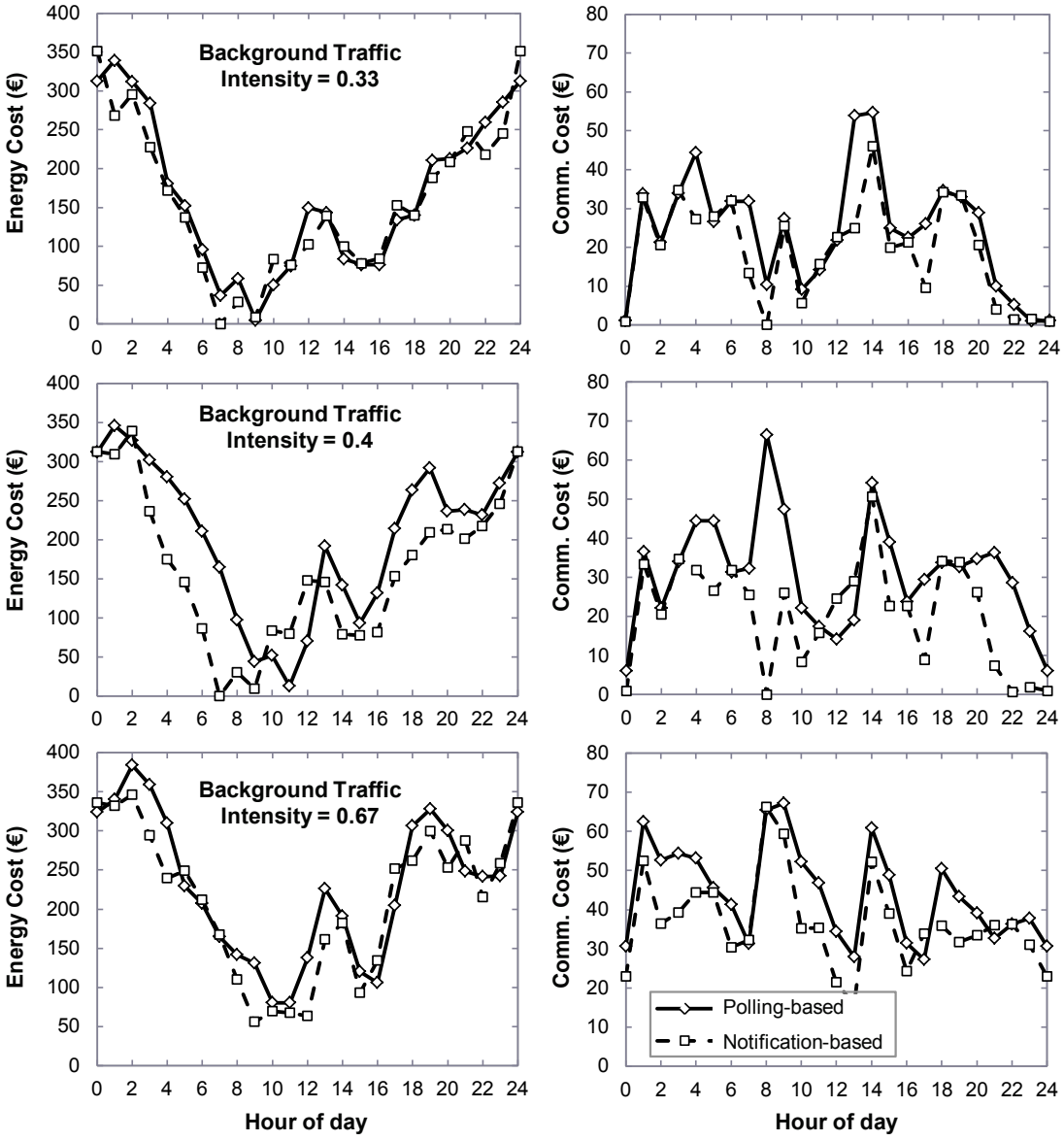

Fig. 7-8 Hourly costs for several background traffic intensities.

Finally, Fig. 7-8 illustrates hourly variation in the energy and communication costs when the polling-based and the notification-based models are applied for three different background traffic loads. The behavior of both models is basically the same and slight hourly energy cost savings can be appreciated, although they are clearly evident for the intermediate load. In contrast, there are some periods with a totally different behavior between polling-based and the notification-based models, especially in the intermediate load. That is as a consequence of that VMs can be placed in those locations so as to minimize cost in the notification-based model so no new migrations are required, whereas massive migrations need to be done *polling-based*, which further increases communications needs.

7.5 Conclusions

A statistical model has been proposed for improving the accuracy of green energy availability estimation in a given location. Taking advantage of the proposed statistical model and extending the centralized ELFADO problem, the STC-ELFADO problem has been presented; the statistical model has been used in the STC-ELFADO MILP formulation to compute optimal VM placement in a DC federation. Both, the statistical and the deterministic approaches have been compared and results showed that the estimation accuracy in the STC-ELFADO results in energy costs reduction.

Moreover, we proposed two connectivity models: polling-based and notification-based. The polling-based model needs periodical retries requesting to increase connection's bitrate, which do not translate into immediate bitrate increments and could have a negative impact on the performance of the network control plane. For this very reason, in this chapter, we proposed a new connectivity model based on notifications, where information from network resources availability is sent to resource managers.

Polling-based and notification-based connectivity models were compared in realistic scenarios. Specifically, energy and communication costs and QoE in a DC federation were analyzed. From the results, we observed that when the network operates under low and medium traffic load costs savings as high as 20% and 45% in energy and communications, respectively, can be obtained when the notification-based model is applied with respect to those of the polling-based. Besides, both connectivity models allow scheduling algorithms to provide the committed QoE.

The proposed notification-based model opens the opportunity to network operators to implement policies so as to dynamically manage connections' bitrate of a set of customers and fulfill simultaneously their SLAs.

In the next chapter, we go in depth and propose an architecture to support the notification-based model.

Chapter 8

Application service orchestrator for transfer-based DC interconnection

Although DC resource managers can request optical connections and control their capacity, in the last chapter we saw that a notification-based connectivity model brings noticeable costs savings. That connectivity model is useful when connectivity can be specified in terms of volume of data and completion time (we call this transfer-based connectivity).

To increase network resource availability, in this chapter, we propose an Application Service Orchestrator, ASO, as an intermediate layer between the cloud and the network. In this architecture, DC resource managers request data transferences using an application-oriented semantic. Then, those requests are transformed into connection requests and forwarded to the ABNO in charge of the transport network. The proposed ASO can perform elastic operations on already established connections supporting transferences provided that the committed completion time is ensured. The polling-based and the notification-based models are compared in a scenario considering ASO.

Then, we propose using not only elastic operations incrementing transfer-based connections' capacity but also scheduling elastic operations both incrementing and decrementing transfer-based connections' capacity. Transfer-based requests from different DC operators received by ASO are routed and resource allocation scheduled to simultaneously fulfill their SLAs, which becomes the Routing and Scheduled Spectrum Allocation (RSSA) problem. An ILP model is proposed and an algorithm for solving the RSSA problem in realistic scenarios is designed. Results

showing remarkable gains in the amount of data transported motivate its use for orchestrating DC interconnection.

8.1 Application Service Orchestrator (ASO)

To provide an abstraction layer to the underlying network, a new stratum on top of the ABNO, the ASO, could be deployed (Fig. 8-1). The ASO implements a northbound interface to request transfer operations. Those applications' operations are transformed into network connection requests. The northbound interface uses application-oriented semantic, thus liberating application developers from understanding and dealing with network specifics and complexity.

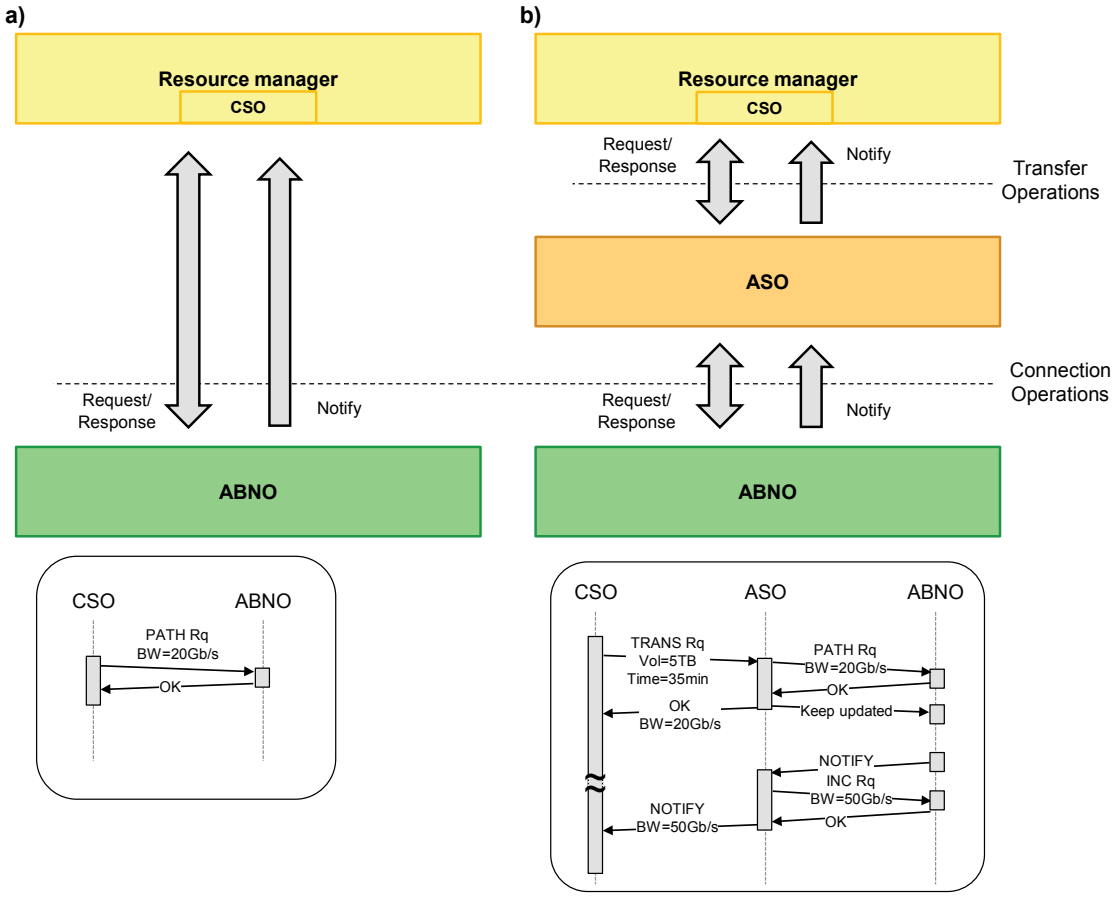

Fig. 8-1 Control architecture where the ABNO interfaces resource managers (a). Proposed architecture where the ASO offers an application-oriented semantic interface (b).

As an example of that paradigm, we propose an operation where applications request transfers using cloud middleware native semantic, i.e.:

- source DC,

- destination DC,
- amount of data to be transferred,
- and completion time.

The ASO is in charge of managing inter-DC connectivity; if not enough resources are available at requesting time, notifications (similar to interruptions in computers) are sent from the ABNO to the ASO each time specific resources are released. Upon receiving a notification, the ASO takes decisions on whether to increase the bitrate associated to a transfer. Therefore, we have effectively moved from a polling-based connectivity model where applications request connection operations to a notification-based connectivity model where transfer operations are requested to ASO, which manages connectivity.

8.1.1 Scenario

We consider a scenario where a follow-the-work strategy for VM migration is implemented; VMs are moved to DCs closer to the users, reducing the user-to-service latency. Cloud scheduling algorithms run periodically taking VM migration decisions. Once the set of VM to be migrated and DB synchronization needs are determined, cloud management performs those operations in collaboration with inter- and intra- DC networks.

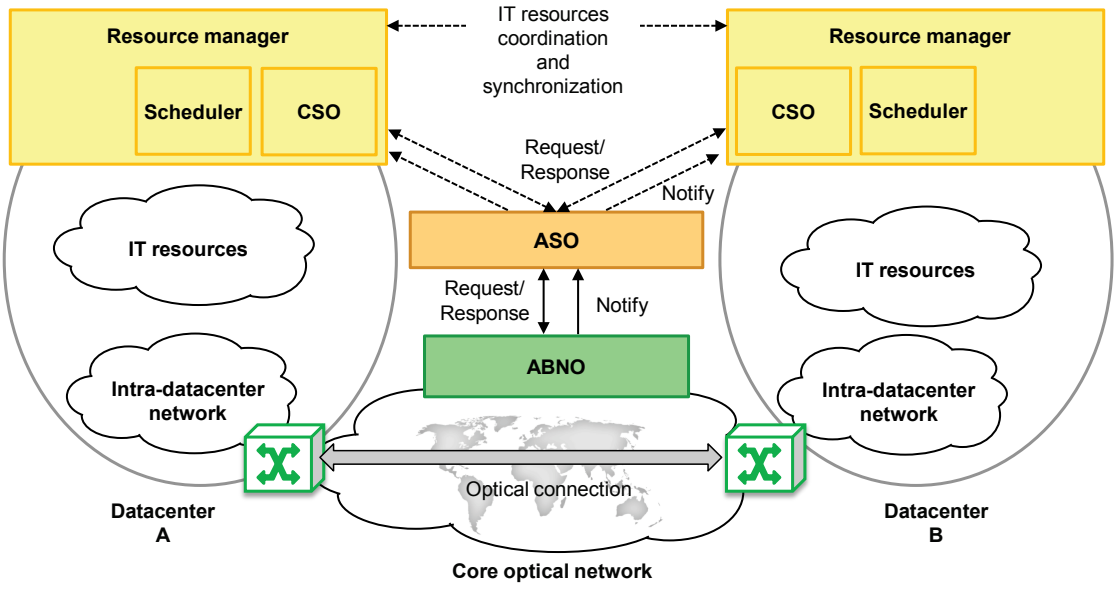

Fig. 8-2 ASO implementing transfer-based operations.

Since our scheduling algorithms run periodically each hour, VMs migrations are required to be performed within each period. In fact, the shorter the transfer time, the better the offered network service is, by having earlier the VMs in their proper locations.

ASO is deployed between the ABNO and DC resource managers (Fig. 8-2) and implements *polling-based* and *notification-based* connectivity models.

In the polling-based model, each local DC resource manager manages connectivity to remote DCs so as to perform VM migration in the shortest total time. CSO in the source DC resource manager requests LSP set-up, teardown, as well as elastic operations to the ASO, which forwards them to the ABNO (Fig. 8-3a). In this model, the ASO works as a proxy between applications and the network. After checking local policies, the ABNO performs LSP operations on the controlled network.

As shown in the previous chapter, although resource managers have full control over the connectivity process, physical network resources are shared with a number of clients and LSP set-up and elastic spectrum increments could be blocked as a result of lack of resources in the network. Hence, resource managers need to implement some sort of periodical retries to increase the allocated bandwidth until reaching the required level. These retries could impact negatively on the performance of the inter-DC control plane and do not ensure achieving higher bandwidth.

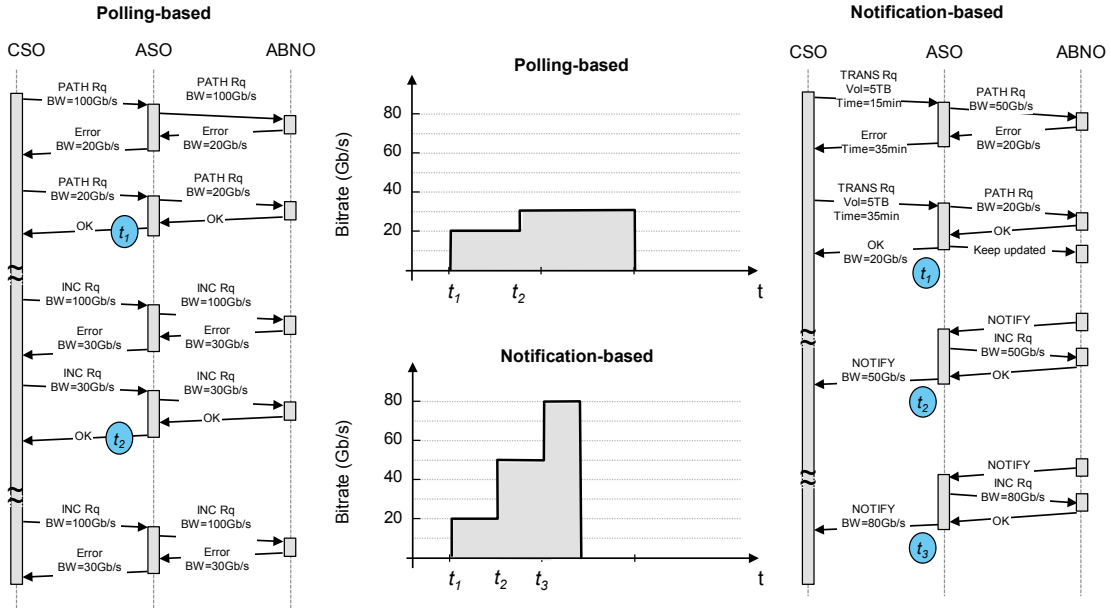

Fig. 8-3 Polling-based (a) and notification-based (b) models for transfer-based connections.

In the notification-based model (Fig. 8-3b), DC resource managers request data transferences instead of connection requests. The source DC resource manager sends a transfer request to the ASO specifying the destination DC, the amount of data that need to be transferred, and the maximum completion time. Upon its reception, the ASO requests the ABNO to find the greatest spectrum width available, taking into account local policies and current SLA and sends a response back to the cloud manager with the best completion time. The source cloud

manager organizes data transference and sends a new transfer request with the suggested completion time. A new connection is established and its capacity is sent in the response message; in addition, the ASO requests ABNO to keep it informed upon more resources are left available in the route of that LSP. ABNO has access to both the TED and the LSP-DB. Algorithms deployed in the ABNO monitor spectrum availability in those physical links. When resource availability allows increasing the allocated bitrate of some LSP, the ASO performs elastic spectrum operations so as to ensure committed transfer completion times. Each time the ASO modifies bitrate by performing elastic spectrum operations, a notification is sent to the source cloud manager containing new throughput. Cloud manager then optimizes VM migration as a function of the actual throughput while delegating ensuring completion transfer time to the ASO controller.

Table 8-1 presents the algorithm that we implemented in the ASO for transfer requests. It translates requested data volume and completion time, into a required bitrate, taking into account frequency slot width in the flexgrid network (line 1). Next an optical connection request is sent towards the ABNO, specifying source and destination of the connection and the bitrate (line 2). In case of lack of network resources (lines 3-6), the maximum available bitrate between source and destination DCs is requested to the ABNO, its result translated into the minimum completion time, which is used to inform the requesting DC resource manager. If the connection could be established, the ASO requests a subscription to the links in the route of the connection, aiming at being aware of available resources as soon as they are released in the network (line 7). Finally, the actual completion time is re-computed taking into consideration the connection's bitrate and both, bitrate and time, are communicated back to the requesting DC resource manager.

Table 8-1 Algorithm for transfer-based connection requests

```
INPUT: source, destination, dataVol, rqTime, slotWidth
OUTPUT: Response
1:  minBitrate ← translateAppRequest(dataVol, rqTime, slotWidth)
2:  netResp ← requestConnection(source, destination, minBitrate)
3:  if netResp == KO then
4:    maxBitrate ← getMaxBitrate(source , destination)
5:    minTime ← translateNetResponse(maxBitrate, dataVol)
6:    return {KO, minTime}
7:  requestSubscription(netResp.connId.route)
8:  time ← translateNetResponse(netResp.connId.bitrate, dataVol)
9:  return {OK, netResp.connId, netResp.connId.bitrate, time}
```

8.1.2 Illustrative results

For evaluation purposes, we developed scheduling algorithms in an OpenNebula-based cloud middleware emulator. Federated DCs are connected to an ad-hoc event-driven simulator developed in OMNET++. The simulator implements the

ASO and the flexgrid network with the ABNO and includes the algorithms described in Chapter 6 for elastic operations.

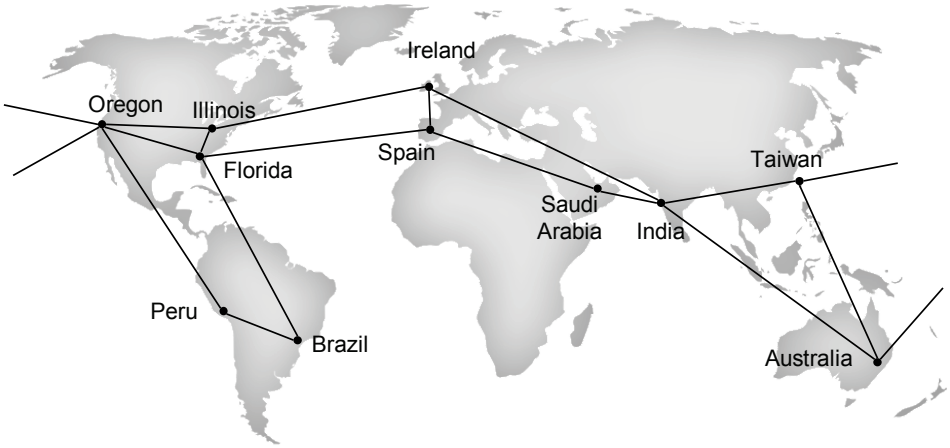

Fig. 8-4 Global inter-DC topology.

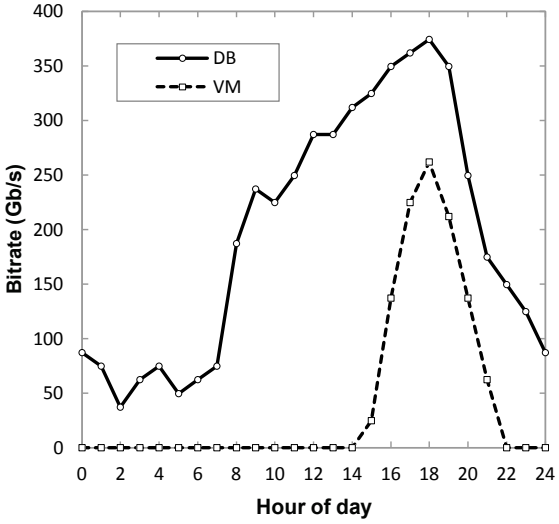

Fig. 8-5. Req. bitrate in a day.

For our experiments, we assume the global 11-node topology depicted in Fig. 8-4. Each node corresponds to locations that are used as sources for U2DC traffic. In addition, four DCs are strategically placed in Illinois, Spain, India, and Taiwan. DC2DC and U2DC traffic compete for resources in the physical network. We fixed the optical spectrum width to 4 THz, the spectral granularity to 6.25 GHz, the capacity for the ports connecting DCs to 1 Tb/s, the number of VMs to 35,000 with an image size of 5 GB each, and we considered 300,000 DBs with a differential image size of 450 MB and a total size of 5 GB each; half the size of Wikipedia [WIKISIZE]. Additionally, TCP, IPv4, GbE, and MPLS headers have been considered.

Fig. 8-5 shows the required bitrate to migrate VMs and to synchronize DBs in 30 minutes. VM migration is performed as a follow-the-work strategy and thus, connectivity is used for just during part of the day. In contrast, DB synchronization is performed along the day, although bitrate depends on the amount of data to be transferred, i.e., on users' activity.

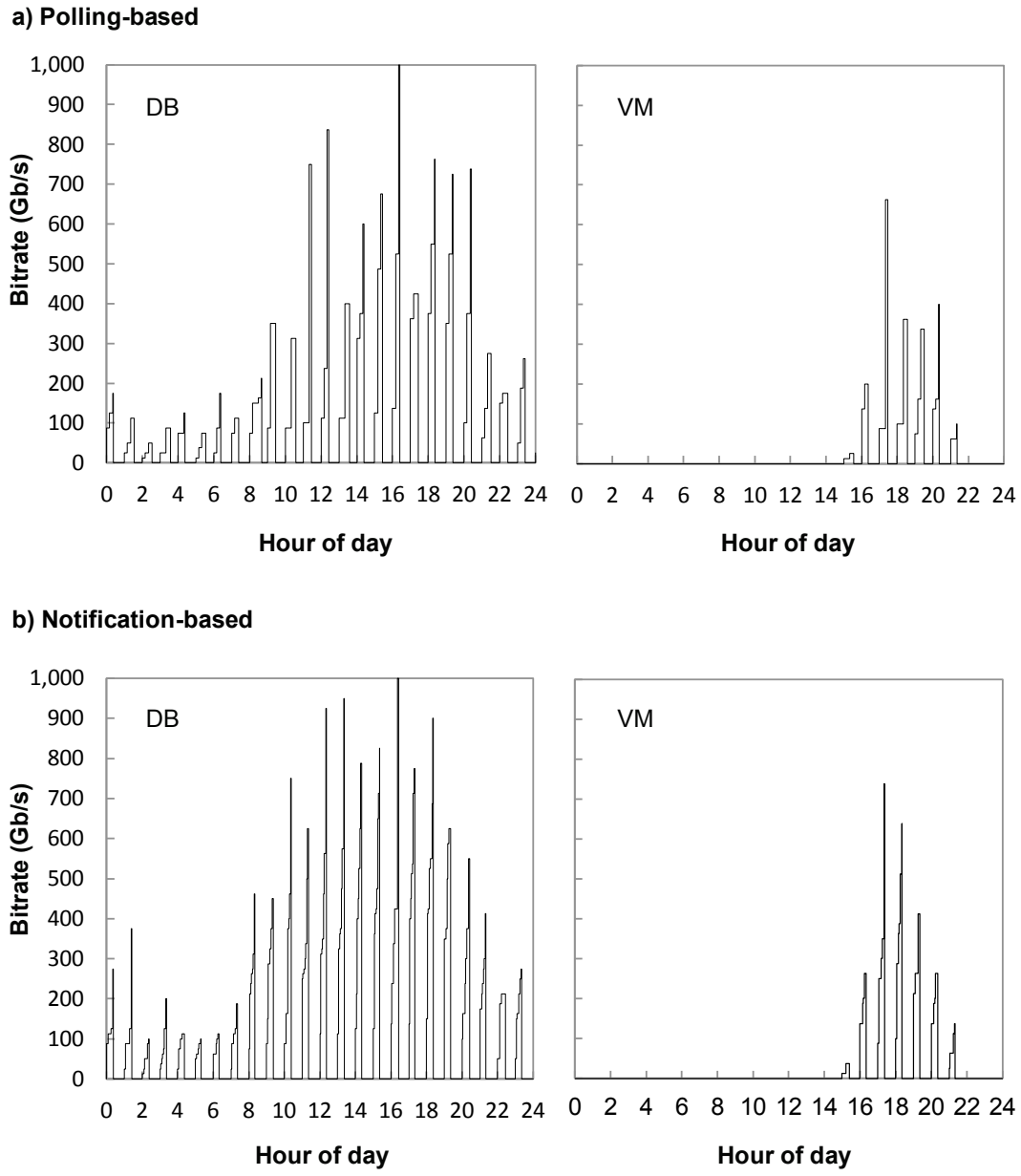

Fig. 8-6 DC2DC connection bitrate vs. time.

Fig. 8-6a depicts the assigned bitrate for DB synchronization and VM migration, respectively, between two DCs during a 24-hour period when the polling-based model is used. Fig. 8-6b shows the assigned bitrate when the notification-based

model is used. Polling-based model tends to provide longer transfer times as a result of not obtaining additional bitrate in retries. Note that although the initial bitrate which is assigned at each interval is the same, intervals tend to be narrower using the notification-based model. The reason is that models assign additional bitrate to the connections as soon as resources are released by other connections.

Table 8-2 shows the number of required requests messages per hour needed to increase bitrate of connections for the whole scenario. As illustrated, only 50% of those requests succeeded to increase connections' bitrate under the polling-based model, in contrast to 100% reached under the notification-based model. Additionally, Table 8-2 shows that when using the notification-based model both the maximum and average required time-to-transfer are significantly lower than when the polling-based model is used. The longest transfers could be done in only 28 minutes when the notification-based model was used compared to just under 60 minutes using the polling-based model. Note that the amount of requested bitrate is the same for both models.

Table 8-2 Performance results

	Requests	**Max/Avg Time-to-transfer (min)**	
	(#/h, %success)	**DBs**	**VMs**
Polling-based	43.1, 53.5%	58.0 / 29.0	54.0 / 28.2
Notification-based	65.3, 100%	28.0 / 22.4	27.0 / 22.2

8.2 Routing and Scheduled Spectrum Allocation

In this section, we explore ASO functionalities by using scheduling techniques in addition to elastic operations, with the twofold objective of fulfilling SLA contracts and increasing network operator revenues by accepting more connection requests. To decide whether a transfer-based connection request is accepted or not, the ASO controller solves the RSSA problem. The RSSA problem decides the elastic operations to be performed on already established connections so as to make enough room for the incoming request ensuring the requested completion time provided that the committed completion times of ongoing transferences are guaranteed.

To generalize the scenario, we consider that other connections not supporting DC transfer-based connections can be also established on the flexgrid network. For those connections, we assume that its duration (*holding time*) is known and its bitrate is fixed, i.e., no elastic operations are allowed. Moreover, it is worth

mentioning that ASO requires some additional elements to solve the RSSA problem, such as a specific service database containing information about the services deployed and a scheduled TED (sTED) to manage scheduling schemes.

8.2.1 Managing transfer-based connections

For illustrative purposes, let us assume two DCs connected through a flexgrid optical network. At some given time, a scheduling algorithm inside the cloud resource manager of one of the DCs (DC-1) decides to migrate a set of VMs towards the other DC (DC-2). Imagine that the total data volume is 54 TB and, because the scheduler runs periodically, 30 minutes is required to perform the whole migration.

As we demonstrated in Section 8.1, transfer-based connections reduce remarkably time-to-transfer; cloud resource managers request transfers using their native semantic, i.e., specifying the amount of data to be transferred, DC destination, and the completion time. The ASO in charge of the inter-DC network transforms the incoming transfer-based connection request into a connection request towards the ABNO. A scheme of the control architecture is shown in Fig. 8-7a.

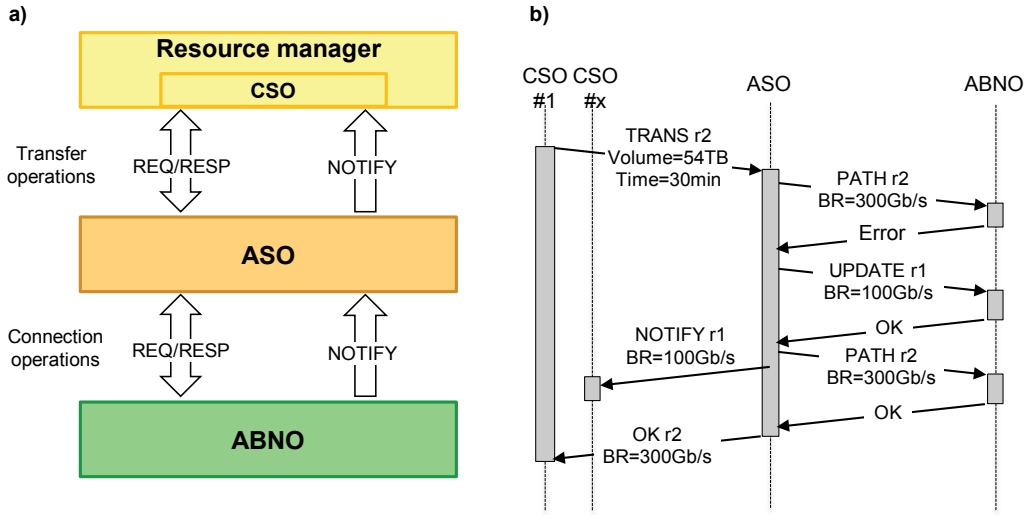

Fig. 8-7 Control architecture scheme (a). Example of messages exchanged to request connection operations (b).

To transfer 54 TB from DC-1 to DC-2 in 30 min, a 300 Gb/s connection needs to be established. Let us assume that no resources are available at requesting time (t^0) to guarantee that completion time, in such case the ASO might try to reduce the resources allocated to other transfers currently in progress without exceeding their committed completion time, so as to make enough room for the incoming transfer. Fig. 8-8 shows examples of the described operation. The bitrate of optical connection supporting ongoing transfer *r1* in Fig. 8-8a can be squeezed; thus, increasing its scheduled completion time t_{r1}, but without violating the committed

completion time, c_{r1}. Released resources can be used to set-up an optical connection to support the requested transfer *r2*.

Fig. 8-7b represents the sequence of messages exchanged to set-up the optical connection for the requested transfer *r2* including updating the bitrate of the optical connection supporting transfer *r1*. Note that the originating service client for transfers *r1* and *r2* are different, i.e., to serve a request from one client, we are reducing the resources allocated to another one. Notwithstanding, this is a *best practice* since in flexgrid networks elastic operations on established connections can be done without traffic disruption and the committed completion time is not exceeded. In addition, the originating service client is notified upon any update in the connection bitrate to help the involved scheduler to organize the underlying data transference.

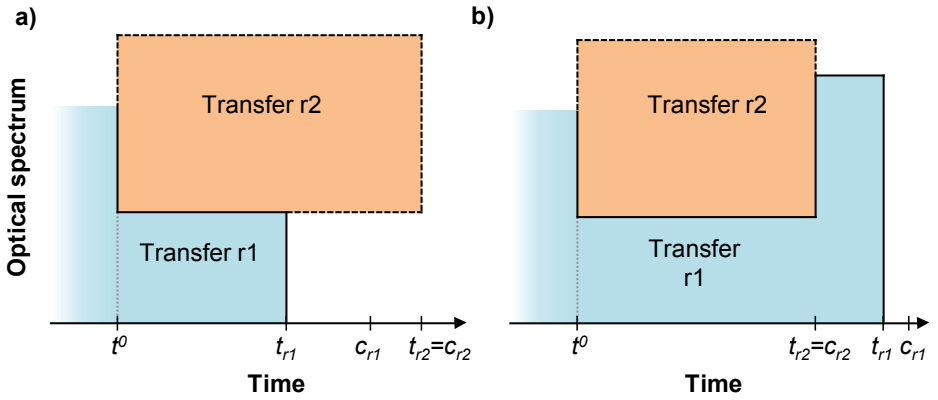

Fig. 8-8 Scheduling schemes for SA. a) Bitrate squeezing only and b) with scheduled resource reservation.

Another example is depicted in Fig. 8-8b. In the example, to ensure the committed completion time for transfer *r1*, some resources need to be allocated again to it as soon as transfer *r2* is completed. In view of the example, it is obvious that spectral resources need to be scheduled so as to be allocated to some transfer in the future. It is clear now that elastic operations include increasing and decreasing connections' bitrate.

8.2.2 The RSSA Problem

In this section, we focus on the RSSA problem to be solved each time a transfer-mode request arrives to ASO. Note that the RSA problem can be used to find a route and SA for fixed-bitrate connections arriving to ASO, provided that the scheduling data is supplied to the PCE within the ABNO architecture.

To simplify the RSSA problem, we limit to one the elastic operations that can be scheduled for each ongoing transference. The RSSA problem can be formally stated as:

Given:

- a network topology represented by a graph $G(L, E)$, being L the set of locations and E the set of fiber links connecting two locations;
- a subset $D \subseteq L$ with those locations source of transfer-mode traffic (DCs);
- a subset $U \subseteq L$ with those locations source of fixed-bitrate traffic;
- the characteristics of the spectrum for each link in E: a set S of available spectrum slices of a given spectrum width;
- the characteristics (capacity and number of flows) of the optical transponders equipped in each location;
- a set R with the transferences currently in progress in the network. For each transference $r \in R$, the tuple $<o_r, d_r, v_r, c_r, r_r, s^0_r, s^1_r, t^1_r, t_r>$ specifies the origin (o_r) and destination (d_r) DCs, the remaining amount of data (v_r) to be transferred, the requested completion time (c_r), the route (r_r) and slot currently allocated (s^0_r), the scheduled slot allocation (s^1_r) to be performed at time t^1_r, and the scheduled completion time (t_r);
- a set C with fixed-bitrate connections currently established in the network. For each connection $c \in C$, the tuple $<o_c, d_c, c_c, r_c, s_c>$ specifies the origin (o_c) and destination (d_c) locations, the requested completion time (c_c), the route (r_c), and the allocated slot (s_c);
- the new transfer-mode request described by the tuple $<o_r, d_r, v_r, c_r>$. Let t^0 denote the current time.

Output:

- the route (r_r), the slot allocation (s^0_r), and the scheduled completion time (t_r) for the new transference request. The bitrate of the connection is given by $b(s^0_r, l(r_r))$, where $l(r_r)$ is the total length of the route r_r.
- the new SA (w^0_r), scheduled reallocation (w^1_r, t^1_r) and completion time (t_r) for each transference request to be re-scheduled.

Objective: Minimize the number of connections to be re-scheduled to make room for the incoming request.

8.2.3 Mathematical formulation

The problem consists in re-scheduling some of the transferences in progress. A re-scheduling for any transference r includes two SAs: s^0_r from t^0 to t^1_r and s^1_r from t^1_r to t_r, so that the combined area of the two rectangles allow conveying the remaining amount of data v_r and $t_r \leq c_r$. It is worth noting that the capacity of the connections is also limited by the available capacity in the optical transponders.

It is clear that scheduled SAs entail deciding both dimensions, spectrum and time, which might lead to non-linear constraints. To avoid nonlinearities, we propose pre-computing any feasible combination of rectangles. Owing to that and without loss of generality, we need to discretize time into time intervals.

Fig. 8-9 shows three feasible re-scheduling solutions for a given transference r. Note that the total area of the three pairs of rectangles is identical and $t_r \leq c_r$ in all three solutions.

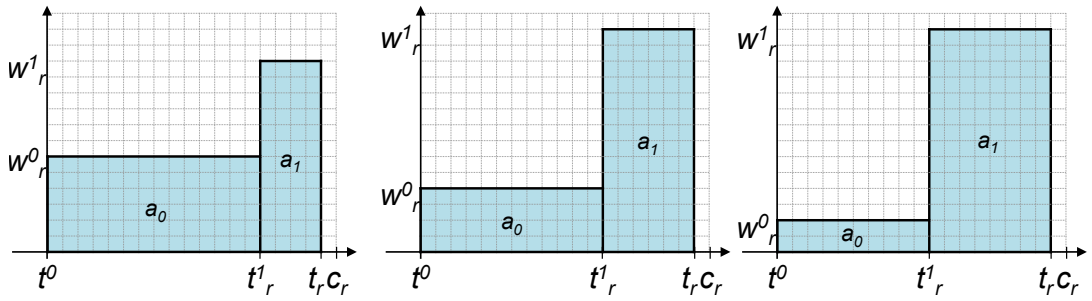

Fig. 8-9 Three feasible re-scheduling for a given transference.

Table 8-3 presents an algorithm to compute the sets of feasible rectangles for each ongoing transference. Sets of rectangles A_0 and A_1 are defined: set A_0 contains rectangles specifying allocation w^0_r from time t^0 to t^1_r, whereas rectangles in set A_1 specify scheduled allocation w^1_r from t^1_r to t_r. In addition, matrix β_{01} indicates whether rectangle pair $a_0 \in A_0$ and $a_1 \in A_1$ is a feasible scheduling.

Table 8-3 Algorithm for computing rectangle sets

```
INPUT: r
OUTPUT: A0, A1, ß01
1:  A0 ← getRectangles(r)
2:  A1 ← Ø
3:  for each a0∈ A0 do
4:    A1 ← A1 U getRectangles(r, a0)
5:  for each a0∈ A0 do
6:    for each a1∈ A1 do
7:      ß01 ← computeFeasibility(r, a0, a1)
8:  return A0, A1, ß01
```

The *getRectangles* function (line 1 in Table 8-3) finds rectangles for set A_0. To prevent that any transference is stopped, area of rectangles in this set must be strictly greater than zero. In addition, SAs increasing current allocation contain current allocation and some contiguous spectrum, whereas those decreasing the allocation are created by including part of the current allocation. Additional constraints are added to ensure that SA does not exceed the capacity of source and destination transponders and to guarantee that the time defined by the rectangles does not exceed the committed completion time for the given request.

Next, for each found rectangle a_0, the set of rectangles A_1 to complete transference r is computed (lines 3-4). Note that one rectangle with area equal to 0 can be added, provided that the transference can be completed with rectangle a_0. Finally, the *computeFeasibility* function (line 7) checks rectangle compatibility to create matrix β_{01}.

A set of pre-computed rectangles A for the incoming transfer request is also found, ensuring that both, the requested completion time and the amount of data to transfer are satisfied.

As for the ILP model, the following sets and parameters are defined:

Sets:

E	Set of fiber links in the network, index e.
S	Set of frequency slices, index s.
T	Set of time intervals, index t.
R	Set of ongoing transferences, index r.
P	Set of routes between origin and destination for the new request, index p.
A	Set of candidate areas for the requested transference, index a.
$A_0(r)$	Set of candidate areas for transference r. Each area starts in t^0 and is defined by t^1_r and a SA.
$A_1(r)$	Set of candidate areas for transference r. Each area is defined by a starting interval t^1_r, a scheduled completion time t_r, and a SA.

Parameters:

ω_{ar}	1 if transference r was assigned to area a; 0 otherwise.
δ_{as}	1 if area a includes slice s; 0 otherwise.
γ_{at}	1 if area a includes time interval t; 0 otherwise.
ρ_{pe}	1 if route p uses link e; 0 otherwise.
ρ_{re}	1 if transference r uses link e; 0 otherwise.
$\beta_{raa'}$	1 if pair of areas $a \in A_0$ and $a' \in A_1$ for transference r are feasible; 0 otherwise.
δ_{kest}	1 if fixed-bitrate connection k uses slice s in link e at time interval t; 0 otherwise.
α_{est}	0 if slice s in edge e is used by any fixed-bitrate connection time interval t, i.e., $\alpha_{est} = 1 - \sum_{k \in K} \delta_{kest}$; 1 otherwise.

Additionally, the following variables are defined:

x_{ap} Binary; 1 if the new transference is assigned to area a through route p; 0 otherwise.

y_r Binary, 1 if transference r is re-scheduled; 0 otherwise.

x^0_{ar} Binary; 1 if transfer r is assigned to area $a \in A_0$; 0 otherwise.

x^1_{ar} Binary; 1 if transfer r is assigned to area $a \in A_1$; 0 otherwise.

The ILP model for the RSSA problem is as follows:

$$\text{(RSSA) minimize} \quad \sum_{r \in R} y_r \tag{8.1}$$

subject to:

$$\sum_{a \in A} \sum_{p \in P} x_{ap} = 1 \tag{8.2}$$

$$\sum_{a \in A_0(r)} x^0_{ar} = 1 \quad \forall r \in R \tag{8.3}$$

$$\sum_{a \in A_1(r)} x^1_{ar} = 1 \quad \forall r \in R \tag{8.4}$$

$$\sum_{a' \in A_1(r)} \beta_{raa'} x^1_{a'r} \geq x^0_{ar} \quad \forall r \in R, a \in A_0(r) \tag{8.5}$$

$$x^0_{ar} - \omega_{ar} \leq y_r \quad \forall r \in R, a \in A_0(r) \tag{8.6}$$

$$\begin{aligned} &\sum_{r \in R} \sum_{a \in A_0(r)} \delta_{as} \cdot \gamma_{at} \cdot \rho_{re} \cdot x^0_{ar} + \\ &\sum_{r \in R} \sum_{a \in A_1(r)} \delta_{as} \cdot \gamma_{at} \cdot \rho_{re} \cdot x^1_{ar} + \quad \forall e \in E, s \in S, t \in \{t^0, \ldots, T\} \\ &\sum_{a \in A} \sum_{p \in P} \delta_{as} \cdot \gamma_{at} \cdot \rho_{pe} \cdot x_{ap} \leq \alpha_{est} \end{aligned} \tag{8.7}$$

The objective function (8.1) minimizes the number of transferences which bitrate needs to be reduced so the new transfer request can be served.

Constraint (8.2) guarantees that the transfer request is served assigning it to a route p and a rectangle a, which defines the SA and the scheduled completion time.

Constraints (8.3)-(8.5) select a feasible pair of rectangles for each ongoing transference r. Constraints (8.3)-(8.4) ensure that one rectangle in set A_0 and another in set A_1 are selected, whereas (8.5) assures that the selected pair of rectangles creates a feasible solution for transference r, i.e., committed completion

time and amount of data to be transferred are satisfied. Constraint (8.6) stores whether transference r needs to be re-allocated comparing current and assigned rectangle a_0.

Finally, constraint (8.7) guarantees that each frequency slice in each link is not used by more than one connection during each time interval.

8.2.4 Complexity analysis

The RSSA optimization problem is *NP-hard*, since it includes the RSA problem, which was proved to be *NP-hard* in [VeCa14] and [RuPi13].

Regarding its size, the number of variables is $O(|P| \cdot |A| + |R| \cdot (|A_0| + |A_1|))$ and the number of constraints is $O(|R| \cdot |A_0| + |E| \cdot |S| \cdot |T|)$. Considering a realistic scenario on a network with 30 links, 640 frequency slices, and 3,600 one-second time periods, the number of variables and the number of constraints are in the order of 10^8, which makes the above ILP model un-tractable even using state-of-the-art computer hardware and the latest commercially available solvers, e.g., CPLEX.

As a result of the size of the RSSA problem, next we propose an algorithm to solve the RSSA problem in realistic scenarios.

8.2.5 Algorithms to manage transfer-based connection requests

In this section, we devise a heuristic algorithm to solve the RSSA problem, which is executed upon a transference request arrives to ASO. In contrast, note that the RSA problem is solved for fixed-bitrate connection requests. In addition, a policy to assign resources to ongoing transferences as soon as they are released is presented.

A. Algorithm for the RSSA Problem

Table 8-4 shows the algorithm proposed for solving the RSSA problem. Firstly, the minimum admissible bitrate for the transference is found (line 1) considering both, data volume and completion time. In lines 2-5, the algorithm solves an RSA problem (we use a shortest-path for route selection with a first-fit SA policy); in case a feasible solution is found, the slot with maximum width is assigned in an attempt to reduce its scheduled completion time.

The remaining of the algorithm in Table 8-4 is specific for RSSA, and it is executed in case that no feasible solution for the RSA problem was found. All shortest routes with the same length in hops than the shortest one are explored in the hope of finding a feasible slot by increasing the set of available slots (whose spectrum width is smaller than the required) that are continuous along a given route. For each available slot, adjacent connections supporting ongoing transferences are considered as candidate to squeeze its current bitrate provided that the committed completion time is satisfied (lines 6-17). Specifically, the *findSlot* function tries to

increase the slot width in an attempt to make room for the incoming transference. The needed re-scheduling are afterwards stored in the *sTED* database and connection update requests are sent to the ABNO controller in charge of the optical networks. To compute the maximum bitrate squeezing that can be applied, scheduling schemes introduced in 8.2.1 are considered.

Table 8-4 Algorithm for the RSSA problem

```
INPUT: o, d, v, c
OUTPUT: route, slot
 1:  br ← getMinBitrate(v, c)
 2:  {route, slot} ← computeRSA(o, d, br, sTED)
 3:    if route ≠ Ø then
 4:      slot ← getMaxSA(route, sTED)
 5:      return {route, slot}
 6:  P ← computeKSP(o, d)
 7:  for each p ∈ P do
 8:    sw ← getSlotWidth(p.length, br)
 9:    S ← getFreeSlots(p, sTED)
10:    sort(S, slot width, DESC)
11:    for each s ∈ S do
12:      Q ← getAdjacentPaths(p, s, sTED)
13:      R ← findSlot(Q, s, sw, sTED)
14:      if R ≠ Ø then
15:        doReschedule (R)
16:        slot ← getSA(p, br, sTED)
17:        return {p, slot}
18:  for each p ∈ P do
19:    sw ← getSlotWidth(p.length, br)
20:    S ← getFeasibleBorderPathSlots(p, sTED)
21:    sort(S, slot width, DESC)
22:    repeat from line 11 to line 17
23:  return {Ø, 0}
```

As a final attempt to find a feasible solution, the computed routes are again explored (lines 18-22). This time, those slices currently allocated to connections supporting ongoing transferences that are at the border of their allocated slot are considered. Then, a slot is generated by reducing the SA of those connections (considering the committed completion time for the transference) (line 20). Next, a procedure similar to that described for available slots is followed.

B. Resource assignment policy

After a connection has been torn down, released resources need to be allocated to those connections supporting ongoing transferences, so as to satisfy the planned scheduling. For example, in Fig. 8-8b, some resources were scheduled to be allocated to transference r_1 then, after connection supporting transference r_2 is torn down, those resources need to be actually allocated. Lines 1-3 in Table 8-5 describe the proposed algorithm, where t^0 specifies the current time.

Table 8-5 Algorithm for Resource Assignment

```
INPUT: R, t^0
1:  for each r ∈ R do
2:    if t^1_r = t^0 then
3:      doReSchedule (r)
4:  sort(R, t_r-t^0, ASC)
5:  for each r ∈ R do
6:    expandMax(r)
```

However, some resources might remain unallocated even after scheduled allocations are done. Aiming at maximizing resource utilization, connections supporting ongoing transferences are offered the opportunity to increase allocated resources reducing thus its scheduled completion time (lines 4-6). Functions in lines 3 and 6 send the appropriate requests to the ABNO controller. Note that the benefit from the latter is completion time reduction, which also facilitates that those connections can be torn down sooner. That effect might be amplified by assigning released resources to those connections whose scheduled completion time is closer to the current time.

8.2.6 Illustrative results

In this section, we present the results obtained on a simulation scenario, where several network topologies have been considered.

A. Network scenario

Aiming at evaluating the performance of the proposed transfer-based connections, we developed an ad-hoc event-driven simulator in OMNeT++.

A dynamic network environment was simulated where incoming connection requests arrive to the system following a Poisson process and are sequentially served without prior knowledge of future incoming connection requests. We assume that the flexgrid optical technology is used in the network data plane, and one BV-OXC with 1 Tb/s BVT is installed in each location, being source of both, fixed-bitrate and transfer-mode connections.

B. Heuristic versus ILP model

Before evaluating the performance of the proposed transfer-mode connections, let us compare those of the methods proposed to solve the RSSA underlying problem: the ILP model and the heuristic algorithm.

We implemented the ILP model in Matlab and used CPLEX to solve the selected problem instances. Regarding the heuristic algorithm, it was implemented in C++ and integrated in OMNeT++. Both methods were run in a 2.4 GHz Quad-Core computer with 16 GB RAM memory under the Linux operating system. To generate problem instances, we used the simulation environment described above and implemented a small network topology with 6 nodes and 8 edges, while setting

the spectrum width to 1 THz, the slice width to 12.5 GHz, the BVT capacity to 800 Gb/s, and the maximum completion time to 30 seconds.

Table 8-6 summarizes some representative parameters for a set of instances and the obtained time-to-solve and the solution when the ILP and the heuristic algorithm were applied. For each problem instance, the number of ongoing transferences and the maximum size of sets A_0 and A_1 for every transference are detailed. Regarding solving times, the values in the ILP column are for CPLEX solving time, i.e., problem generation time (in the order of tens of seconds) and rectangle sets computation (hundreds of milliseconds) are not included.

Table 8-6 ILP model vs. heuristic comparison

$\|R\|$	max ($\|A_0\|$)	max ($\|A_1\|$)	Time-to-solve		Solution	
			ILP	Alg.	ILP	Alg.
7	691	3,409	10.3s	5 ms	1	1
8	2,173	7,203	15.4s	13 ms	1	1
11	1,442	3,693	19.8s	9 ms	1	1
14	1,592	5,845	25.9s	31 ms	1	1
15	1,447	4,571	26.8s	6 ms	1	1
17	1,463	5,381	19.8s	27 ms	1	2

Results show that several seconds are needed to solve really small problem instances using the ILP model. In contrast, time-to-solve is in the order of tens of milliseconds when the proposed RSSA algorithm is used, illustrating thus its applicability to real scenarios. In view of that, the heuristic algorithm is used to solve the RSSA problem in the simulation experiments presented next. Notwithstanding the computation time, the algorithm provides virtually the same solution as the ILP model; only for one instance, the algorithm reduced SA of two established connections, while the ILP limited that number to only one.

C. RSA vs. RSSA comparison

Three realistic national network topologies have been considered (Fig. 8-10): the 21-node Telefonica (TEL), the 20-node British Telecom (BT), and the 21-node Deutsche Telekom (DT) networks. For these experiments, the optical spectrum width is fixed to 4 THz in each link, the slice width to 12.5 GHz with granularity 6.25 GHz, and all connections use the QPSK modulation format.

Fixed-bitrate connections request 100 Gb/s during a fixed holding time of 500 seconds; an exponential inter-arrival time is considered. Source/destination pairs are randomly chosen with equal probability (uniform distribution) among all nodes. Different values of the offered network load are created by changing the inter-

arrival rate. We assume that no retrial is performed, i.e., if a fixed-bitrate connection request cannot be served, it is immediately blocked.

In contrast, transfer-based offered load is related to VM migration among DCs. The amount of VMs to migrate between two DCs is proportional to the number of considered VMs and randomized using a uniform probability distribution function. Requested completion time is limited to 1 hour. Different values of the offered network load are created by changing the total amount of VMs and the size of every VM. The amount of VMs ranges from 15,000 to 45,000, whereas VMs size ranges from 2.5 to 7.5 GB. Data volume is translated into required network throughput considering TCP, IP, Ethernet, and MPLS headers. In this case, we assume that transfer-mode requests are generated asking to complete the transference in 30 min. If the request cannot be served, one retrial is performed requesting resources so the completion time to be at most 1 hour. If the request cannot be served, it is finally blocked.

The ability to deal with transfer-mode connection requests of the RSSA algorithm in Table 8-4 was compared against that of the RSA, i.e., only lines 1-5 in the algorithm in Table 8-4. To that end, we used the simulation environment described above, where we implemented the ASO on top of the ABNO architecture in the control plane of the flexgrid network. Finally, all the results were obtained after requesting 15,000 connections and each point in the results is the average of 10 runs.

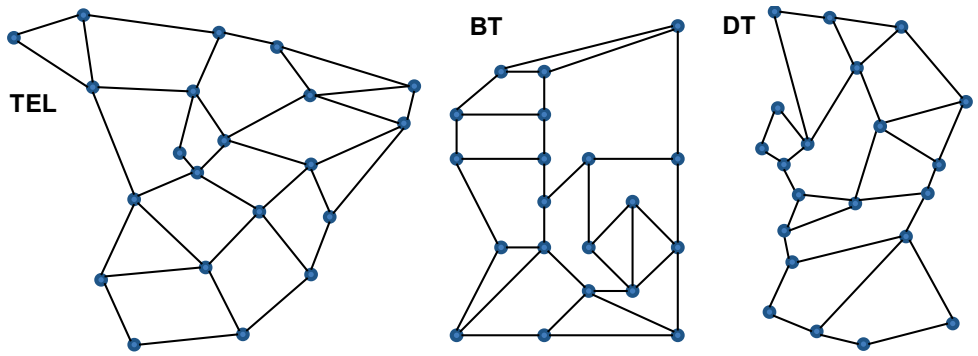

Fig. 8-10 The Telefonica (TEL), British Telecom (BT), and Deutsche Telekom (DT) network topologies used in this section.

To compare both the RSA and the RSSA algorithms, we first run simulations considering that only transfer-mode traffic is requested. Fig. 8-11a shows the percentage of UB for several offered loads and for each of the considered network topologies. VMs' size was fixed to 5 GB. For the sake of comparability between topologies, offered loads have been normalized to the value of the highest load.

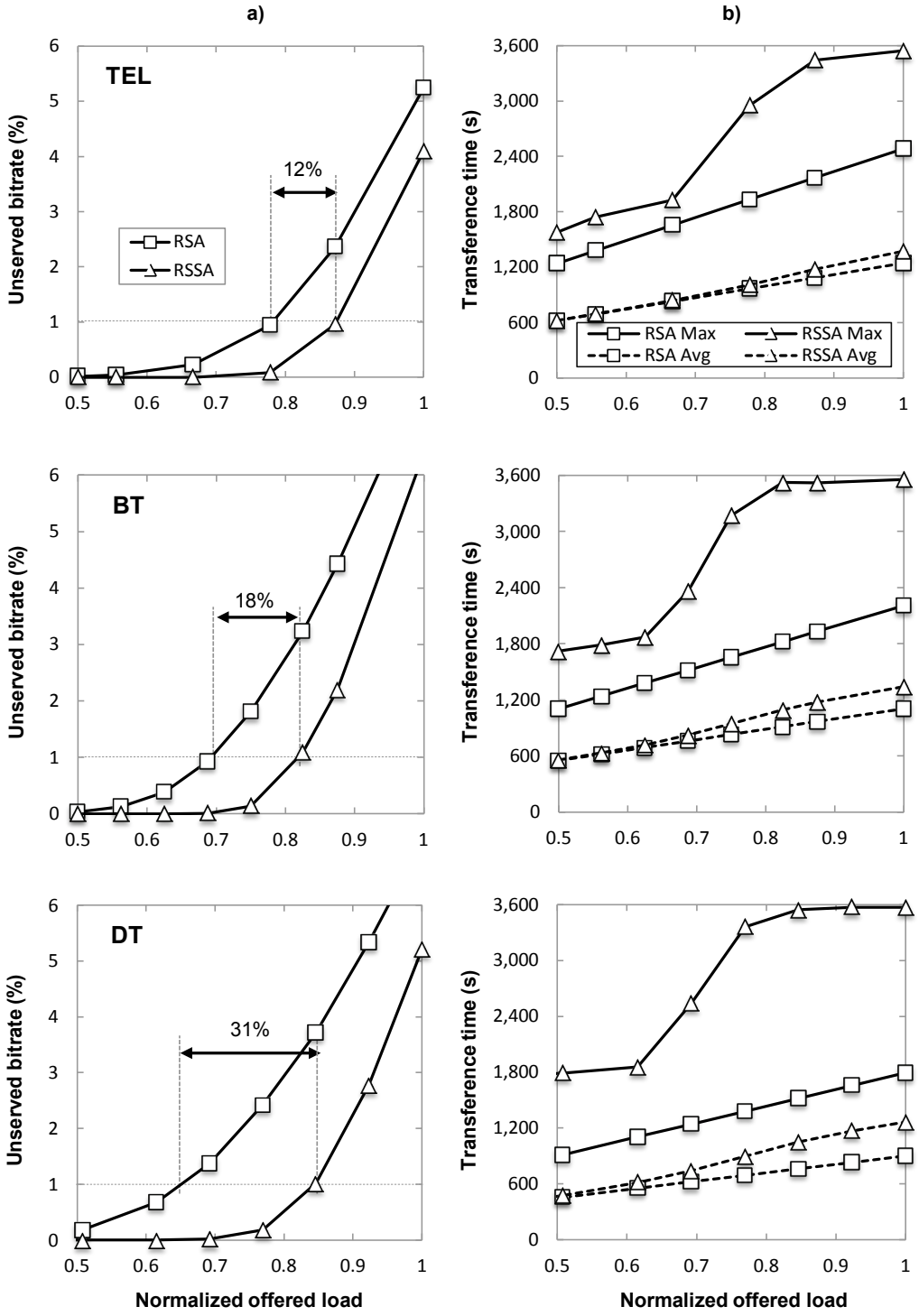

Fig. 8-11 Unserved bitrate and transference completion time vs. load (in terms of number of VMs). Only transfer-mode requests are considered.

The proposed RSSA algorithm is able to serve more traffic in all three network topologies compared with using just RSA. Interestingly, gain from using the RSSA is small under low loads since enough resources are available to serve all the

requests. As soon as the load increases, the ability of RSSA to perform elastic operations on the established optical connections, together with the scheduling to ensure that the committed completion time is not exceeded, brings clear benefits. Obviously, for high loads those benefits would cancel as few opportunities to reduce resources would exist. If we focus on 1% of UB, we observe that noticeable gains in the offered load are obtained. Table 8-7 presents those values, which range from 12% to more than 30% traffic gain when the RSSA algorithm is used.

Table 8-7 Offered load gain and transference time comparison

		TEL	BT	DT
Gain at 1% UB		12.2%	17.8%	30.9%
Transference time in seconds (RSSA)	**Max**	3,444	3,523	3,543
	Avg	1,180	1,092	1,048
Transference time in seconds (RSA)	**Max**	2,167	1,822	1,518
	Avg	1,085	912	761

Additionally, for the sake of providing a thorough study, we fixed the amount of VMs while varying their size from 2.5 to 7.5 GB. To that end, we selected the load unleashing 1% of UB when VMs' size was fixed to 5 GB. Fig. 8-12a shows the percentage of UB, while Fig. 8-12b plots transference completion times. As expected, the results obtained are in line with those in Fig. 8-11 obtained by varying the number of VMs; the amount of data to be transported is the underlying parameter being increased.

Table 8-8 Elastic operations at 1% of unserved bitrate

		TEL	BT	DT
# Spectrum reductions	**Max**	3.5	3.6	3.9
	Avg	1.09	1.13	1.18
# Spectrum expansions	**Max**	4.1	4.8	5.6
	Avg	1.18	1.27	1.37
% of connections experiencing elastic ops.	**Red.**	9.3	18.8	24.4
	Exp.	23.6	43.5	53.4

Because the number of elastic operations performed on each connection is an important factor, Table 8-8 summarizes the number of elastic operations (spectrum reduction and expansion) performed over each established connection supporting a

transference. The percentage of connections experiencing elastic operations is also summarized.

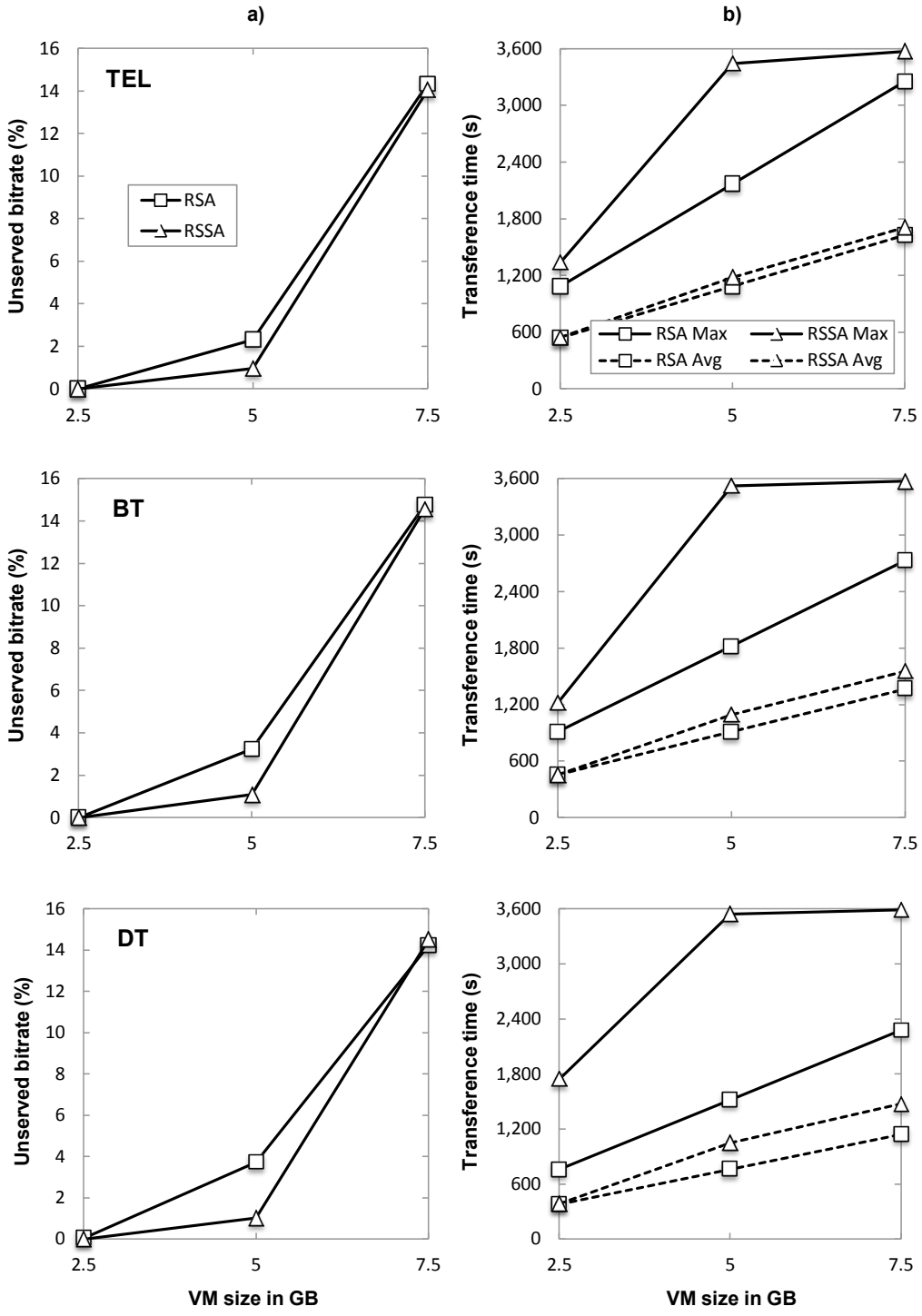

Fig. 8-12 Unserved bitrate and transference completion time vs. VMs' size. Only transfer-mode requests are considered.

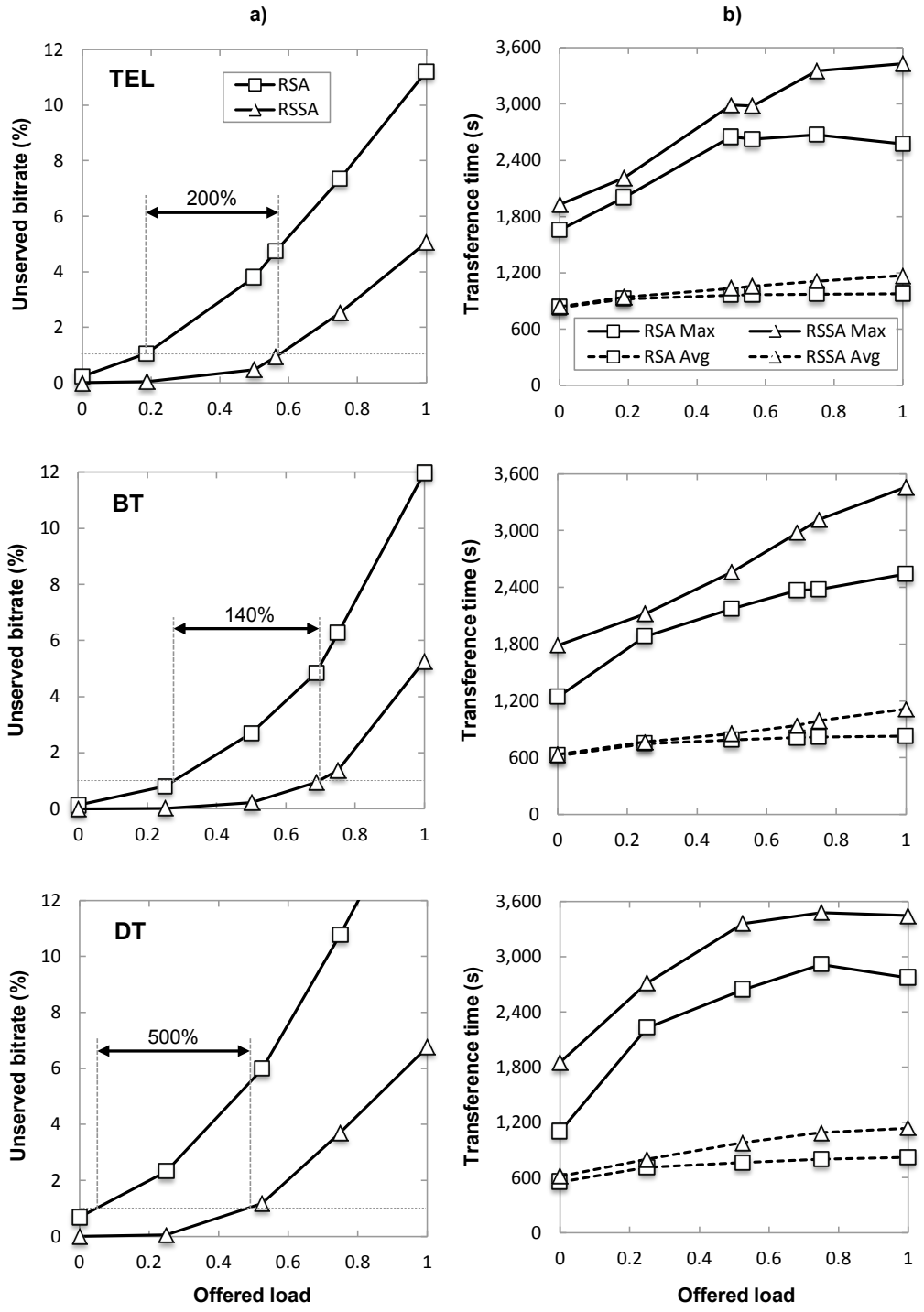

Fig. 8-13 Unserved bitrate of transfer-mode requests and transference completion time vs. offered load of fixed-bitrate traffic.

The average number of spectrum reductions and spectrum expansions that each connection experiences is just over one and arrives to a maximum of 4 reductions and 6 expansions. However, if we analyze the percentage of connections experiencing elastic operations, we realize that the percentage of expansions

doubles that of spectrum reductions. Recall that spectrum reduction can be performed when a connection request arrives, whereas spectrum expansions are performed when connections are torn down, to implement scheduled allocations and to reduce transference completion time. It is clear that scheduled allocations are as a result of previous spectrum reduction, and thus the number of reductions should be greater or equal to the number of expansions. Therefore, in view of Table 8-8, the algorithm for resource assignment proposed in Table 8-5 is responsible for performing extra spectrum expansions to reduce transference completion time and increase resource utilization.

Finally, we study the effect of mixing fixed-bitrate and transfer-mode traffics. Note that spectral resources allocated to fixed-bitrate connections must remain invariant along connections' life. We set transfer-mode load to one of the lower values in Fig. 8-11 and increased fixed-bitrate traffic load. Fig. 8-13a plots the percentage of unserved transfer-mode bitrate, whilst Fig. 8-13b shows transference completion times as a function of fixed-bitrate traffic load. Similarly to the previous study, offered loads have been normalized to the value of the highest load. Results show that the proposed RSSA algorithm leads to remarkable gains, ranging from 140% to 500% with respect to the RSA, in the amount of fixed-bitrate offered traffic. As for completion times, average values are similar for both algorithms and in line with those in Fig. 8-11b. Plots for maximum times, although higher in value for RSSA, follow a trend more similar in Fig. 8-13b. This is due to the fact that fixed-bitrate traffic reduces the possibility of creating room to incoming requests by reducing SAs on transfer-mode connections. To improve that, defragmentation techniques, such as the described in [Ca12], could be applied.

8.3 Conclusions

In this chapter, ASO implementing a northbound interface with application-oriented semantic has been proposed as a new abstraction layer between DC resource managers and the ABNO in charge of flexgrid-based interconnection network. Each resource manager can request transfer operations specifying the destination DC, the amount of data to be transferred, and the desired completion time.

The polling-based and the notification-based connectivity models have been compared in a DC federation scenario considering ASO. In the polling-based model, resource managers request connection operations to the ASO, which forwards them to the ABNO in charge of the interconnection network. Moreover, it needs periodical retries requesting increase connection's bitrate, which do not translate into immediate bitrate increments and could have a negative impact on the performance of the inter-DC control plane. In contrast, in the notification-based model, DC resource managers request transfer-based operations to ASO and it takes advantage of the use of notify messages. ASO is then aware of DC resource

managers' necessities as well as network resources availability; thus, being able to reduce time-to-transfer circa 20% on average and up to 50% for maximum values.

After the benefits on time-to-transfer when ASO is responsible of managing DC interconnection were shown, we considered the scenario where a set of customers request transfer operations to ASO. We defined the problem of deciding the set of already established connections for which their current SA can be reduced without violating the committed completion time, so as to make enough room to the incoming transfer-mode request; we named it as the RSSA problem.

We formally stated the RSSA problem and modeled it using an ILP formulation. In view of the complexity of the problem and owing to the fact that it needs to be solved in real time, a heuristic algorithm was proposed. Aiming at taking advantage of the resources released, when connections have been torn down, a policy to assign them to ongoing transferences was eventually proposed. The policy gives more priority to those connections with the closest scheduled completion time.

Illustrative results were obtained by simulation on three telecom network topologies. The performance of the proposed RSSA algorithm when dealing with transfer-mode requests was compared against that of the RSA. Results showing remarkable gains, as high as 30% and 500% in the amount of transfer-mode and fixed-bitrate traffic, respectively, were obtained for the considered scenarios.

Since RSSA entails reducing and scheduling increments in spectrum allocated to established connections, maximum completion times were increased to just under committed completion times. Notwithstanding, average completion times obtained when RSSA was applied showed values similar to that of the obtained with RSA.

It is worth highlighting that connections supporting transferences experienced few elastic operations, just above one reduction and one expansion.

To conclude, in this chapter, we have shown the benefits of using ASO on top of ABNO to manage inter-DC connectivity considering transfer-based connections and using scheduling techniques and elastic operations to simultaneously fulfill SLAs from a set of customers. Next chapter studies benefits from virtualization techniques to satisfy connectivity requirements considering service-specific SLA parameters.

Chapter 9

On-demand customer virtual network provisioning

As studied in Chapter 5, emerging applications and services over IP transport networks require not only intelligent control and management architectures to provide bandwidth-on-demand, but also service-specific SLA parameters to guarantee QoS and avoid service interruption. Thanks to network virtualization, operators can optimize their infrastructure while offering virtual network resources for service-specific requirements. Among the wide range of services over IP transport networks, those related to NFV are of particular interest. To facilitate network resources virtualization to support NFV, we assume that an ASO is connected on top of the ABNO to implement the ACTN framework. Customers can request on-demand connectivity between their EPs to reconfigure CVN topology dynamically. In addition to CVN reconfiguration, QoS constraints and service guarantees can be requested to satisfy service-specific requirements.

In this chapter, an algorithm for CVN reconfiguration with QoS constraints and bitrate guarantees is presented. An MPLS virtual network supporting CVNs is pre-planned or dynamically reconfigured. Then, we focus on the study of two different approaches to satisfy the committed bitrate guarantees in CVN links taking advantage of SRLGs: partial protection and diversity.

Exhaustive simulation results show remarkable network CAPEX savings as well as performance improvement when the MPLS network is reconfigured instead of pre-planned and when the diversity strategy is considered against partial protection.

9.1 CVN reconfiguration

Based on the ACTN model, let us assume a generic landscape with two main actors: *i*) a network operator and *ii*) a set of customers requiring CVN reconfiguration.

In this scenario, we consider the layered network depicted in Fig. 9-1, where three layers can be identified; from top to bottom:

- the customer layer with CVNs connecting customer's EPs. Every CVN's link is supported by one or more MPLS paths;
- the network operator's IP/MPLS network layer consisting in a number of IP/MPLS routers connected through virtual links supported by optical connections;
- the network operator's optical core network consisting in a number of OXCs and optical links.

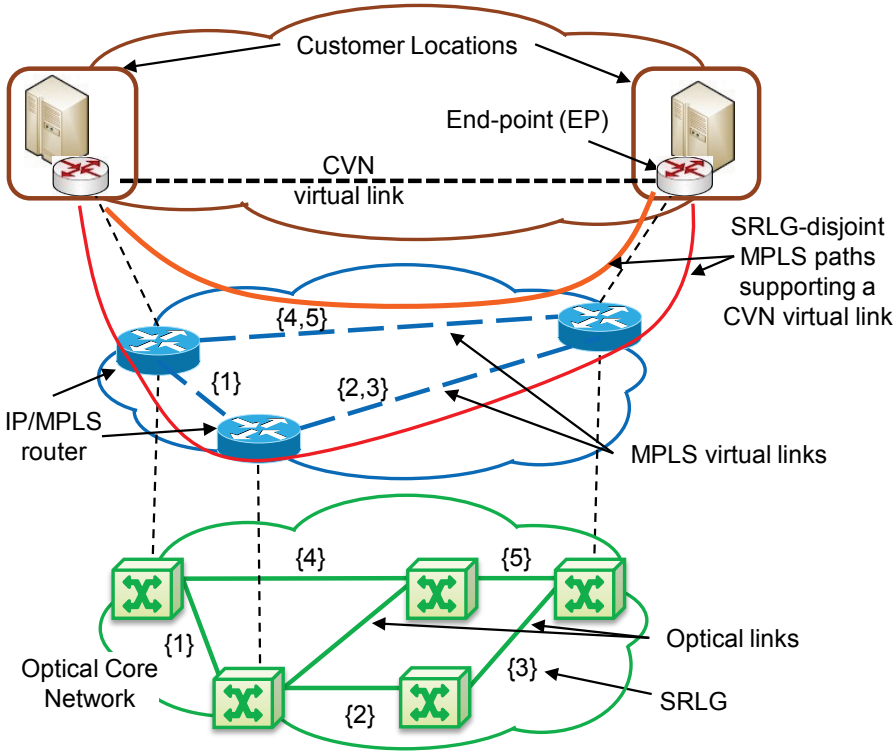

Fig. 9-1 3-layered network topology

Regarding the management architecture, the one in Fig. 9-2, based on ASO and ABNO, is considered. Benefits from using ASO on top of ABNO have been shown in previous chapters. In this chapter, the ASO module maintains service-related databases (e.g., service and CVN DBs), whereas ABNO maintains network-related databases, i.e., TED and LSP-DB.

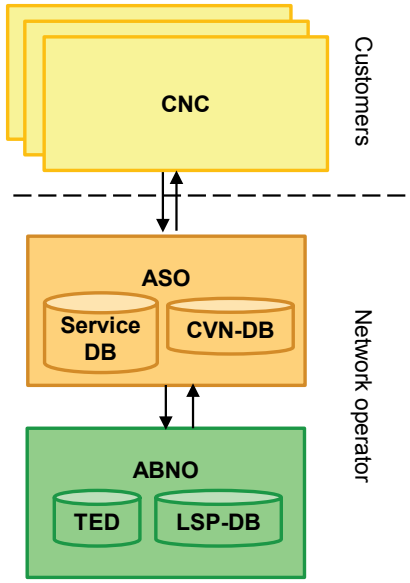

Fig. 9-2 Management architecture

In the considered scenario, customers can request CVN reconfigurations to ASO to adapt their CVN to their current needs. CVN reconfiguration requests include set-up or tear-down CVN links and increase or decrease their capacity. Fig. 9-3 presents an example of how a CVN is modified with the time.

In addition to dynamic reconfiguration, SLAs between customers and the network operator can include QoS constraints and bitrate guarantees; QoS can be based on maximum delay permitted between EPs, whereas minimum bandwidth could be requested under failure conditions.

A way to implement bitrate guarantees is by using spatial diversity, i.e., when bitrate guarantee is requested, two or more SRLG-disjoint MPLS paths are established to simultaneously support a CVN link. An example is depicted in Fig. 9-1, where every optical link is associated with a different SRLG identifier. The SRLG identifier of an MPLS virtual link is the union of the SRLGs supporting that link. Finally, the represented CVN link is supported by two SRLG disjoint MPLS paths: one supported by SRLGs {1, 2, 3} and the other by SRLGs {4, 5}. Therefore, in the event of an optical link failure, the CVN link capacity is squeezed to the bandwidth of the remaining MPLS path. Note that, since only two SRLG were set up, only half of the total CVN link capacity can be guaranteed. For the sake of simplicity, in this chapter, we assume that condition.

To support CVNs, the network operator might either plan and deploy an MPLS virtual network with enough capacity or allow the MPLS virtual network to be dynamically adapted by creating or releasing MPLS virtual links.

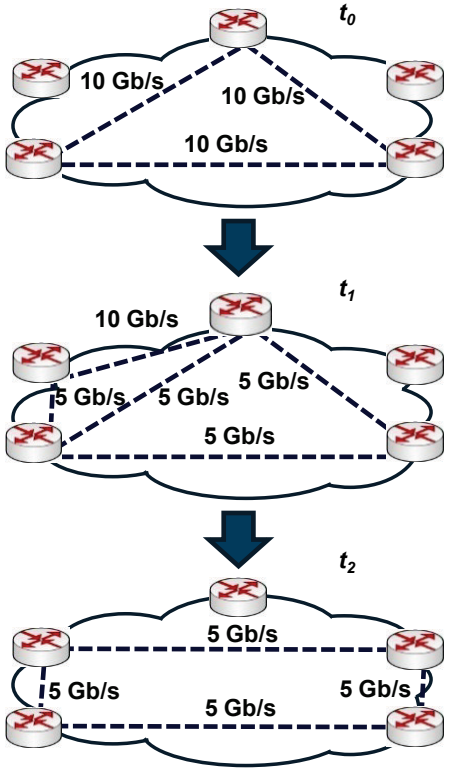

Fig. 9-3 CVN evolution

9.2 CVN reconfiguration with QoS constraints and bitrate guarantees

In this section, we focus on providing on-demand reconfiguration of CVN with QoS constraints and bitrate guarantees. Impacts on performance and CAPEX, in terms of amount of BVTs to be installed, are studied.

9.2.1 CVN-QBG problem

The CVN reconfiguration with QoS constraints and Bitrate Guarantees (CVN-QBG) problem can be formally stated as follows:

Given:

- a multilayer network represented by the graph $G_o(V_o, E_o)$; being V_o the set of optical nodes and E_o the set optical of links;
- a MPLS network represented by a graph $G_v(V_v, E_v)$; being V_v the set of MPLS nodes and E_v the set of MPLS virtual links.

- a set of customers C; each $c \in C$ manages its own CVN, which topology is represented by a fully meshed graph $G_c(V_c, E_c)$; being V_c the set of EPs and E_c the set of CVN virtual links.
- a CVN reconfiguration request from customer c, which is represented by the tuple $<c, B^r, Q^r, W^r>$, where c identifies the customer, B^r is the capacity matrix of the CVN links between EPs, Q^r is the QoS matrix, and W^r is a matrix with CVN links capacity to be guaranteed ($w^r \leq 0.5 \cdot b^r$).
- the *MPLSreconf* parameter indicating whether MPLS reconfiguration is allowed or not.
- the *MPLSreopt* parameter specifying whether MPLS reoptimization is allowed or not.

Output: the set L of MPLS paths over G_v and lightpaths (if MPLS reconfiguration is allowed) over G_o that need to be established to serve the CVN reconfiguration request.

Objective: minimize the used resources.

Aiming to solve the CVN-QBG problem in dynamic scenarios, we propose the algorithm in Table 9-1. Those CVN links with unchanged or decreased requirements are first updated to release resources that can be afterwards reused (lines 2-4 in Table 9-1). The rest of the CVN links are de-allocated from G_c (lines 5-7) and set up again (lines 8-11) using the *setupCVNLink* algorithm described in Table 9-2. Note that a CVN reconfiguration request is blocked if one of the CVN links could not be updated.

Table 9-1 Algorithm for CVN-QBG.

INPUT: G_c, B^r, Q^r, W^r
OUTPUT: L

```
 1:  D ← ∅, L ← ∅
 2:  for each e ∈ E_c do
 3:    if B^r(e) ≤ B(e) AND Q^r(e) ≤ Q(e) AND W^r(e) ≤ W(e) then
 4:      update(e, B^r(e), Q^r(e), W^r(e))
 5:    else
 6:      dealloc(e, G_c)
 7:      D ← D U {e}
 8:  for each d ∈ D do
 9:    l ← setupCVNLink(d, B^r(d), Q^r(d), W^r(d), G_c)
10:    if l = ∅ then return ∅
11:    L ← L U {l}
12:  return L
```

The *setupCVNLink* algorithm starts finding an MPLS path with capacity w on G_v ensuring q (line 2 in Table 9-2). In the case that no route is found and MPLS reoptimization is allowed, reoptimization on the shortest path for d in G_v is performed (line 5). The reoptimization algorithm tries to make room for the requested bandwidth by reallocating MPLS paths established on the selected path.

If no route is found for d and MPLS reconfiguration is allowed, new MPLS virtual links are set-up adding enough capacity to G_v so as to serve d (lines 7-9). If a path is not found, CVN link d cannot be served. Otherwise, it is allocated in G_v. Next, a number of MPLS paths, limited by the k parameter and disjoint with the MPLS path guaranteeing w, are established to convey the remaining bandwidth to be served.

Table 9-2 setupCVNLink algorithm.

```
INPUT: d, b, q, w, Gc
OUTPUT: L
 1:  L ← Ø
 2:  R ← findPath(d, w, q)
 3:  if R = Ø then
 4:    if MPLSreopt then
 5:      reoptimize(shortestPath(d, q), w)
 6:      R ← findPath(d, w, q)
 7:    if R = Ø AND MPLSreconf then
 8:      L ← L U setupMPLSLinks(d, w, q)
 9:      R ← findPath(d, w, q)
10:    if R = Ø then return Ø
11:  L ← L U allocate(Gc, d, R, w)
12:  bpend ← b-w; subpaths ← 0
13:  while bpend > 0 AND subpaths < k do
14:    S ← findDisjointPath(d, 1, q, R)
15:    if S = Ø then (procedure similar to lines 4-10)
16:    bS ← min(bpend, getMaxBandwidth(S))
17:    L ← L U allocate(Gc, d, S, bS)
18:    bpend ← bpend-bS; subpaths++
19:  return L
```

9.2.2 Illustrative results

The performance of the proposed algorithm for the CVN-QBG problem is studied considering three incremental approaches:

- a *static* MPLS network approach in which the IP/MPLS virtual network is pre-planned beforehand, k is fixed to 1 (no multipath), and *MPLSreconf* is fixed to 0 (reoptimization is not allowed);
- a MPLS *dynamic* approach allowing MPLS network reconfiguration;
- and a MPLS *dynamic* approach allowing multipath and reoptimization.

The static MPLS virtual network topology was pre-planned so as to provide the same performance as that of the MPLS dynamic approach at 1% of blocking probability (BP).

A 30-location network topology based on the Telefonica's national network was considered, where each location is equipped with an optical node and an IP/MPLS router with 100 Gb/s BVTs. The management architecture in Fig. 9-2 and the

proposed algorithm have been implemented in an OMNeT++-based network simulator.

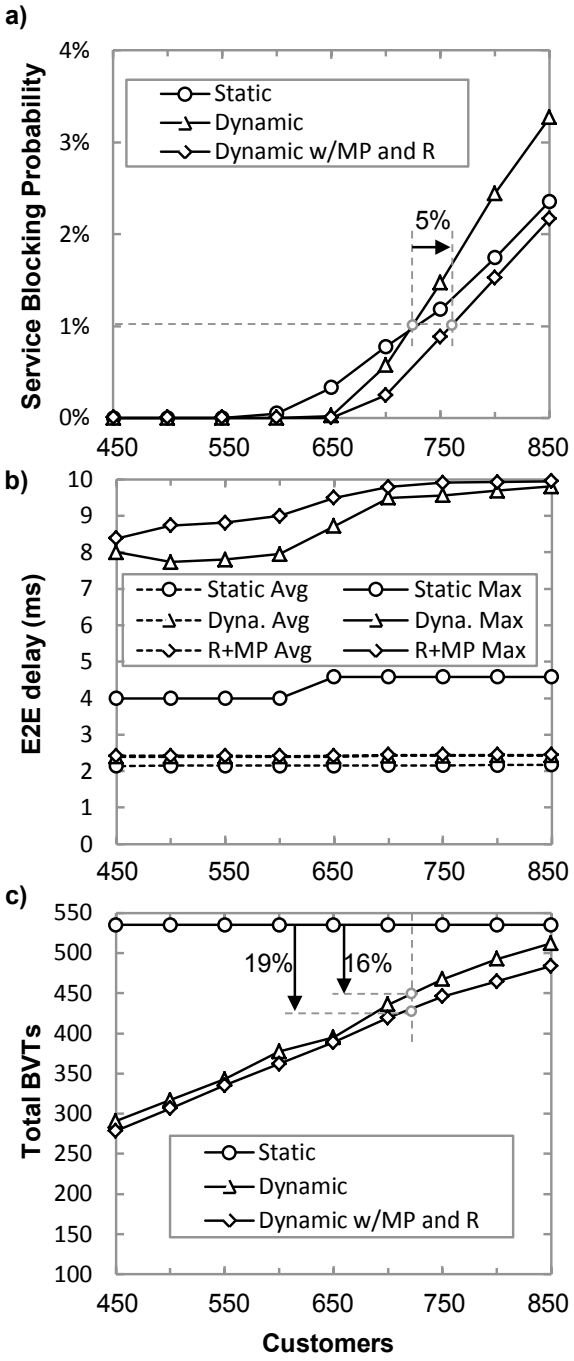

Fig. 9-4 Service blocking (a), end-to-end delay (b), and total number of BVTs (c) against number of clients

Regarding customers, two differentiated services have been considered: one requiring regional CVN topologies during office hours and 50% bitrate guaranteed;

the other requiring nation-wide CVN topologies during off-peak periods. EPs are connected to the closest IP/MPLS router. The maximum delay allowed between EPs is 10 ms. CVN reconfiguration requests arrive during service-defined periods, where the capacity requested between EPs is randomly chosen in the range [1-10] Gb/s. The offered load is related to the number of customers being served. Finally, CVN reconfiguration requests arrive to ASO following an exponential distribution with mean 1 hour.

Fig. 9-4a plots CVN service blocking probability as a function of the load. As commented above, the MPLS static and dynamic approaches perform the same for 1% blocking. When reoptimization and multipath options are applied, a gain of 5% in the offered load (at 1% blocking probability) is observed.

Fig. 9-4b focuses on QoS. As shown, the MPLS preplanning approach provides the best QoS for both average and maximum delay. However, dynamic approaches provide QoS under the specified maximum being on-average delay comparable to the static approach.

Finally, Fig. 9-4c concentrates on the total number of BVTs to install. As observed, the required number of BVTs increases with the load in the dynamic approaches, in contrast to the static one. CAPEX savings as high as 16% for the dynamic MPLS and 19% the dynamic with reoptimization approaches are shown.

9.3 Study of bitrate guarantees strategies

As previously introduced, service-specific SLAs become crucial to guarantee not only the required QoS, but also a minimum bitrate to minimize service interruption.

Two main schemes to provide bitrate guarantees are considered and compared in this section:

- *partial protection*, where a percentage of the bitrate requested for each CVN link is protected using an MPLS path which is SRLG-disjoint with the primary one. For instance, let us assume that a 10 Gb/s CVN link needs to be served guaranteeing 5 Gb/s in case of failure. Then, the primary MPLS path will convey 10 Gb/s, whereas the secondary protection MPLS path will convey 5 Gb/s;
- *diversity*, where the bitrate requested for each CVN link is divided and one MPLS path conveys the guaranteed bitrate, and the other path the rest of the CVN link capacity. In the previous example for 10 Gb/s CVN link provisioning, both MPLS paths will convey 5 Gb/s. Note that since the number of paths is restricted to two, the maximum guaranteed bitrate cannot exceed 50% of the total capacity.

An example to illustrate the impact of each scheme is depicted in Fig. 9-5. A four-node IP/MPLS network is represented, where each MPLS virtual link is supported by one or more lightpaths on the underlying optical network and its capacity is represented as *available capacity/total capacity* in Gb/s. For clarity purposes, the optical network is not depicted in the figure. EPs {A1, A2, A3} and {B1, B2} owned by two customers, A and B respectively, are connected to IP/MPLS routers. In the example, the MPLS path routing is shown for the protection (left) and diversity (right) schemes, and each CVN topology and links' capacity is shown.

In this small example, one can observe that, although MPLS routing is the same under both schemes (in general this is not true), the remaining capacity in the existing MPLS virtual links is not equivalent, since primary paths must convey all the requested CVN capacity under the protection scheme in contrast to only the guaranteed bitrate under the diversity one.

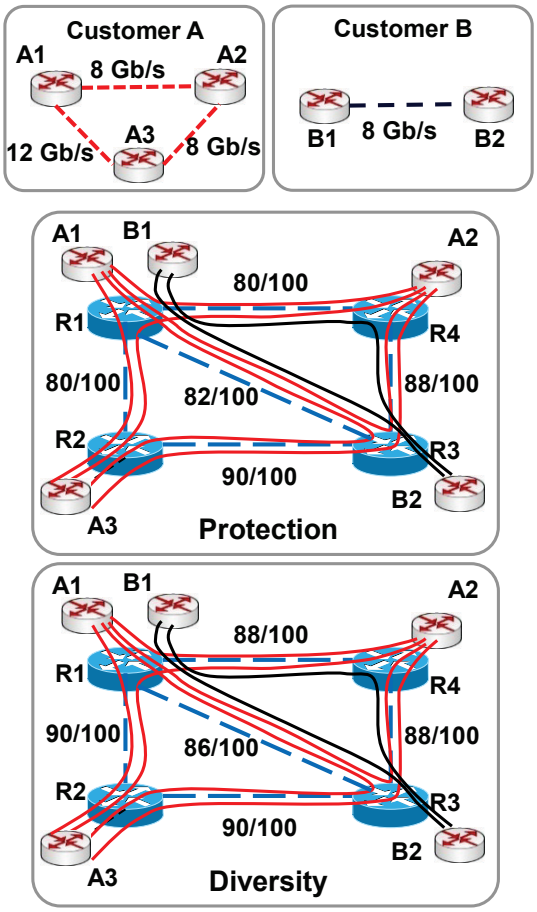

Fig. 9-5 Example of CVN based on protection and diversity schemes to support CVN links.

Notwithstanding the same amount of bitrate per CVN link is guaranteed under both schemes (i.e., 50%), the actual bitrate that is lost under a single link failure is very different. For instance, let us assume that every virtual MPLS link is

supported using one single link in the optical layer. In such very simple scenario, a failure in the optical link supporting MPLS link R1-R2 would impact primary paths A3-A1 and A3-A2, so 10 Gb/s will be lost under both the partial protection and the diversity schemes. However, a failure in the optical link supporting MPLS link R3-R4 would impact secondary paths A1-A2, A2-A3, and B1-B2, so no traffic is lost under the protection scheme, whereas 12 Gb/s will be lost under the diversity scheme. In this example, about 32% of the traffic will be lost on average under the diversity scheme, whereas only near 12% of the traffic will be lost under the partial protection scheme for single optical link failure scenarios.

The above might have implications in terms of CAPEX (e.g., amount of transponders to install), performance, and reliability. The next section formally states the CVN reconfiguration problem and presents methods for solving it.

9.3.1 CVN service reconfiguration with QoS constraints and bitrate guarantees (SQUBA)

In this section, the CVN service reconfiguration with QoS constraints and bitrate guarantees (SQUBA) problem is formally stated and MILP formulation to model it presented. In the view of its complexity and the short time in which it must be solved, a heuristic algorithm is also proposed.

A. Problem statement

The SQUBA problem can be formally stated as follows:

Given:

- a multilayer optical network represented by the graph $G^O(N, L)$; being N the set of optical nodes and L the set optical of links;
- a MPLS network represented by a graph $G^V(V, E)$; being V the set of MPLS nodes and E the set of MPLS virtual links, where each MPLS virtual link can be supported by several lightpaths.
- a set of customers C; each $c \in C$ manages its own CVN service, which topology is represented by a fully meshed graph $G_c(V_c, E_c)$; being V_c the set of EPs and E_c the set of CVN virtual links of c.
- a CVN service reconfiguration request coming from customer c that is represented by the tuple $<B^r, Q^r, W^r>$, where B^r is the capacity matrix of CVN links between EPs, Q^r is the QoS matrix, and W^r is the matrix with CVN links capacity to be guaranteed ($w^r \leq 0.5 \cdot b^r$).

Output: the set of MPLS paths over G^V and lightpaths over G^O that need to be established to serve the CVN reconfiguration request.

Objective: minimize the cost of the used resources.

B. Mathematical formulation

We assume that the flexgrid technology is used in the optical network, as stated in the introduction. In that regard, we solve the RSA following a link-path formulation based on the pre-computing frequency slots [VeCa14]. In addition, a parameter a is defined to represent whether diversity or protection is selected to support every CVN link.

Thus, the following sets and parameters have been defined:

N Set of optical nodes.

L Set of optical links, index l.

P Set of optical routes, index p.

S Set of frequency slices, index s.

K Set of frequency slots, index k.

G Set of SRLG identifiers, index g.

β Capacity of lightpaths in Gb/s.

δ_{pl} 1 if route p uses optical link l; 0 otherwise.

δ_{ks} 1 if slot k uses slice s; 0 otherwise.

δ_{ls} 1 if slice s in link l is available; 0 otherwise.

δ_{lg} 1 if optical link l is supported by SRLG g; 0 otherwise.

q_p Delay introduced by route p.

The following sets and parameters have been defined for the IP/MPLS network:

V Set of MPLS nodes.

E^+ Augmented set of virtual links to connect every pair of MPLS nodes, index e.

$E^+(v)$ Subset of E^+ with the virtual links incident to node v.

M Set of MPLS paths, index m.

$P(e)$ Subset of P with optical routes for MPLS virtual link e.

δ_{me} Equal to 1 if MPLS path m uses MPLS virtual link e; 0 otherwise.

δ_{eg} 1 if MPLS virtual link e is supported by SRLG g; 0 otherwise.

φ_e Available capacity in virtual link e; in Gb/s.

t_v Number of available transponders in MPLS node v.

q_V Delay introduced by MPLS routers.

q_e Delay introduced by MPLS virtual link e; 0 for those virtual links not supported by any lightpath.

The following sets and parameters have been defined for CVN c requesting a reconfiguration:

V_c Set of CVN c nodes.

$E_c = D$ Set of CVN virtual links, index d.

$M(d)$ Subset of M with the MPLS paths for CVN link d.

b_d Requested capacity for CVN link d.

q_d Requested delay for CVN link d.

w_d Requested capacity to be guaranteed for CVN link d.

Finally, the additional parameters have been defined:

κ_e Cost per Mb/s of using MPLS virtual link e.

κ_l Cost per Mb/s of using optical link l supporting an MPLS virtual link.

a 1 if *diversity* is considered; 0 for *protection*.

The decision variables are:

z_{dm} Binary. Equal to 1 if CVN link d is supported by MPLS path m as primary path.

r_{dm} Binary. 1 if MPLS path m is selected as the SRLG-disjoint MPLS path to guarantee w_d for CVN link d.

x_{mg} Binary. Equal to 1 if MPLS path m is supported by SRLG g.

z_{dmg} Binary. Equal to 1 if CVN link d is supported by primary path MPLS path m and SRLG g.

r_{dmg} Binary. Equal to 1 if CVN link d is supported by SRLG-disjoint MPLS path m and SRLG g.

u_{de} Continuous, with the amount of flow in MPLS virtual link e from CVN link d.

y_{epk} Binary. Equal to 1 if MPLS virtual link e uses optical route p and slot k.

h_e Continuous with the maximum delay of virtual link e.

h_m Continuous with the maximum delay of MPLS path m.

The MILP model for the SQUBA problem is as follows:

$$Minimize \quad \sum_{e\in E^+}\sum_{d\in D}\kappa_e \cdot u_{de} + \sum_{l\in L}\sum_{e\in E^+}\sum_{p\in P(e)}\sum_{k\in K}\kappa_l \cdot \delta_{pl} \cdot y_{epk} \tag{9.1}$$

subject to:

$$\sum_{m \in M(d)} z_{dm} = 1 \quad \forall d \in D \tag{9.2}$$

$$\sum_{m \in M(d)} r_{dm} = 1 \quad \forall d \in D \tag{9.3}$$

$$u_{de} = \sum_{m \in M(d)} \delta_{me} \cdot \left[\left(b_d - \alpha \cdot w_d \right) \cdot z_{dm} + w_d \cdot r_{dm} \right] \quad \forall e \in E^+, d \in D \tag{9.4}$$

$$\sum_{d \in D} u_{de} \le \varphi_e + \beta \cdot \sum_{p \in P(e)} \sum_{k \in K} y_{epk} \quad \forall e \in E^+ \tag{9.5}$$

$$\sum_{e \in E^+} \sum_{p \in P(e)} \sum_{k \in K} \delta_{ks} \cdot \delta_{pl} \cdot y_{epk} \le \delta_{ls} \quad \forall l \in L, s \in S \tag{9.6}$$

$$\sum_{e \in E^+(v)} \sum_{p \in P(e)} \sum_{k \in K} y_{epk} \le t_v \quad \forall v \in V \tag{9.7}$$

$$h_e \ge q_e \quad \forall e \in E^+ \tag{9.8}$$

$$h_e \ge \sum_{k \in K} q_p \cdot y_{epk} \quad \forall e \in E^+, p \in P(e) \tag{9.9}$$

$$h_m = \sum_{e \in E^+} \delta_{me} \cdot \left(q_V + h_e \right) + q_V \quad \forall m \in M \tag{9.10}$$

$$q_d + \left(1 - z_{dm} \right) \cdot bigM \ge h_m \quad \forall d \in D, m \in M(d) \tag{9.11}$$

$$q_d + \left(1 - \alpha \cdot r_{dm} \right) \cdot bigM \ge h_m \quad \forall d \in D, m \in M(d) \tag{9.12}$$

$$\sum_{e \in E^+} \delta_{me} \cdot \left(\delta_{eg} + \sum_{p \in P(e)} \sum_{k \in K} \delta_{pl} \cdot \delta_{lg} \cdot y_{epk} \right) \le bigM \cdot x_{mg} \quad \forall m \in M, g \in G \tag{9.13}$$

$$z_{dmg} + 1 \ge z_{dm} + x_{mg} \quad \forall d \in D, m \in M(d), g \in G \tag{9.14}$$

$$z_{dmg} \le z_{dm} \quad \forall d \in D, m \in M(d), g \in G \tag{9.15}$$

$$z_{dmg} \le x_{mg} \quad \forall d \in D, m \in M(d), g \in G \tag{9.16}$$

$$r_{dmg} + 1 \ge r_{dm} + x_{mg} \quad \forall d \in D, m \in M(d), g \in G \tag{9.17}$$

$$r_{dmg} \le r_{dm} \quad \forall d \in D, m \in M(d), g \in G \tag{9.18}$$

$$r_{dmg} \le x_{mg} \quad \forall d \in D, m \in M(d), g \in G \tag{9.19}$$

$$\sum_{m \in M(d)} z_{dmg} + \sum_{m \in M(d)} r_{dmg} \le 1 \quad \forall d \in D, g \in G \tag{9.20}$$

The objective function (9.1) minimizes the cost of using resources in MPLS virtual links and that of the optical resources to support new MPLS virtual links.

Constraints (9.2)-(9.7) deal with paths and lightpaths serving demands. Constraints (9.2) and (9.3) ensure that exactly two MPLS paths support each CVN link d disregarding whether diversity or protection is selected. Constraint (9.4) computes the amount of bitrate from a given demand that is conveyed through each virtual link. Constraint (9.5) ensures that every virtual link has enough aggregated capacity and forces a new lightpath to be set-up if necessary. Constraint (9.6) guarantees that every frequency slot is used by one new lightpath at most, provided it was unused. Constraint (9.7) limits the number of new lightpaths that are set-up to the number of available transponders

Constraints (9.8)-(9.12) ensure the requested QoS. Constraints (9.8) and (9.9) compute the maximum delay of every virtual link considering current delay and the delay of every new lightpath supporting such link, respectively. Constraint (9.10) computes the delay of every MPLS path. Constraints (9.11) and (9.12) guarantee that every demand is served using a primary MPLS path and a SRLG disjoint path if diversity is selected, such that their delay do not exceed the requested QoS.

Finally, constraints (9.13)-(9.20) deal with SRLGs. Constraint (9.13) computes SRLG support for every MPLS path. Constraints (9.14), (9.15), (9.16), (9.17), (9.18), (9.19) account for SRLG support for every MPLS path that is used to support a given demand. Constraint (9.20) ensures that the primary path and the SRLG-disjoint path are not supported by any common SRLG for every demand.

The SQUBA problem is *NP-hard* since simpler multilayer network problems have been proved to be *NP-hard* (e.g., [Zh11]). Regarding problem size, the number of variables is $O(|D| \cdot |M| \cdot |G| + |D| \cdot |E^+| + |E^+| \cdot |P| \cdot |K|)$ and the number of constraints is $O(|D| \cdot |M| \cdot |G| + |E^+| \cdot (|D| + |P|) + |L| \cdot |S| + |V|)$. Considering a realistic scenario on a network like those in the next section, the number of variables is in the order of 10^6 and the number of constraints in the order of 10^5, which makes the above MILP model unsolvable within the times required for serving on-demand CVN reconfiguration requests, even using state-of-the-art computer hardware and the latest commercially available solvers. As a result, a heuristic algorithm is needed aiming at providing near optimal solutions in realistic scenarios.

C. Heuristic Algorithm

We propose the random algorithm presented in Table 9-3 to solve the SQUBA problem. The algorithm is executed a fixed number of iterations and the best solution found is eventually returned.

Table 9-3 Algorithm for the SQUBA Problem

```
INPUT: Gc(Vc, Ec), Br, Qr, Wr, α
OUTPUT: Ω
 1:  D ← Ec, Ω ← Ø
 2:  for each e ∈ Ec do
 3:    if Br(e) ≤ B(e) and Qr(e) ≤ Q(e) and Wr(e) ≤ W(e) then
 4:      update(e, Br(e), Qr(e), Wr(e))
 5:    else
 6:      dealloc(e, Gc)
 7:      D ← D \ {e}
 8:  shuffle(D)
 9:  for each d ∈ D do
10:    ω ← setupCVNLink(d, Br(d), Qr(d), Wr(d), Gc, α)
11:    if ω = Ø then return Ø
12:    Ω ← Ω ∪ {ω}
13:  return Ω
```

The algorithm updates first those CVN links with unchanged or decreased requirements to release resources that can be reused afterwards (lines 2-4 in Table 9-3). The rest of the CVN links are de-allocated from G_c and added to the set D (lines 5-7). The set D is randomly sorted (line 8) and every CVN link is then set up (lines 9-12) using the *setupCVNLink* algorithm described in Table 9-4. Finally, the set Ω with the MPLS paths and lightpaths to be established is returned. Note that, if one of the CVN links cannot be updated, then the complete CVN reconfiguration request is blocked (line 11).

Table 9-4 setupCVNLink algorithm for the SQUBA Problem

```
INPUT: d, b, q, w, Gc, α
OUTPUT: Ω
 1:  Ω ← Ø
 2:  if α = 0 then bp = b
 3:  else bp = b-w
 4:  R ← findPath(d, bp, q)
 5:  if R = Ø then
 6:    Ω ← Ω ∪ setupMPLSLinks(d, bp, q)
 7:    R ← findPath(d, bp, q)
 8:    if R = Ø then return Ø
 9:  Ω ← Ω ∪ allocate(GV, d, R, bp)
10:  S ← findDisjointPath(d, q, R, w)
11:  if S = Ø then
12:    Ω ← Ω ∪ setupDisjointMPLSLinks(d, w, q, R)
13:    S ← findDisjointPath(d, w, q)
14:    if S = Ø then return Ø
15:  Ω ← Ω ∪ allocate(GV, d, S, w)
16:  return Ω
```

The *setupCVNLink* algorithm (Table 9-4) starts computing the capacity to be assigned to each MPLS path supporting the CVN link depending on the scheme considered. In the case that protection is selected, b is the bitrate to be carried by the primary path; otherwise, the required capacity for CVN links is split (lines 2-3).

Then, the algorithm finds an MPLS path with capacity b_p in G^V for the primary path to guarantee the requested QoS q (line 4). In the case no path is found, the algorithm tries to increase capacity in G^V so as to serve d by adding new MPLS virtual links and another try is performed. If no path is found, CVN link d cannot be served (lines 5-8). Otherwise, the primary path is allocated in G^V (line 9). Next, the algorithm tries to find an SRLG-disjoint MPLS path (lines 10-15). The set Ω with the MPLS paths and lightpaths to be established is eventually returned (line 16).

9.3.2 Illustrative results

Performance, CAPEX, and bitrate lost in case of single optical link failure for the two considered bitrate guarantees schemes are compared for two national network topologies: the 30-node Spanish Telefonica (TEL) and the 34-node France Telecom (FR) network topologies depicted in Fig. 9-6. No bitrate guarantees are additionally considered (one *single path* is set up to support CVN links) aiming at providing reference values. The heuristic described in the previous section and the management architecture including CNCs, ASO, and ABNO were developed in C++ and integrated in an ad-hoc event driven simulator based on OMNET++. 5-node CVNs topologies were considered; EPs were strategically placed in regions so as to cover the entire national geography. CVN reconfiguration requests were generated following an exponential distribution with mean 1 hour. Each CVN link's capacity varies according to a uniform distribution in the range [1,10] Gb/s. QoS constraints required by each customer were set to 10 ms. A CVN reconfiguration request is accepted if all the requested CVN links' capacity, QoS, and bitrate to be guaranteed are served; otherwise the complete request is blocked.

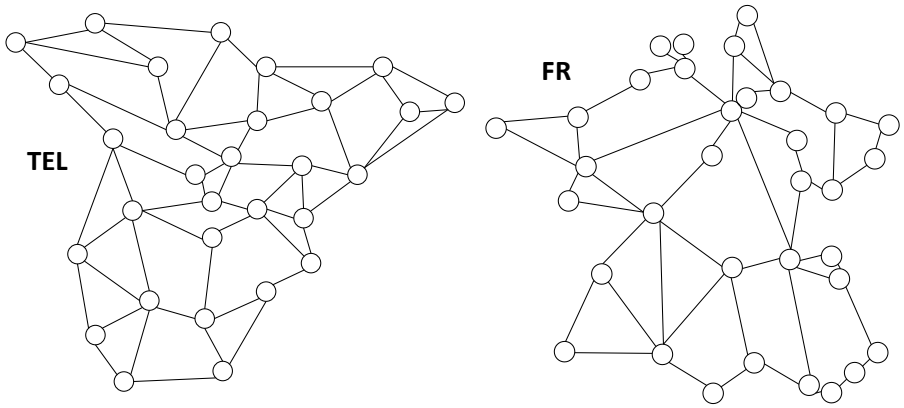

Fig. 9-6 Network topologies considered in the study.

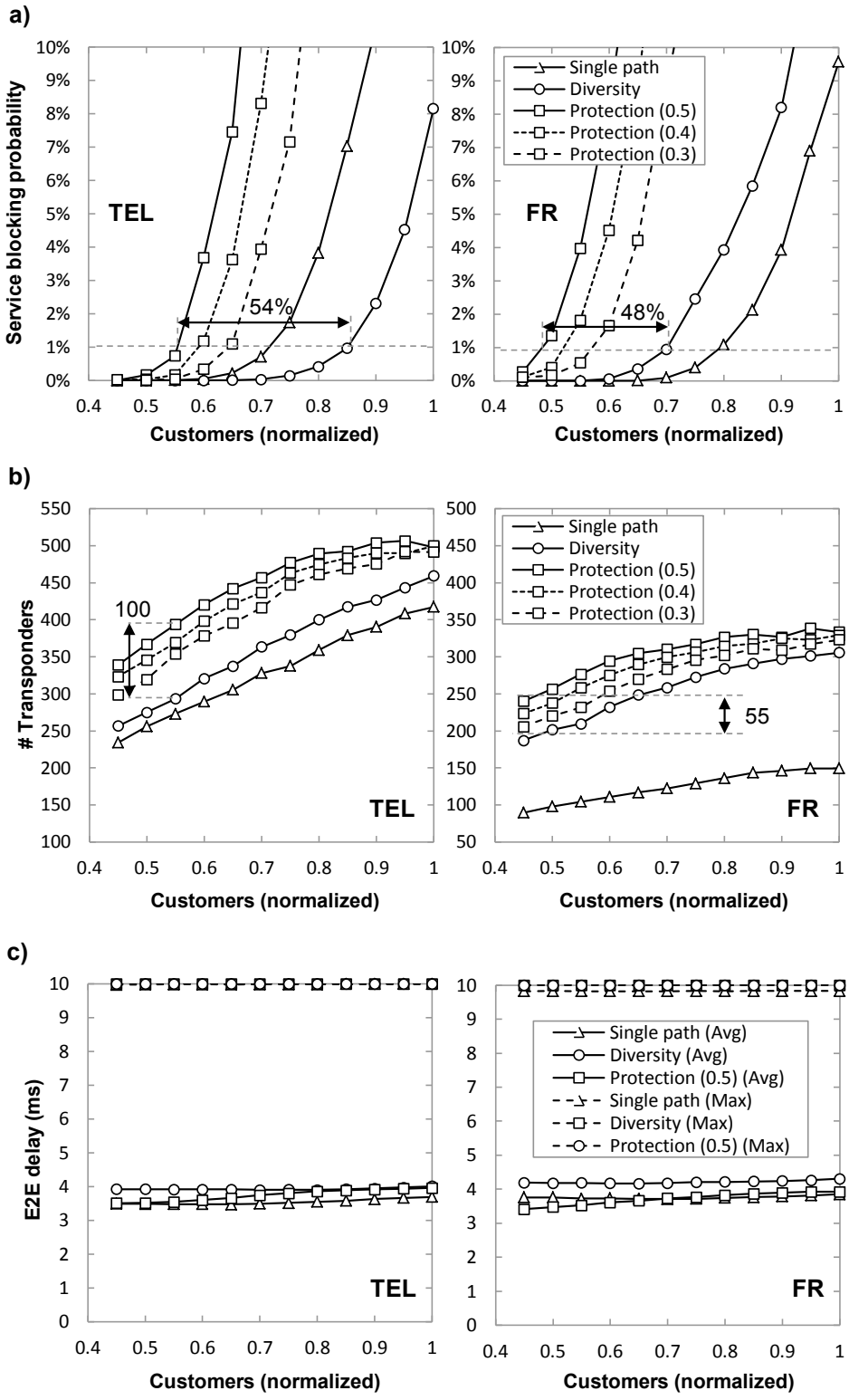

Fig. 9-7 Performance (a), total number of transponders (b), and e2e delay (c).

Regarding the transport network configuration, fiber links with spectrum width equal to 2 THz were considered, the spectral granularity was set to 6.25 GHz, and 100 Gb/s transponders using 37.5 GHz were installed in the MPLS routers [Mo15]. The number of transponders installed in each MPLS router was limited to 30. Finally, each point in the results is the average of 10 runs with more than 10,000 CVN reconfiguration requests requiring an increment of capacity at least in one CVN link.

Firstly, we study the impact of each scheme in terms of performance and CAPEX. Fig. 9-7 shows the obtained results in terms of service blocking probability, total number of used transponders, and average weighted e2e delay for the TEL and FR networks, against the number of customers (normalized).

As a result of the different bitrate requirements of the partial protection and diversity schemes (see example in Fig. 9-5), different service blocking probabilities are obtained. Fig. 9-7a shows gains in the offered load as high as 50% for the TEL and FR network topologies when the bitrate to be guaranteed is 50% of the required capacity. Although the obtained blocking probability improves when the bitrate to be guaranteed under the partial protection scheme decreases (40% and 30% of the required capacity), it is still far from that obtained under the diversity scheme. Interestingly, service with no bitrate guarantees does not always perform better than diversity in terms of service blocking probability, since benefits from using two MPLS paths (despite of being SRLG-disjoint) arise against using a single path.

Similarly, one can expect that the transponders to be installed in the network are higher under the partial protection scheme compared to the diversity one; Fig. 9-7b plots the number of transponders to be installed in each case. For the point of 1% blocking probability under the 50% partial protection, savings in the number of transponders as high as 100 and 55, in TEL and FR respectively, are observed when diversity is considered. Note also that the diversity scheme needs few more transponders than the no guarantees scheme.

Regarding the delay in CVN links, note that under normal conditions CVN link bitrate is conveyed by two MPLS paths under the diversity scheme (compared to only one path under partial protection). This fact might result in higher e2e delay. However, results in Fig. 9-7c show that, although average e2e delay is slightly lower under partial protection for low loads, both schemes perform similar. This is a consequence of the number of hops of primary and secondary MPLS paths under both schemes (Table 9-5). In general, the number of hops under the diversity scheme is slightly lower than under the partial protection scheme as a result of the latter requires more bitrate on the primary path. However, the average number needs to be considered for the diversity scheme, which results in a slightly higher number of hops and e2e delay.

Table 9-5 Number of Hops in MPLS Paths

		TEL Network					**FR Network**				
		# Customers (normalized)					**# Customers (normalized)**				
Scheme	**MPLS Path**	0.5	0.6	0.7	0.8	0.9	0.5	0.6	0.7	0.8	0.9
Single path	**Primary**	**2.22**	**2.21**	**2.22**	**2.24**	**2.29**	**1.36**	**1.35**	**1.34**	**1.33**	**1.34**
Partial Protection	**Primary**	**2.25**	**2.30**	**2.37**	**2.43**	**2.46**	**1.58**	**1.62**	**1.66**	**1.69**	**1.72**
	Secondary	2.82	2.84	2.86	2.87	2.91	2.35	2.36	2.36	2.37	2.37
Diversity	**Primary**	**2.08**	**2.10**	**2.10**	**2.11**	**2.13**	**1.49**	**1.48**	**1.48**	**1.49**	**1.51**
	Secondary	2.89	2.90	2.90	2.91	2.94	2.41	2.41	2.43	2.44	2.45
	Average	**2.49**	**2.50**	**2.50**	**2.51**	**2.54**	**1.95**	**1.95**	**1.95**	**1.96**	**1.98**

Let us now evaluate the performance of both schemes in the event of a single optical links failure. To that end, the network was firstly loaded and then, a failure was simulated in every optical links used to support MPLS virtual links; results are shown in Fig. 9-8 for the TEL and FR networks. Fig. 9-8a shows average values for the traffic lost per CVN and the total bitrate per CVN as a function of the load for the TEL and FR network topologies. Bitrate lost shows an almost constant behavior for all the considered loads. The diversity scheme, in addition to provide bitrate guarantees, improves bitrate loses even though it is more likely that a failure affects one of the two paths. For the protection scheme, bitrate lost slightly decreases as a consequence of the remarkable decrement in the average bitrate per CVN, direct result of the highest blocking probability of that scheme.

Finally, Fig. 9-8b shows the *bitrate lost*/*bitrate served* ratio as a function of the load. The ratio for the diversity scheme is under 12 and 18% for TEL and FR topologies respectively, much lower than that computed in the small example in Fig. 9-5, being under 6% for the restoration scheme. Note that the diversity scheme notably reduces the bitrate lost ratio compared to that from the no guarantees scheme.

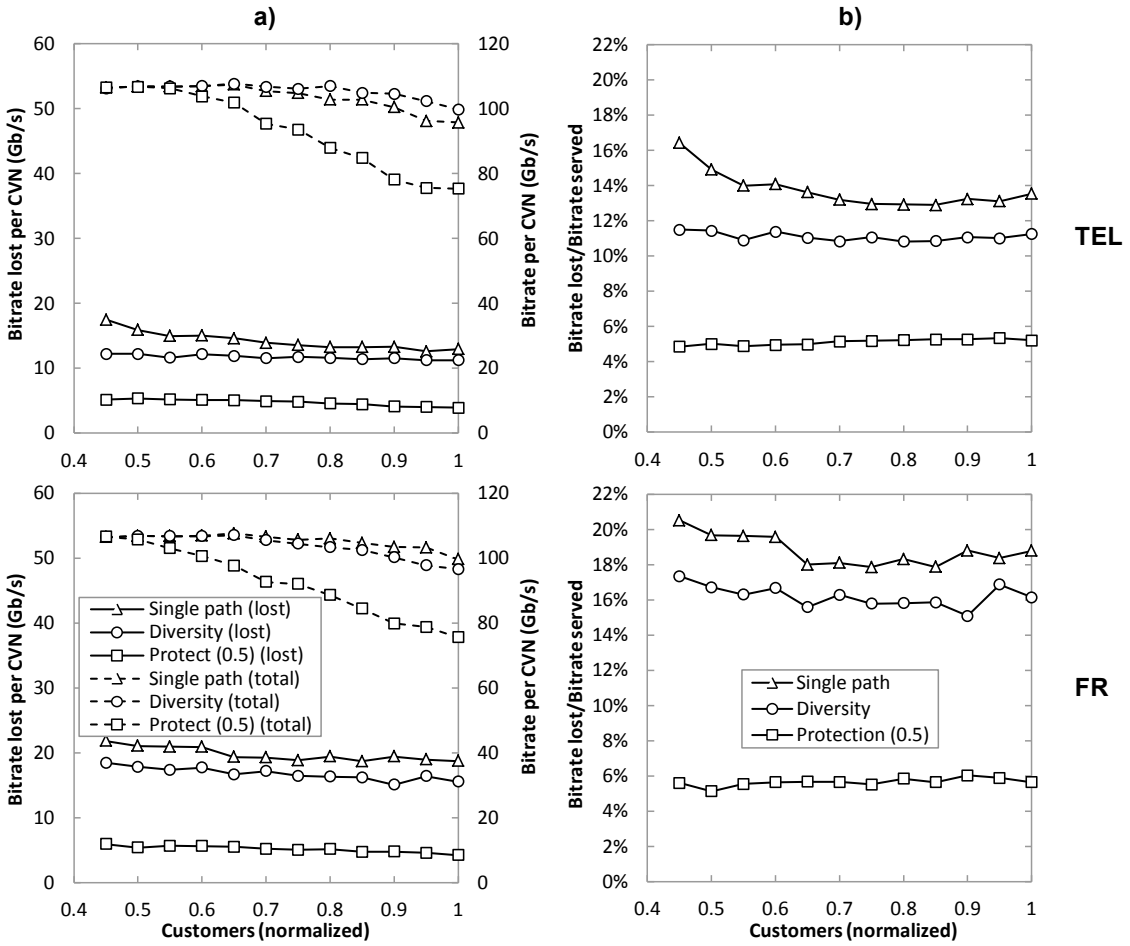

Fig. 9-8 Bitrate in service (a) and bitrate lost/bitrate in service (b).

9.4 Conclusions

An algorithm for dynamic CVN reconfiguration with QoS and bitrate guarantees has been proposed and evaluated through exhaustive simulations. Results showed CAPEX savings in the range from 16% to 19% under different dynamic strategies.

Then, considering dynamic strategies for CVN reconfiguration, we deepened into the problem of guaranteeing bitrate for CVNs. To guarantee the bitrate, network operators can take advantage of SRLG-disjoint paths to support CVN links. Two different approaches were considered to provide bitrate guarantees using SRLGs: partial protection and diversity. The former requires a primary MPLS path to convey the requested bitrate of a CVN link and a secondary path to provide bitrate guarantees in case of single failure, whereas the latter divides the requested bitrate into two SRLG-disjoint MPLS paths.

The CVN service reconfiguration with QoS constraints and bitrate guarantees (SQUBA) problem was formally stated and a MILP formulation to model the problem was presented. Nonetheless, solving times required for serving on-demand CVN reconfiguration requests makes the model impractical for realistic scenarios. As a result, a heuristic algorithm was developed aiming at providing a better tradeoff between optimality and computation time.

Partial protection and diversity strategies to support CVN links were afterwards compared through simulations on two real-size networks based on the Spanish Telefonica and the France Telecom national network topologies. For completeness, the no bitrate guarantees scheme was also considered. Strategies were compared in terms of performance, CAPEX (related to the number of transponders to equip), e2e delay, and bitrate guarantees in scenarios with single failures in optical links.

Results showed remarkable gains as high as 50% in the supported offered load (targeting at 1% of service blocking probability) when diversity was considered with respect to partial protection. Moreover, noticeable CAPEX savings in terms of required number of transponders were also shown. Interestingly, although partial protection results in slightly lower average e2e delay, no significant differences were obtained compared to diversity.

Finally, notwithstanding both strategies guarantee bitrate for each CVN link, partial protection performs better in terms of total bitrate lost/bitrate served.

Considering the performance shown, diversity is worth to be considered to support CVN reconfiguration with QoS and bitrate guarantees as an alternative to partial protection.

Chapter 10

Closing Discussion

10.1 Main contributions

The main contributions of this PhD thesis are:

- From the studies carried out in Chapter 4 and Chapter 5, we identified connectivity requirements for DC interconnection and to support services on the telecom cloud. Specifically, from the ELFADO problem, presented in Chapter 4, we showed the connectivity requirements that need to be considered to support elastic operations in DC federations and fulfilled goal G1.1. Regarding the connectivity requirements for services on the telecom cloud, first, live-TV distribution was studied in Chapter 5 and it helped to devise particular connectivity requirements to deploy services requiring strict QoS and the goal G1.2 was fulfilled. Next, we presented the CRAM optimization problem for network CAPEX minimization to support C-RAN in LTE-A scenarios. The telecom cloud infrastructure was considered to provide the required connectivity and to host the equipment needed. Goal G1.3 was achieved with this study and connectivity requirements for services requiring certain QoS and bitrate guarantees were shown.
- In Chapter 6, we proposed a CSO and two polling-based dynamic connectivity models (dynamic and dynamic elastic) for DC interconnection. The connectivity models were compared with the current static one. It was shown that both dynamic models presented remarkable bitrate savings against the static one, but it was demonstrated that dynamic connectivity is not enough to satisfy inter-DC connectivity to support elastic operations in DC federations due to the lack of network resources at requesting time and that dynamic elastic connectivity alleviates this to some extent. This fact motivated the study of the network ability to support elastic operations in

dynamic scenarios, and thus algorithms were proposed to increment or decrement network resources allocated to connections once in operation. With these studies, goal G2.1 and partially goal G3 were fulfilled.

- In Chapter 7, the ELFADO problem was extended with a stochastic version of it, where solar green energy availability was predicted using a stochastic model, thus resulting in STC-ELFADO. In addition, to improve network resources availability, a novel notification-based model was proposed, where some information about network resources availability is shared from the network operator with the DC operator. The notification-based model was compared to the polling-based, dynamic elastic, connectivity model to solve STC-ELFADO and resulted in noticeable costs savings. From the network operator side, the proposed notification-based model is useful when connectivity can be specified in terms of volume of data and completion time.
- Considering the notification-based model, in Chapter 8, the transfer-based connectivity model was proposed (where connectivity can be specified in terms of volume of data and completion time), an ASO, and formally stated the RSSA problem. The RSSA problem was modeled using an ILP formulation and a heuristic algorithm was proposed. It was shown that transfer-based connectivity not only reduces time-to-transfer but also improves network performance (more connections can be accepted) while fulfilling SLAs of a set of customers simultaneously, resulting beneficial for both DC and network operators. The studies carried out in the works in Chapter 7 and Chapter 8 contributed to achieve goals G2.2 and partially goal G3.
- Finally, in Chapter 9, based on the ACTN framework, it was proposed to use an ASO to manage CVNs requiring not only dynamic e2e connectivity but also certain QoS and bitrate guarantees. An algorithm was presented for dynamic MPLS virtual network reconfiguration and it was compared to a static approach where the MPLS virtual network was pre-planned. Results showed remarkable network CAPEX savings when dynamic virtual network reconfiguration was considered. Then, two approaches for guaranteeing bitrate: partial protection and diversity, were compared. Diversity resulted in noticeable performance gains and CAPEX savings while e2e delay kept similar to the approach where partial protection was considered. Results obtained in this chapter fulfilled goal G2.3 and contributed to fulfill goal G3.

10.2 List of Publications

10.2.1 Publications in Journals

[As15.1] **A. Asensio**, M. Ruiz, and L. Velasco, "Orchestrating Connectivity Services to support Elastic Operations in Datacenter Federations," Springer Photonic Network Communications, vol. 29, pp. 291-306, 2015.

[As14.1] **A. Asensio** and L. Velasco, "Managing Transfer-based Datacenter Connections," IEEE/OSA Journal of Optical Communications and Networking, vol. 6, pp. 660-669, 2014.

[Ve14.1] L. Velasco, **A. Asensio**, J. Ll. Berral, E. Bonetto, F. Musumeci, V. López, "Elastic Operations in Federated Datacenters for Performance and Cost Optimization," Elsevier Computer Communications, vol. 50, pp. 142-151, 2014.

[Ve14.2] L. Velasco, **A. Asensio**, J.Ll. Berral, A. Castro, V. López, "Towards a Carrier SDN: An example for Elastic Inter-Datacenter Connectivity," (Invited Paper) OSA Optics Express, vol. 22, pp. 55-61, 2014.

[Ve13.1] L. Velasco, **A. Asensio**, J.Ll. Berral, V. López, D. Carrera, A. Castro, and J.P. Fernández-Palacios, "Cross-Stratum Orchestration and Flexgrid Optical Networks for Datacenter Federations," IEEE Network Magazine, vol. 27, pp. 23-30, 2013.

10.2.2 Publications in Conferences

[As16.1] **A. Asensio,** M. Ruiz, and L. Velasco, "Connectivity Requirements for Cloud-based Services," accepted in IEEE International Conference on Transparent Optical Networks (ICTON), 2016.

[As16.2] **A. Asensio**, M. Ruiz, and L. Velasco, "Requirements to Support Cloud, Video and 5G Services on the Telecom Cloud," accepted in European Conference on Network and Optical Communications (NOC), 2016.

[As16.3] **A. Asensio**, P. Saengudomlert, M. Ruiz, and L. Velasco, "Study of the Centralization Level of Optical Network-Supported Cloud RAN," accepted in IEEE International Conference on Optical Network Design and Modeling (ONDM), 2016.

[As16.4] **A. Asensio**, M. Ruiz, and L. Velasco, "Bitrate Guarantees Strategies for Customer Virtual Networks Dynamic Reconfiguration," in Proc. in Design of Reliable Communication Networks (DRCN), 2016.

[As15.2] **A. Asensio**, M. Ruiz, L. M. Contreras, L. Velasco, and G. Junyent, "Dynamic Customer Virtual Network Reconfiguration with QoS Constraints and Bandwidth Guarantees", in Proc. European Conference on Optical Communication (ECOC), 2015.

[As15.3] **A. Asensio**, M. Ruiz, and L. Velasco, "Routing and Scheduled Spectrum Allocation for Transfer-based Datacenter Connections," in Proc. IEEE International Conference on Transparent Optical Networks (ICTON), 2015.

[As15.4] **A. Asensio**, L. M. Contreras, M. Ruiz, V. Lopez, and L. Velasco, "Scalability of Telecom cloud Architectures for Live-TV Distribution," in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), 2015.

[As14.2] **A. Asensio**, Ll. Gifre, M. Ruiz, and L. Velasco, "Carrier SDN for Flexgrid-based Inter-Datacenter Connectivity," in Proc. IEEE International Conference on Transparent Optical Networks (ICTON), 2014.

[As14.3] **A. Asensio**, L. Velasco, M. Ruiz, and G. Junyent, "Carrier SDN to Control Flexgrid-based Inter-Datacenter Connectivity," in Proc. IEEE International Conference on Optical Network Design and Modeling (ONDM), 2014.

[Ve13.2] L. Velasco, **A. Asensio**, J.Ll. Berral, A. Castro, and V. López, "Towards a Carrier SDN: An example for Elastic Inter-Datacenter Connectivity," in Proc. European Conference on Optical Communication (ECOC), 2013.

[As13.1] **A. Asensio**, A. Castro, L. Velasco, and J. Comellas, "An Elastic Netwoks OMNET++ -based Simulator," in Proc. IEEE International Conference on Transparent Optical Networks (ICTON), 2013.

[As13.2] **A. Asensio**, M. Klinkowski, M. Ruiz, V. López, A. Castro, L. Velasco, and J. Comellas, "Impact of Aggregation Level on the Performance of Dynamic Lightpath Adaptation under Time-Varying Traffic," in Proc. IEEE International Conference on Optical Network Design and Modeling (ONDM), 2013.

10.2.3 Other publications

[Lo16] A. Stavdas, C. Matrakidis, M. Gunkel, **A. Asensio**, L. Velasco, E. Varvarigos, K. Christodoulopoulus, "Taking Advantage of Elastic Optical Networks," In: V. López and L. Velasco (eds.), "Elastic Optical Networks: Architectures, Technologies, and Control", accepted in Optical Networks book series, Springer, 2016.

10.3 List of Research Projects

During the PhD thesis, I collaborated in both European and National funded projects. In particular, the research work of the PhD thesis has received funding from the projects listed below:

10.3.1 European funded projects

IDEALIST: "Industry-Driven Elastic and Adaptive Lambda Infrastructure for Service and Transport Networks", Ref: FP7-ICT-2011-8, 2012-2015. Web: http://www.ict-idealist.eu

10.3.2 Spanish funded projects

ELASTIC: "Enhanced Optical Networks Featuring Adaptable and Highly Scalable Multi-Granular Transport", Ref: TEC2011-27310, 2012-2014.

SYNERGY: "Service-oriented Hybrid Optical Network and Cloud Infrastructure Featuring High Throughput and Ultra-low Latency", Ref: TEC2014-59995-R, 2015-2017.

10.4 Research stay

To complement the research carried out at Universistat Politècnica de Catalunya, a 3-month research stay was done in Bangkok University at the Center of Research in Optoelectronics, Communications and Control Systems (BU-CROCCS), under the guidance of Dr. Poompat Saengudomlert.

Dr. Poompat Saengudomlert holds a B.S.E. degree in Electrical Engineering from Princeton University, USA, and a M.S. and Ph.D. degrees, both in Electrical Engineering and Computer Science, from the Massachusetts Institute of Technology (MIT), USA. Dr. Poompat Saengudomlert is the author of the book "Optimization for Communications and Networks" [Sa11].

The research carried out during this stay was focused on C-RAN, resulting in the work presented in Chapter 5.

10.5 Topics for further research

The connectivity models devised in this thesis jointly with the proposed cross-stratum orchestration to satisfy connectivity requirements from cloud services, open the opportunity to further explore, and realize, NFV in the telecom cloud. In fact, new PhD thesis have been recently started within the Optical Communications Group that will use the results from this thesis as a starting point.

List of Acronyms

ABNO	Application-Based Network Operations
ACTN	Abstraction and Control of Transport Networks
AIP	Aggregation Infrastructure Power
AR	Advance Reservation
ASO	Application Service Orchestrator
BBU	Baseband Unit
BT	British Telecom
BV-OXC	Bandwidth-Variable Optical Cross-Connect
BVT	Bandwidth-Variable Transponders
CAPEX	Capital Expenditures
CF	Central Frequency
CNC	Customer Network Controller
CO	Central Office
CPRI	Common Public Radio Interface
CRAM	Cloud-Radio Access Network Capital Expenditures Minimization
C-RAN	Cloud-Radio Access Network
CSO	Cross-Stratum Orchestrator
CVN	Customer Virtual Network
CVN-QBG	Customer Virtual Network Reconfiguration with Quality of Service constraints and Bitrate Guarantees
DB	Database

DC	Datacenter
DC2DC	Datacenter-to-Datacenter
U2DC	User-to-Datacenter
DOPN	Dynamic Optical Path Network
DT	Deutsche Telekom
DWDM	Dense Wavelength Division Multiplexing
E2E	End-to-end
EB	Exabyte
ELFADO	Elastic Operations in Federated Datacenters for Performance and Cost Optimization
EON	Elastic Optical Network
EP	End-Point
EVPI	Expected Value of Perfect Information
FR	France Telecom
FT	Fixed Transponders
GB	Gigabyte
Gb/s	Gigabits per second
GbE	Gigabit Ethernet
GHz	Gigahertz
GMT	Greenwich Mean Time
HD	High Definition
IaaS	Infrastructure as a Service
IETF	Internet Engineering Task Force
ILP	Integer Linear Programming
IP	Internet Protocol
IP/MPLS	IP-over-MPLS
IT	Information Technology
k-PSK	k- Phase-Shift Keying
k-QAM	k- Quadrature Amplitude Modulation
kWh	Kilowatt hour
L2	Layer 2

LAN	Local Area Network
LSP	Label-Switched Path
LSP-DB	Label-Switched Path- Database
LTE	Long-Term Evolution
LTE-A	Long-Term Evolution -Advanced
Mb/s	Megabits per second
MBS	Macro Base Station
MHz	Megahertz
MILP	Mixed Integer Linear Programming
MIMO	Multiple Input Multiple Output
MME	Mobility Management Entity
MPLS	Multi-Protocol Label Switching
NaaS	Network as a Service
NFV	Network Functions Virtualization
NMS	Network Management System
NSI	Network Services Interface
OAM	Operation, Administration and Management
OGF	Open Grid Forum
O-OFDM	Optical OFDM
OPEX	Operational Expenditures
OXC	Optical Cross-Connects
PCE	Path Computation Element
PM	Physical Machine
QoE	Quality of Experience
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RAN	Radio Access Network
RF	Radio Frequency
RRH	Remote Radio Head
RSA	Routing and Spectrum Allocation
RSSA	Routing and Scheduled Spectrum Allocation

RTT	Round Trip Time
RWA	Routing and Wavelength Assigment
SA	Spectrum Allocation
SaaS	Software as a Service
SBVT	Sliceable Bandwidth-Variable Transponders
SD	Standard Definition
SD-EON	Software Defined – Elastic Optical Network
SDI	Serial Digital Interface
SDN	Software Defined Network
S-GW	Serving Gateway
SLA	Service Level Agreement
SQUBA	CVN service reconfiguration with QoS constraints and bitrate guarantees
SRLG	Shared Risk Link Group
STC-ELFADO	Stochastic-ELFADO
sTED	Scheduled Traffic Engineering Database
Tb/s	Terabits per second
TCO	Total Cost of Ownership
TCP	Transmission Control Protocol
TED	Traffic Engineering Database
TEL	Telefonica
TP	Traffic Profile
UB	Unserved bitrate
UHD	Ultra-High Definition
USA	United States of America
VM	Virtual Machine
VNF	Virtual Network Functions
VNT	Virtual Network Topology
VON	Virtual Optical Network
WAN	Wide Area Network
WDM	Wavelength Division Multiplexing

References

[Ar10] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Communications of the ACM, vol. 53, pp. 50-58, 2010.

[Bu13] J Buysse, K. Georgakilas, A. Tzanakaki, M. Leenheer, B. Dhoedt, and C. Develder, "Energy-Efficient Resource-Provisioning Algorithms for Optical Clouds," IEEE/OSA Journal of Optical Communications and Networking, vol. 5, no. 3, pp. 226-239, 2013.

[Ca12] A. Castro, L. Velasco, M. Ruiz, M. Klinkowski, J. P. Fernández-Palacios, and D. Careglio, "Dynamic Routing and Spectrum (Re)Allocation in Future Flexgrid Optical Networks," Elsevier Computers Networks, vol. 56, pp. 2869-2883, 2012.

[Ca14] N. Carapellese, M. Tornatore and A. Pattavina, "Energy-Efficient Baseband Unit Placement in a Fixed/Mobile Converged WDM Aggregation Network," IEEE Journal on Selected Areas in Communications, vol. 32, no. 8, pp. 1542-1551, 2014.

[Ce15] D. Ceccarelli and Y. Lee, "Framework for Abstraction and Control of Transport Networks," IETF draft, 2015.

[Ch11] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Dynamic bandwidth allocation in flexible OFDM-based networks," in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), 2011.

[Ch13] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Time-Varying Spectrum Allocation Policies and Blocking Analysis in Flexible Optical Networks," IEEE Journal on Selected Areas in Communications, vol. 31, no. 1, pp.13-25, 2013.

[Ch14] A. Checko, H. Holm, and H. Christiansen, "Optimizing small cell deployment by the use of C-RANs," in Proceedings of European Wireless 2014, 20th European Wireless Conference, vol., no., pp.1-6, 2014.

[CMR] "C-RAN the road towards green RAN," China Mobile Research, 2011.

[Co12] L. M. Contreras, V. López, O. González, A. Tovar, F. Muñoz, A. Azañón, J.P. Fernández-Palacios, and J. Folgueira, "Towards Cloud-Ready Transport Networks," IEEE Communications Magazine, vol. 50, pp. 48-55, 2012.

[CPLEX] CPLEX, http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/.

[CPRI] http://www.cpri.info/downloads/CPRI_v_7_0_2015-10-09.pdf

[Cr13] E. Crabbe, I. Minei, J. Medved, and R. Varga, "PCEP Extensions for Stateful PCE," IETF draft, April 2015.

[Dh12] D. Dhody, Y. Lee, and H. Yang, "Cross Stratum Optimization (CSO) Enabled PCE Architecture," in Proc. IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA), 2012

[Di02] J. Díaz and E. Fernández, "A Branch-and-Price Algorithm for the Single Source Capacitated Plant Location Problem," Journal of the Operational Research Society (JORS), vol. 53, pp. 728-740, 2002.

[EEP] Europe's Energy Portal: http://www.energy.eu/

[Fa07] X. Fan, W. Weber, and L. A. Barroso, "Power Provisioning for a Warehouse-sized Computer", In Proc. of ACM International Symposium on Computer Architecture (ISCA), 2007.

[Fa08] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in Proc. ACM SIGCOMM, 2008.

[G694.1] "Spectral grids for WDM applications: DWDM frequency grid," ITU-T Recommendation G.694.1 (ed. 2.0), 2012.

[Ga14] M. Gattulli, M. Tornatore, R. Fiandra, and A. Pattavina, "Low-Emissions Routing for Cloud Computing in IP-over-WDM Networks with Data Centers," IEEE Journal on Selected Areas in Communications, vol. 32, no. 1, pp. 28-38, 2014.

[GCI12] Cisco, "Global Cloud Index", 2012.

[Ge07] D. Gesbert, M. Kountouris, R. W. Heath Jr., C. b. Chae, and T. Salzer, "Shifting the MIMO Paradigm," IEEE Signal Processing Magazine, vol. 24, no. 5, pp. 36–46, Sep. 2007.

[Ge12] O. Gerstel, M. Jinno, A. Lord, and S. Ben Yoo, "Elastic Optical Networking: A New Dawn for the Optical Layer?," IEEE Communications Magazine, vol. 50, pp. s12-s20, 2012.

[Go10] I. Goiri, J. Guitart, and J. Torres, "Characterizing Cloud Federation for Enhancing Providers' Profit," in Proc. IEEE International Conference on Cloud Computing (CLOUD), 2010

[Go12] I. Goiri, K. Le, T. D. Nguyen, J. Guitart, J. Torres, and R. Bianchini, "GreenHadoop: Leveraging Green Energy in Data-Processing Frameworks," in Proc. ACM European Conference on Computer Systems (EuroSys), 2012.

[Gr04] W.D. Grover, “Mesh-based Survivable Networks,” Prentice Hall PTR, New Jersey, 2004.

[GreenGrid] The Green Grid: www.thegreengrid.org/

[GSNFV] ETSI, GSNFV. Network Functions Virtualisation (NFV): Architectural Framework. ETSI GS NFV, 2013.

[Ha07] E. Harney, S. Goasguen, J. Martin, M. Murphy, and M. Westall, “The Efficacy of Live Virtual Machine Migrations over the Internet,” in Proc. VTDC, 2007

[He12] W. Van Heddeghem, F. Idzikowski, W. Vereecken, D. Colle, M. Pickavet, and P. Demeester, “Power consumption modeling in optical multilayer networks,” Springer Photonic Network Communications, vol. 24, pp 86-102, 2012.

[Ja13] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, U. Hölzle, S. Stuart, and A. Vahdat, “B4: Experience with a Globally-Deployed Software Defined WAN,” in Proc. SIGCOMM, 2013

[Ji09] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, and S. Matsuoka, “Spectrum-Efficient and Scalable Elastic Optical Path Network: Architecture, Benefits, and Enabling Technologies”, IEEE Communications Magazine, Vol. 47, pp. 66-73, 2009.

[Ji10] M. Jinno, B. Kozicki, H. Takara, A. Watanabe, Y. Sone, T. Tanaka, and A. Hirano, “Distance-adaptive spectrum resource allocation in spectrum-sliced elastic optical path (SLICE) network,” IEEE Communications Magazine, vol. 48, pp. 138-145, 2010.

[Ji12] M. Jinno, H. Takara, Y. Sone, K. Yonenaga, and A. Hirano, “Multiflow Optical Transponder for Efficient Multilayer Optical Networking,” IEEE Communications Magazine, vol. 50, no. 5, pp. 56-65, 2012.

[Ki04] D. King, W. Boyson, J. Kratochvil, “Photovoltaic Array Performance Model,” Sandia National Laboratories Report, SAND2004-3535, 2004.

[Kl13] M. Klinkowski, M. Ruiz, L. Velasco, D. Careglio, V. Lopez, and J. Comellas, “Elastic spectrum allocation for time-varying traffic in flexgrid optical networks,” IEEE Journal on Selected Areas in Communications, vol. 31, pp. 26-38, 2013.

[Ku13] T. Kudoh, G. Roberts, and I. Monga, “Network Services Interface: An Interface for Requesting Dynamic Inter-datacenter Networks,” in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), 2013

[Ku15] T. Kurosu, S. Suda, H. Kakibayashi, K. Solis-Trapala, J. Kurumida, K. Ishii, K. Tanizawa, and S. Namiki, “What is the true value of dynamic optical path switching?,” Opto-Electronics and Communications Conference (OECC), 2015.

[KuTa13] K. Tanizawa, S. Namiki, R. Akimoto, H. Kuwatsuka, T. Hasama, H. Ishikawa, T. Nakatogawa, K. Oyamada, Y. Tanaka, S. Ide, H. Onaka, and T. Asami, "Dynamic Optical Path Switching in 172-Gb/s OTDM Transmissions of Ultra-High Definition Video Signals Using Fast Channel-Identifiable Clock Recovery and Integratable Devices," Journal of Lightwave Technology, vol. 31, no. 4, pp. 594-601, Feb.15, 2013.

[Li02] G. Li, J. Yates, D. Wang, and C. Kalmanek, "Control plane Design for Reliable Optical Networks," IEEE Communications Magazine, vol. 40, no. 2, Feb. 202, pp.90-96.

[Li09] L. Liu, H. Wang, X. Liu, X. Jin, W. He, Q. Wang, and Y. Chen, "GreenCloud: A New Architecture for Green Data Center," in Proc. IEEE International Conference on Autonomic Computing and Communications Industry Session (ICAC-INDST), 2009.

[Li10] Y. Lin, L. Shao, Z. Zhu, Q. Wang and R. Sabhikhi, "Wireless network cloud: Architecture and system requirements," IBM J. of Research and Development, vol.54, no.1, pp.4:1-4:12, 2010.

[Li11] Z. Liu, M. Lin, A. Wierman, S. Low, and L. Andrew, "Geographical Load Balancing with Renewables," in Proc. of ACM Greenmetrics, 2011.

[Li15] S. Li, W. Lu, X. Liu, and Z. Zhu, "Fragmentation-aware service provisioning for advance reservation multicast in SD-EONs," Optics Express 23, 25804-25813 (2015)

[Lu13] W. Lu and Z. Zhu, "Dynamic Service Provisioning of Advance Reservation Requests in Elastic Optical Networks," IEEE/OSA Journal of Lightwave Technology, vol. 31, pp. 1621-1627, 2013.

[Lu14] W. Lu, S. Ma, C. Chen, X. Chen, and Z. Zhu, "Implementation and Demonstration of Revenue-Driven Provisioning for Advance Reservation Requests in OpenFlow-Controlled SD-EONs," IEEE Communications Letters, vol 18, no. 10, 2014.

[MATLAB] www.mathworks.com/products/matlab/

[METAWIKI] Meta-Wiki. http://meta.wikimedia.org/wiki/User:Stu/comScore_data_on_Wikimedia#Geographic_breakdown.

[Mi12] M. Mishra, A. Das, P. Kulkarni, and A. Sahoo, "Dynamic Resource Management Using Virtual Machine Migrations," IEEE Communications Magazine, vol. 50, pp. 34-40, 2012.

[Mo15] A. Morea, J. Renaudier, T. Zami, A. Ghazisaeidi, and O. Bertran-Pardo, "Throughput comparison between 50-GHz and 37.5-GHz grid transparent networks," IEEE/OSA J. of Optical Communications and Networking, vol.7, pp. A293-A300, 2015

[Mu16] F. Musumeci, C. Bellanzon, N. Carapellese, M. Tornatore, A. Pattavina, and S. Gosselin, “Optimal BBU placement for 5G C-RAN deployment over WDM aggregation networks,” Journal of Lightwave Technology vol. 34, no. 8, pp. 1963-1970, April15, 15 2016.

[Na11] S. Namiki, T. Kurosu, K. Tanizawa, J. Kurumida, T. Hasama, H. Ishikawa, T. Nakatogawa, M. Nakamura, and K. Oyamada, “Ultrahigh-Definition Video Transmission and Extremely Green Optical Networks for Future”, IEEE Journal of Selected Topics in Quantum Electronics, vol.17, no.2, pp.446,457, 2011.

[OGF] https://www.ogf.org

[OMNETPP] OMNET++, http://www.omnetpp.org/

[ONEBULA] OpenNebula, http://www.opennebula.org/

[ONF] “Open Networking Foundation,” https://www.opennetworking.org/

[OPENFLOW] OpenFlow. http://www.openflow.org/

[OSTACK] OpenStack, http://www.openstack.org/

[Pf15] T. Pfeiffer, “Next generation mobile fronthaul and midhaul architectures,” IEEE/OSA Journal of Optical Communications and Networking, vol.7, pp. B38-B45, 2015.

[Pi11] J.M. Pierson, “Green Task Allocation: Taking into Account the Ecological Impact of Task Allocation in Clusters and Clouds,” Journal of Green Engineering, vol. 1, pp. 129–144, 2011.

[Po13] F. Ponzini, L. Giorgi, A. Bianchi, and R. Sabella, “Centralized radio access networks over wavelength-division multiplexing: a plug-and-play implementation,” IEEE Communications Magazine, vol.51, no.9, pp.94-99, 2013.

[Po14] F. Poulin, T. Kernen, and A. Kouadio, “Ultra High Definition TV Over IP Networks,” EBU Technical Review, 2014.

[Ra13] F. Rambach, B. Konrad, L. Dembeck, U. Gebhard, M. Gunkel, M. Quagliotti, L. Serra, and V. Lopez, “A multilayer cost model for metro/core networks,” IEEE/OSA Journal of Optical Communications and Networking, vol.5, pp.210-225, 2013.

[RFC3031] E. Rosen, A. Viswanathan, and R. Callon, “M ultiprotocol Label Switching Architecture,” IETF RFC3031, 2001.

[RFC4655] A. Farrel, J.-P. Vasseur, and J. Ash, “A Path Computation Element (PCE)-Based Architecture,” IETF RFC4655, 2006.

[RFC7491] D. King and A. Farrel, “A PCE-Based Architecture for Application-Based Network Operations,” IETF RFC7491, 2015.

[RuPi13] M. Ruiz, M. Pióro, M. Zotkiewicz, M. Klinkowski, and L. Velasco, "Column Generation Algorithm for RSA Problems in Flexgrid Optical Networks," Springer Photonic Network Communications, vol. 26, pp. 53-64, 2013.

[RuVe13] M. Ruiz, L. Velasco, J. Comellas, G. Junyent, "A Traffic Intensity Model for Flexgrid Optical Network Planning under Dynamic Traffic Operation," in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), 2013.

[Sa11] P. Saengudomlert, "Optimization for Communications and Networks." Science Publishers, Enfield, NH, USA, 2011.

[Se01] P. Sebos, J. Yates, G. Hjálmtýsson, A. Greenberg, "Auto-discovery of Shared Risk Link Groups," Optical fiber Commun. Conf., March 2001.

[Sh09] A. Shapiro, D. Dentcheva, and A. Ruszczynski, "Lectures on Stochastic Programming: Modeling and Theory," MPS-SIAM, 2009.

[Sh10] N. Sharma, J. Gummeson, D. Irwin, and P. Shenoy, "Cloudy Computing: Leveraging Weather Forecasts in Energy Harvesting Sensor Systems," in Proc. SECON, 2010.

[Sh11] G. Shen; Q. Yang, S. You, and W. Shao, "Maximizing time-dependent spectrum sharing between neighboring channels in CO-OFDM optical networks," in Proc. IEEE International Conference on Transparent Optical Networks (ICTON), 2011.

[Sh13] S. Shen, W. Lu, X. Liu, L. Gong and Z. Zhu, "Dynamic advance reservation multicast in data center networks over elastic optical infrastructure," Optical Communication (ECOC 2013), 39th European Conference and Exhibition on, London, 2013, pp. 1-3.

[St01] J. Strand, A. Chiu, and R. Tkach, "Issues for routing in the optical layer," IEEE Communications Magazine, February 2001, pp. 81-87.

[ST2036-1] ST 2036-1 "Ultra High Definition Television —Image Parameter Values for Program Production," SMPTE, 2009.

[Tz13] A. Tzanakaki, Markos P. Anastasopoulos, and K. Georgakilas, "Dynamic Virtual Optical Networks Supporting Uncertain Traffic Demands," IEEE/OSA J. of Optical Communications and Networking, vol.5, 2013.

[Tz14] A. Tzanakaki, M. Anastasopoulos, K. Georgakilas, G. Landi, G. Bernini, N. Ciulli, J. F. Riera, E. Escalona, J. A. Garcia-Espin, X. Hesselbach, S. Figuerola, Shuping Peng, R. Nejabati, D. Simeonidou, D. Parniewicz, B. Belter, and J. Rodriguez Martinez, "Planning of dynamic virtual optical cloud infrastructures: The GEYSERS approach," IEEE Communications Magazine, vol. 52, no. 1, pp. 26-34, 2014.

[USAEP] U.S. Department of Labor, Average Energy Prices, http://www.bls.gov/ro5/aepchi.htm

[USDOE] US Department of Energy, http://apps1.eere.energy.gov/buildings/energyplus/weatherdata_about.cfm

[USEIA] US Department of Energy. US Energy Information Administration. 2009. Web site http://www.eia.doe.gov/.

[VeCa14] L. Velasco, A. Castro, M. Ruiz, and G. Junyent, "Solving Routing and Spectrum Allocation Related Optimization Problems: from Off-Line to In-Operation Flexgrid Network Planning," [Invited] IEEE/OSA Journal of Lightwave Technology (JLT), vol. 32, pp. 2780-2795, 2014.

[VeCo15] L. Velasco, L.M. Contreras, G. Ferraris, A. Stavdas, F. Cugini, M. Wiegand, and J. P. Fernández-Palacios, "A Service-Oriented Hybrid Access Network and Cloud Architecture," IEEE Communications Magazine, vol. 53, pp. 159-165, 2015.

[VeGo14] L. Velasco, O. González de Dios, V. López, J. Fernández-Palacios, and G. Junyent, "Finding an Objective Cost for Sliceable Flexgrid Transponders," in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), 2014.

[VeKi14] L. Velasco, D. King, O. Gerstel, R. Casellas, A. Castro, and V. López, "In-Operation Network Planning," IEEE Communications Magazine, vol. 52, pp. 52-60, 2014.

[VeKl12.1] L. Velasco, M. Klinkowski, M. Ruiz, and J. Comellas, "Modeling the Routing and Spectrum Allocation Problem for Flexgrid Optical Networks," Springer Photonic Network Communications, vol. 24, pp. 177-186, 2012

[VeKl12.2] L. Velasco, M. Klinkowski, M. Ruiz, V. López, and G. Junyent, "Elastic Spectrum Allocation for Variable Traffic in Flexible-Grid Optical Networks," in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), 2012.

[VERIZON] Verizon, http://www.verizonbusiness.com/about/network/latency/

[VNI13] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2014–2019," CISCO Whitepaper, Feb. 2013.

[VNI14] Cisco, "Cisco Visual Networking Index," 2014.

[WIKISIZE] Wikipedia, http://en.wikipedia.org/wiki/Wikipedia_talk:Size_of_Wikipedia#size_in_GB

[Wu15] J. Wu, Z.F. Zhang, Y. Hong, and Y.G. Wen, "Cloud radio access network (C-RAN): a primer," IEEE Network Magazine, vol.29, pp.35-41, 2015.

[Yo14] M. Yoshinari, Y. Ohsita, and M. Murata, "Virtual network reconfiguration with adaptability to traffic changes," IEEE/OSA J. of Optical Communications and Networking, vol.6, pp.523-535, 2014.

[Zh11] X. Zhang, F. Shen, L. Wang, S. Wang, L. Li, and H. Luo, "Two-layer mesh network optimization based on inter-layer decomposition," Springer Photonic Network Communications, vol. 21, pp. 310-320, 2011

[ZhAn13] Y. Zhang and N. Ansari, "On Architecture Design, Congestion Notification, TCP Incast and Power Consumption in Data Centers," IEEE Communications Surveys & Tutorials, vol. 15, pp. 39-64, 2013.

[ZhVu13] X. Zhao, V. Vusirikala, B. Koley, V. Kamalov, and T. Hofmeister, "The Prospect of Inter-Data-Center Optical Networks," IEEE Communications Magazine, vol. 51, pp. 32-38, 2013.

[ZhZh13] J. Zhang, Y. Zhao, H. Yang, Y. Ji, H. Li, Y. Lin, G. Li, J. Han, Y. Lee, and T. Ma, "First demonstration of enhanced Software Defined Networking (eSDN) over elastic Grid (eGrid) Optical Networks for Datacenter Service Migration," in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), PDP5B.1, 2013.