Traffic Allocation Strategies in WSS-based Dynamic Optical Networks [Invited]

Ali Shakeri, Miquel Garrich, Anderson Bravalheri, Davide Careglio, Josep Solé-Pareta, and Andrea Fumagalli

***Abstract* — Elastic optical networking (EON) is a viable solution to meet future dynamic capacity requirements of Internet Service Provider (ISP) and inter-datacenter networks. At the core of EON, Wavelength Selective Switches (WSSs) are applied to individually route optical circuits, while assigning an arbitrary bandwidth to each circuit. Critically, the WSS control scheme and configuration time may delay the creation time of each circuit in the network. In this paper, we first detail the WSS-based optical data-plane implementation of a metropolitan network test-bed. Then, we review an SDN application, designed to enable dynamic and fast circuit setup. Subsequently, we introduce a WSS logical model that captures the WSS time-sequence and is used to estimate the circuit setup response time. Then, we present two batch service policies that aim to reduce the circuit-setup response time by bundling multiple WSS reconfiguration steps into a single SDN command. Resulting performance gains are estimated through simulation.**

***Index Terms* — Wavelength Selective Switch, Circuit switching; Optical Networks; Software Defined Networking.**

I. Introduction

Elastic optical networking (EON) is a viable solution to accommodate future capacity requirements in metro networks using sliceable bandwidth-variable transponders and flexible-grid reconfigurable optical add/drop multiplexers (ROADMs) [1]. Specifically, ROADMs commonly use liquid crystal on silicon (LCoS)-based wavelength selective switches (WSSs) capable of independently switching and routing fiber spectrum slices of 6.25 GHz [2]. Indeed, this EON flexibility of spectrum usage can increase the efficiency of the network fiber infrastructure by better exploiting the spectrum of the already deployed optical fibers [1]. Additionally, dynamic circuit allocation strategies may provide additional benefits avoiding the waste of static circuits underutilized out of peak-hours. Indeed, dynamic label-switched path (LSP) provisioning is attracting interest to address the communication needs in ISP and datacenter networks [3], [4]. In particular, current commercial products for intra-data center networks make use of Micro-Electro-Mechanical Systems (MEMS) fiber switches solutions [5], [6]. These solutions offer fast switching capabilities in the order of tens of milliseconds, but do not permit wavelength signals to be routed independently [5], [6]. On the contrary, WSS-based solutions enable wavelengths to be switched individually [7], [8]. However, the drawback of these latter solutions is the relatively slow switching time of the WSS devices. Unless adequately addressed, this drawback may prevent the use of WSS technology in support of future metropolitan network scenarios or intra-data center network characterized by highly dynamic data flows.

In this context, control-plane response time (including signaling latency) and WSS configuration time in the data-plane are the two major contributors to the overall service provisioning time (SPT), which delays the establishment of operational LSPs and limits the optical network efficiency. The latency factor introduced by the control-plane signaling has been extensively studied in the literature for both WDM networks [9] and EONs [10] considering a Generalized Multi-Protocol Label Switching (GMPLS) control-plane. Regarding the data-plane, both ROADMs and optical amplifiers (which are required to compensate for signal power loss) must be carefully configured to guarantee reliable LSPs. Specifically, WSSs and their low-level control firmware require a certain configuration time to carry out the commands received from the control-plane. Their impact on the SPT has not yet been sufficiently addressed in the literature.

Software-defined networking (SDN) provides a (logical) global network view by abstracting both the data-plane infrastructure and WDM equipment [11], [12]. Thus, SDN may help mitigate the adverse effect of both control-plane reservation and data-plane configuration times on SPT.

Manuscript received October 5 2016.

Ali Shakeri and Andrea Fumagalli are with OpNeAR Lab, Erik Jonsson School of Engineering and Computer Science, The University of Texas at Dallas, Richardson, Texas, USA.

Miquel Garrich is with Dipartimento di Elettronica e Telecomunicazioni, Politecnico di Torino, Italy and with CPqD.

Anderson Bravalheri is with CPqD, rua Dr. Ricardo Benetton Martins, s/n, 13086-902, Campinas, SP, Brazil.

Davide Careglio and Josep Solé-Pareta are with the Advanced Broadband Communications Center, Universitat Politècnica de Catalunya, Barcelona, Spain.

This paper first reviews the proposal and preliminary experimental results reported in [13]. In particular, we explore the benefits that SDN provides when configuring data-plane devices in support of dynamic traffic allocation in an EON-enabled metropolitan network. We review a recently proposed *global strategy* for controlling the data-plane based on a single Network-wide Agent and compare it against a *distributed strategy* comprising multiple per-node ROADM Agents capable of establishing each hop of an LSP in *parallel* or *sequentially*. Moreover, with a *debouncing*[1] enhancement technique, we achieve an efficient control- and data-plane interaction. We also evaluate the SPT using two proposed service batch policies that bundle WSS operations in a single command, namely, *Distinct Batch Service* (DBS) and *Single Batch Service* (SBS) policy. A simulation platform captures the essence of the experimental test-bed time-consuming procedures and is leverage to estimate the average SPT. Simulation results for the two proposed service batch policies help identify the optimal number of commands per batch that minimizes the average SPT.

II. Related Work

The majority of related work on optical LSP dynamic provisioning can be divided in two major categories.

A. ISP assumptions / requirements

LSP dynamic provisioning and fast restoration are desirable characteristics in metropolitan/backbone EONs. Studies in the literature commonly rely on dynamic traffic assumptions to analyze key performance indicators network-wide. For instance, [10] proposes a routing and spectrum assignment scheme that includes advanced transmission technologies assuming connections that last one-hour on average using the Spanish nation-wide topology as reference. Instead, Lu *et al.* [14] assume an EON operating in time slots (TS) to provision immediate and advance reservation requests (avg. duration of 5 and10 TS, respectively) guaranteeing the LSP configuration time within one TS [14]. In particular, they discuss that practical values of the TS duration would be in the order of tens of minutes in the NSFNET topology.

It is also worthwhile to mention that Giorgetti *et al.* [15] investigate the impact of the control-plane implementation (GMPLS and SDN) on LSP restoration considering the PAN-European network. They report average recovery times in the order of seconds at different network loads, node configuration and path computation times.

In summary, most related works in the literature assume connection requests that form a Poisson arrival process and require a holding time with an exponential distribution. However, in ISP topologies, practical values for these assumptions commonly rely on inter-DC connectivity requirements. Indeed, network architectures have been recently proposed to jointly address ISP and inter-DC needs [3] and their impact on wide-area networks discussed in [4].

B. Inter-datacenter assumptions / requirements

Inter-DC communication and dynamic circuit provisioning have motivated many research efforts in recent years. In 2010, GRIPhoN (globally reconfigurable intelligent photonic network) was proposed as an advanced ROADM architecture able to improve the performance and operational flexibility of optical transport networks [16], and was then extensively studied to provide bandwidth on demand for inter-DC connectivity through backbone optical networks [17]. Moreover, optical circuit switching has been proposed to combine both inter- and intra-DC connectivity for long-lived flows [18]. Indeed, dynamic optical provisioning has been proposed for DCs using micro-electro-mechanical systems (MEMS) in Helios [19] and cThrough [20] architectures; and with a combination of MEMS and WSS in Proteus [5] and OSA [8].

In summary, optical technologies for DC connectivity are considered as viable solutions to fulfill inter-DC [21] and intra-DC [22] highly dynamic traffic requirements.

III. Optical Data-Plane Configuration

In this section, we first provide an overview of the equipment in our optical network test-bed including the WSS device specifications. Subsequently, we detail the essential elements that comprise the optical data-plane. Finally, characterization of the WWS device response time is carried out experimentally.

A. Optical network test-bed overview

The experimental results of this work are obtained using the Autonomous Network (AN), an SDN-enabled five-node metropolitan optical network test-bed located at CPqD [23]. More specifically, the AN comprises 4 ROADMs of degree 3 and a central ROADM of degree 4 interconnected to form a partial mesh topology using 100-km single mode fiber (SMF) links as shown at the bottom of Fig. 2(a). The ROADM node architecture is a broadcast-and-select structure using one splitter per input port and one WSS per output port. EDFAs are placed at each input and output port to compensate for span and node losses. No physical dispersion compensation modules are used. Transmission impairments and nonlinear effects are assumed to be compensated at the receiver (out of the scope of this work). Prior works address nonlinear impairments compensation [24], carrier recovery [25] and chromatic dispersion estimation and compensation [26]. Related works with additional details of the AN test-bed include SDN applications for advanced WSS-based equalization [27], joint optimization of EDFA and WSS [28] and overshoot suppression when undesired transient responses occur in optical networks composed of WSS-based ROADMs [29].

The AN WSS devices are from Finisar and belong to its *ROADMs & Wavelength Management* product portfolio.

[1] *Debouncing* usually refers to the technique that ensures only one action is triggered upon multiple signals generated from a single event. For instance, a key press on a computer keyboard generates multiple signals and a debouncing technique ensures that only one digital signal can be registered by the computer within the space of a given time (usually milliseconds).

More specifically, 1 × 5 Flexgrid® WSSs[2] acquired in 2010 are used in the central ROADM, whereas 1 × 4 Flexgrid® WSSs[3] acquired in 2012 are used in the ROADMs at the edges of the network.

B. WSS configuration implementation

WSS devices are usually controlled through proprietary interfaces and protocols. Firmware modules can be implemented to hide custom solution from the SDN controlled, as shown in Fig. 1(a). This figure shows the data-plane access through a local software layer, i.e., *firmware* module, in our optical AN test-bed. An abstraction of the hardware equipment is created using a YANG [30] model consisting on a set of (potentially nested) attributes including procedures related to its control. From the YANG model, a framework automatically builds a RESTful application programming interface – API [31]. Callbacks are triggered at each attribute change invoking native hardware component commands to access the physical device.

C. WSS response time

Three major contributors compose the optical data-plane response time required to operate a WSS given, for instance, an LSP establishment command from the control-plane. First, the *physical* procedure required by the LCoS technology requires a configuration time due to the free-space optics mechanisms to apply a variable optical attenuation (VOA) at each 6.25-GHz spectrum slice. Second, the *firmware* execution may increase the configuration time adding an overhead due to the protocol translation and interaction with caching mechanisms. Third, the *communication* response time between the control-plane location and the particular hardware component to be configured may be seen as an additional overhead. In our test-bed, such communication channel is a Layer 2 Ethernet network that replicates the actual WDM topology.

Fig. 1(b) shows the WSS configuration time measured from the SDN controller via REST to switch (circles) and (un)block[4] (squares) different number of channels (solid lines and full symbols). Results report the average of ten configuration times of three different WSS devices. Additionally, the manufacturer specifications of the maximum hardware switching and (un)blocking times are shown as benchmarking reference (dashed lines and empty symbols, denoted as "Specs."). Therefore, the manufacturer specifications can be considered as an upper bound to the device contribution to the response time because they correspond to the WSS direct configuration via the manufacturer protocol (see Fig. 1(a)). Indeed, WSS prototype response times are around 0.1 second for an (un)block operation of a single channel [32]. It is relevant to

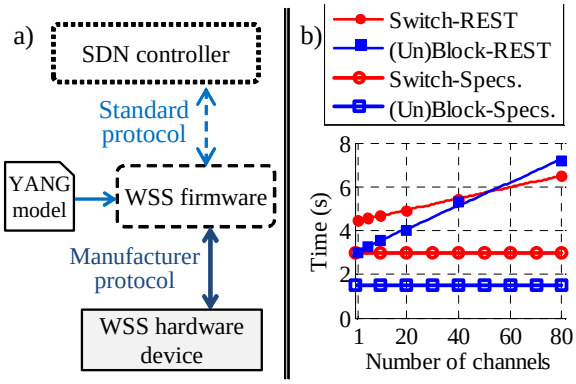

Fig. 1.(a) Communication between the SDN controller and data-plane (WSS hardware) using a YANG-based firmware approach. (b) Wavelength selective switch (WSS) configuration time for switching (circles) and (un)blocking (squares) for different number of channels; solid lines with full symbols are for configuration time from the SDN controller via REST interface and dashed lines with empty symbols denote manufacturer specifications of the hardware response time.

recall that manufacturer specifications usually report a maximum time that corresponds to commands operating the entire C band, e.g., 80 fixed-grid ITU-T channels. However, the actual measured values exhibit rather large fluctuations upper-bounded by the specifications. The variation depends on the temperature and the number of LCoS switching steps (i.e., hardware response time) that may be required to complete one operation [33].

Both switch and (un)block configuration times clearly present an overhead time on the manufacturer specifications due to the communication and firmware execution. Indeed, both configuration times using REST exhibit a linear dependence on the number of channels to be switched and (un)blocked with minimal standard deviation (confidence intervals using vertical lines not reported for the sake of legibility). We have applied a simple linear regression model to our experimental data (solid lines in Fig. 1(b)). The outcome of this model is used as input for the simulation analysis reported in Sections VI and VII, denoted as $X_{WSS}(W)$, which is the WSS response time to configure a batch of size W wavelengths.

Finally, note that the response time for an (un)block operation of a single channel is around three seconds. Therefore, operating on 40 channels sequentially (e.g. set up or tear down an LSP) would require around 2 minutes. In contrast, Fig. 1(b) shows that only five seconds are needed if the parallel (un)blocking operation is adopted. This result clearly motivates our debouncing proposal of Section IV.C.

IV. SDN Application for Traffic Allocation

In this Section, we detail the dynamic traffic loader application that runs on top of our SDN controller (see upper part of Fig. 2(a)). Such application receives as input a list of connection requests containing their creation time, duration, source and destination nodes. Moreover, it is possible to provide within the list of connection requests their path (intermediate nodes) and channel. In case this information is not provided, the application will execute the routing and wavelength assignment (RWA). Results are

[2] Product Code: 10WSPA05ZZL. Discontinued product. Preliminary version of the current 1 × 9 and 1 × 20 WSS devices detailed in https://www.finisar.com/roadms-wavelength-management/10wsaaxxfll

[3] Product Code: EWP-AA-104-96F-ZZ-L https://www.finisar.com/roadms-wavelength-management/ewp-aa-010x-96f-zz-l

[4] Block is the action of tearing down an established optical channel by applying the maximum attenuation, whereas the unblock action sets up an optical channel reducing the attenuation from the maximum to a given value. Both operations required the same time, reported as (un)block.

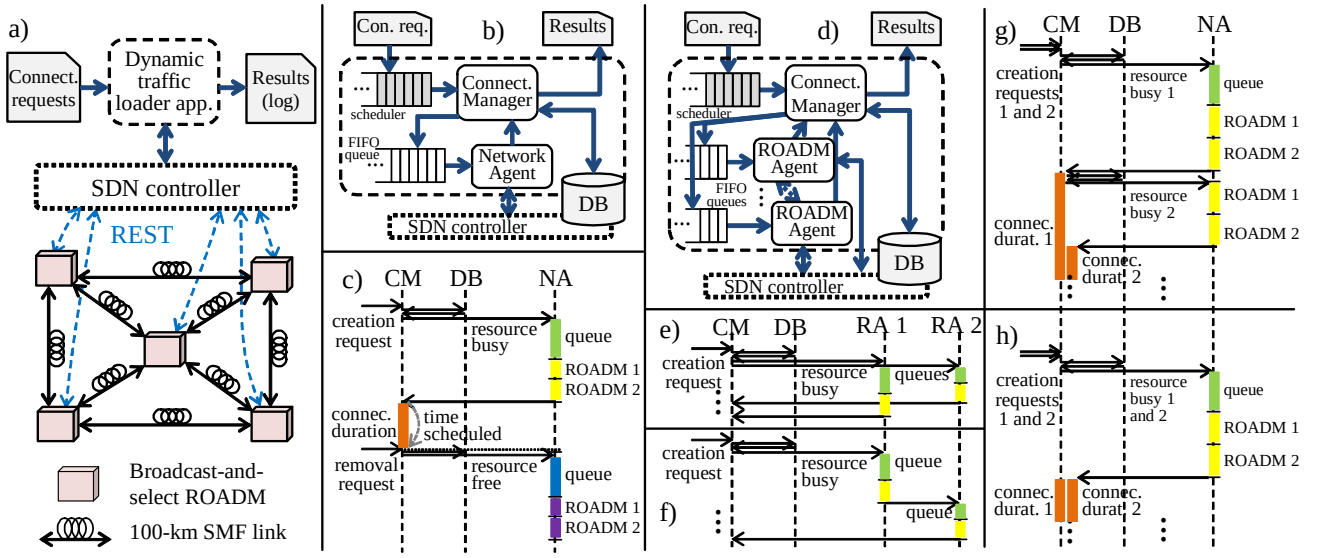

Fig. 2.(a) CPqD's metropolitan mesh optical network test-bed comprising five ROADMs and the SDN controller. Dynamic traffic loader application with (b) global and (d) distributed strategies. Events for a single 2-hop connection request using the (c) global, (e) distributed-parallel, and (f) distributed-sequential modes, respectively. Events for two 2-hop connection requests in the same path using the global strategy (g) without and (h) with the debouncing technique. Connect./Con. req.: Connection requests, CM: Connection Manager, DB: Database, NA: Network Agent, RA 1/2: ROADM Agent 1/2, act.: actuation.

stored in a log file. The SDN dynamic traffic application and equipment firmware run in conventional commodity hardware with an Ubuntu 12.04 LTS operating system. A personal computer (Intel® Core™ i7 2.8 GHz CPU with 8 GB RAM) is used as host for the SDN application, while the firmware is executed in commercial embedded microcomputers (Beaglebone® ARM® Cortex-A8 720-MHZ CPU with 256 MB RAM).

A. *Global strategy*

Fig. 2(b) presents the implementation of the dynamic traffic loader application for the global strategy. The connection requests are loaded into a scheduler, according to its expected creation moment. The *Connection Manager* (CM) uses an internal representation of the network state provided by a database (DB) that belongs to the SDN controller and observes the scheduler to allow or deny connections based on the availability of resources. The CM sends configuration requests to the *Network Agent* (NA) using a first-in-first-out (FIFO) queue. The NA actuates in a single ROADM of the network at a time by communicating with the SDN controller to perform the configuration of the devices in the data-plane.

Fig. 2(c) shows the sequence of operations for the global strategy handling a two-hop connection request (ROADMs 1 and 2). First, the CM receives a creation request and queries the DB to check the availability of resources (path and channel). Since the resources are available, creation operations of each connection hop are inserted in the NA queue and, simultaneously, the corresponding resources are updated as busy in the DB. Then, these operations wait a certain *creation queue time* until they are handled by the NA, who then orders the *creation actuation* at ROADM 1 and 2 sequentially. A notification is sent from the NA back to the CM when the actuation at the last ROADM is completed. Then, the CM inserts in the scheduler a removal request according to the *connection duration*. When the removal request is triggered, the CM frees the resources in the DB and sends the operations to the NA analogously as for the creation request. The removal operation may wait in the FIFO queue of the NA (*removal queue time*) until actuation at the ROADMs is allowed (*removal actuation*).

B. *Distributed Strategy*

Fig. 2(d) shows the implementation of the dynamic traffic loader application for the distributed strategy. A similar implementation of the CM communicates with independent *ROADM Agents* (RAs) who execute the create/remove operations at each data-plane switching device via the SDN controller and using independent FIFO queues. Two different operation modes handle multiple-hop connections.

Fig. 2(e) shows the operations in the *Parallel* mode only for the creation request considering the previous two-hop connection example. The CM receives the creation request, queries the DB, updates the resources as busy, and one creation operation is inserted at each RA queue. The creation operation of each hop waits at each RA queue independently until actuation is allowed. When the actuations of all the ROADMs are completed, the CM inserts in the scheduler a removal request according to the connection duration. ROADM actuations may not occur in the sequence of the connection path such as illustrated in Fig. 2(e).

Similarly, Fig. 2(f) shows the operations in the *Sequential* mode where communication between RAs follows the path of the connection. Indeed, the CM now sends the creation operation only to the first RA of the path, who sends a creation operation to the next RA after its actuation. A notification of LSP establishment is sent to the CM when the actuation is completed at the last RA. Ordered ROADM configurations provided by this mode benefit transient suppression control mechanisms and network stability [29]. In particular, authors in [29] experimentally demonstrated

that cascaded WSS-based ROADMs using conventional controllers may amplify optical power fluctuations leading to undesired overshoots. Therefore, an ordered sequence of configurations for cascaded WSS-based ROADMs offer stability of the signal power (during LSP creation) since a hop-by-hop operation may avoid the need for power readjustment. Moreover, this sequential mode can be seen as an improved alternative to the Resource Reservation Protocol (RSVP)[5] in GMPLS.

In case creation requests arrive during removal queue and actuation times, the FIFO queues used in both strategies guarantee that removal actions occur first thus avoiding potential conflicts.

C. Debouncing technique

A simple implementation of the proposed strategies and operation modes is to realize the establishment of each hop in each connection by sending a new command to the corresponding data-plane equipment. By doing so, a series of requests to (de)activate different channels in a given WSS may be generating unnecessary control- and data-plane interaction just to perform incremental changes. Moreover, as observed in Section II, the complexity of the change (number of channels simultaneously configured) does not strongly influence actuation time. However, the data-plane response time perceived at the control-plane notably increases as the number of requests accumulates.

The debouncing technique takes advantage of the centralized view of the SDN controller, by grouping series of incremental commands in a single, atomic, physical change. This enhancement is implemented by accepting a grouping time margin in the scheduling procedure. In particular, given that a single operation command will experience at minimum an (un)block configuration time (see Fig. 1(b) for one channel), a simple rule to determine the grouping time is to set it equal to the (un)block operation duration. Consequently, the experiments reported in the following used a debouncing grouping time of 3 seconds.

Fig. 2(h) shows the sequence of events for two 2-hop connection requests in the same path (i.e., parallel WDM establishment can be exploited) using the global strategy with the debouncing technique. As requests arrive, the CM groups those close enough according to the expected creation moment, generating groups. The CM sends then a single reconfigure operation to each agent queue. After waiting all the notifications of completion, the CM repeats the procedure, grouping the requests by expected duration before scheduling new single reconfigure operations for each agent. Note that debouncing reduces the number of interactions between the CM and NA compared to the example of Fig. 2(g).

[5] We recall that GMPLS traverses the nodes in the LSP direction for resource reservation, and then follows the LSP backwards for the actual configuration of the ROADMs. Therefore, the overhead of this two-phase commit is avoided thanks to the centralized view of the SDN controller that provides a globally synchronized state of the network enabling simultaneous signaling and actuation.

V. Experimental Results and Discussion

In this section, we first analyze several examples. Then, three different scenarios are used to analyze the average service provisioning time ($\overline{SPT}$) considering 3-hop connections that request available resources in the network. In particular, source, destination, creation time, duration, path and channel details are precomputed. Our objective with this choice is to evaluate the network-wide $\overline{SPT}$ only due to the WSS response time avoiding the interference of potential blocking situations. System parameters are listed in Tab. 1. These parameters are frequently used to describe the proposed strategies later in this study.

Tab. 1: Main parameters and variables in the experiments

Acronym	Definition
CD	Connection Duration
IAT	Inter Arrival Time
SPT	Service Provisioning Time

A. Examples

Fig. 3(a), (b) and (c) show the experimental results for 10 connections of three hops using different ROADMs with 3s of inter arrival time (IAT) and connection duration (CD) of 6s (first 5 connections shown). IATs and CDs are chosen to be in a range of seconds, which are comparable to the actuation time required by commercial WSSs and the SDN controller response time, as seen in Section II. In Fig. 3(a), the global strategy introduces notable creation/removal queue delays since a single creation/removal operation at a time is allowed in the network by the NA. Note that no creation/removal actuations overlap in time. On the other hand, queue times are remarkably reduced by the distributed strategies due to the parallel creation/removal operation of each connection by the RAs. Moreover, in Fig. 3(b) the parallel mode further reduces the actuation times parallelizing the three RA actions required at each connection compared to the sequential mode in Fig. 3(c).

Fig. 3(d), (e) and (f) show the experimental results for 5 simultaneous connection requests of three hops in the same path (i.e., WDM comb of 5 channels) with CD of 6s using the debouncing technique. In Fig. 3(d), the global strategy requires the highest time because, again, the single NA operating the network introduces a creation actuation time overhead. When combining the distributed-sequential mode with debouncing in Fig. 3(f), precisely three sequential operations are required at the three ROADMs of the three-hop path. Indeed, note that both creation and removal actions are around 12 seconds in accordance with the characterization results of Section III.C. Finally, in Fig. 3(e), the combination of distributed-parallel mode with debouncing guarantees the minimum SPT parallelizing both actions in the ROADM nodes for the 5 WDM channels.

Tab. 2: Benchmarking scenario avg. service provisioning time – in seconds for 80 connections of 3 hops (load - erlang).

		Without debouncing																	
		Global strategy						**Distributed strategy**											
								Parallel mode						**Sequential mode**					
Inter arrival time IAT(s)→		**1**	**2**	**3**	**4**	**6**	**8**	**1**	**2**	**3**	**4**	**6**	**8**	**1**	**2**	**3**	**4**	**6**	**8**
Conn. duration CD(s)	**3**	423.1 (39.4)	404.6 (35.4)	382.1 (31.8)	364.6 (28.5)	327.9 (23)	247.5 (17.7)	158.6 (32.6)	129.3 (24.8)	94.4 (17.8)	55.7 (10.6)	3.8 (1.3)	4.6 (0.9)	159.7 (28.6)	124.4 (23.7)	96.4 (18.3)	105.5 (15.2)	10.5 (2.7)	9.2 (1.3)
	6	419.7 (39.7)	402.9 (35.5)	381.6 (32.1)	356.5 (28.8)	306.1 (23.1)	247 (17.9)	160 (32.8)	128.2 (25.3)	94.5 (18.4)	55.2 (11.1)	5.7 (1.9)	3.8 (1.2)	158.9 (29.2)	124.1 (24.3)	123.9 (17.1)	66.8 (12.7)	11.6 (2.7)	12.3 (2.2)
	12	416.4 (40.3)	404.5 (35.7)	446.1 (33.7)	357.1 (29.2)	304.8 (24.5)	247.9 (18.3)	157.8 (33.5)	129.3 (26.2)	93.1 (19.2)	56.8 (12.3)	3.9 (2.5)	5.2 (2.1)	158.8 (30.4)	124.4 (25.3)	95.3 (19.5)	66.6 (13.6)	12.6 (3.6)	8.7 (2.6)
		With debouncing																	
Conn. duration CD(s)	**3**	31.9 (17.7)	28.3 (11.3)	27.1 (8.6)	27.3 (6.7)	26.2 (4.6)	26.2 (3.5)	11.7 (12.1)	11.3 (6.6)	11.2 (4.8)	11 (3.6)	9.1 (2.2)	10.5 (1.5)	23.5 (15.8)	21.3 (9.5)	20 (6.7)	20.4 (5.2)	19.4 (3.5)	17.7 (2.3)
	6	32.1 (17.7)	29 (11.3)	27.2 (8.6)	26.9 (6.7)	25.4 (4.6)	26.3 (3.5)	11.9 (12.1)	11.5 (6.9)	11 (4.8)	11.1 (3.7)	10.3 (2.6)	10 (1.8)	22.7 (16.1)	21.6 (10.1)	20 (7.4)	20.1 (5.8)	17.4 (3.7)	15.9 (2.9)
	12	32.8 (23.7)	29.6 (16.2)	27.7 (11.5)	27.3 (8.9)	26.5 (8.9)	26.2 (4.7)	12.4 (17.1)	11.3 (9.9)	11.4 (7.1)	11.5 (5.4)	10.5 (3.8)	9.9 (2.9)	24.1 (20.4)	22.5 (12.7)	20.7 (9.2)	20 (7)	17.5 (4.7)	16.8 (3.2)

B. Becnhmarking scenario

This scenario explores the impact of IAT and CD for 80 connections of three hops on the SPT. Again, IATs and CDs are comparable to the actuation time required by commercial WSSs and the SDN controller response time. SPT corresponds to the interval between the LSP expected creation moment at the control-plane and its actual establishment at the data-plane, which is equivalent to the sum of creation queue and actuation times. In this benchmarking scenario, the load in erlang is experimentally computed as $L(\text{erlang}) = N \cdot \bar{t}(s) / T$, where N is the total number of hops, $\bar{t}(s)$ is the average time of each connection (comprises CD, removal queue and actuation) and T is the total time of the experiment.

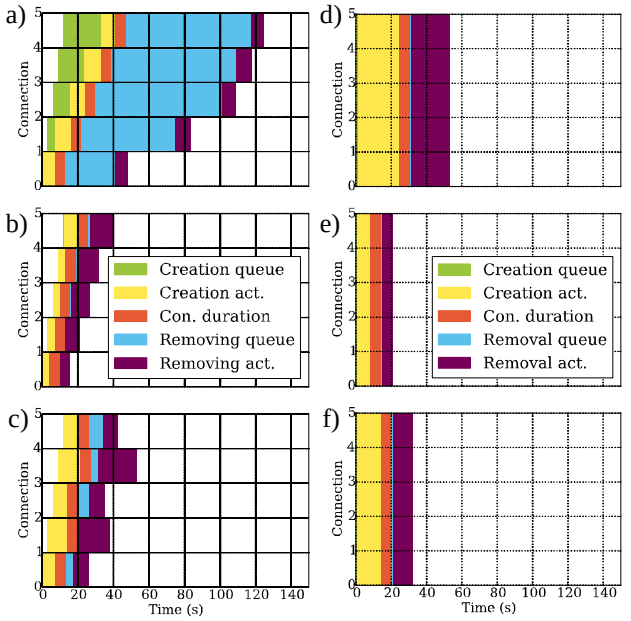

Fig. 3. Experimental results for 10 connection requests (first 5 shown) of three hops at different ROADM nodes using (a) global, (b) distributed-parallel, and (c) distributed-sequential modes, respectively. Experimental results for 5 simultaneous connection requests of three hops in the same path (i.e., WDM comb of 5 channels) using (d) global, (e) distributed-parallel, and (f) distributed-sequential modes and debouncing, respectively.

Tab. 2 lists the $\overline{SPT}$ (in brackets, load in erlang) of this benchmarking scenario for the proposed strategies and operation modes without (upper part) and with (lower part) debouncing, respectively. No impact is observed for the different CDs in the performance of all strategies and operations modes, regardless of the usage (or not) of the debouncing technique. However, on the one hand, when debouncing is not applied, IAT remarkably affects the $\overline{SPT}$. Indeed, low (high) IATs increase (decrease) the creation queue and therefore the average SPT. Moreover, the distributed strategy outperforms the global strategy thanks to the parallelization of the RA actions. In more detail, both distributed modes exhibit a similar performance for low IATs. However, for high IATs, the benefits of the parallel mode against the sequential mode are observed.

On the other hand, when debouncing is applied, the IAT dependence on the $\overline{SPT}$ is reduced for all the strategies and operation modes. In more detail, debouncing is able to reduce the queuing times due to an efficient control- and data-plane interaction even at low IATs. This leads to almost constant $\overline{SPT}$ of around 30 s in the global, 20 s in the distributed-sequential and 10 s in the distributed-parallel modes, respectively. Indeed, this is almost an order of magnitude reduction when compared to the $\overline{SPT}$ without debouncing for the global strategy and the distributed strategy at low IATs. However, higher $\overline{SPT}$ values are observed for high IATs for the distributed strategy. Indeed, the debouncing technique does not efficiently group data-plane actions for low-load cases using the distributed strategy, leading to an even higher performance reduction in the parallel mode compared to the sequential mode. This limitation can be overcome enabling a grouping factor in the debouncing technique to detect low-load scenarios and reduce (or even disable) the grouping time margin.

C. High load scenario

This scenario analyses IAT impact in high load conditions establishing connections with very high CD. Considering $T \approx N \cdot \text{IAT} + \bar{t}(s)$, and that CD dominates $\bar{t}(s)$ for very high duration values, it is possible to approximate $L \approx N$.

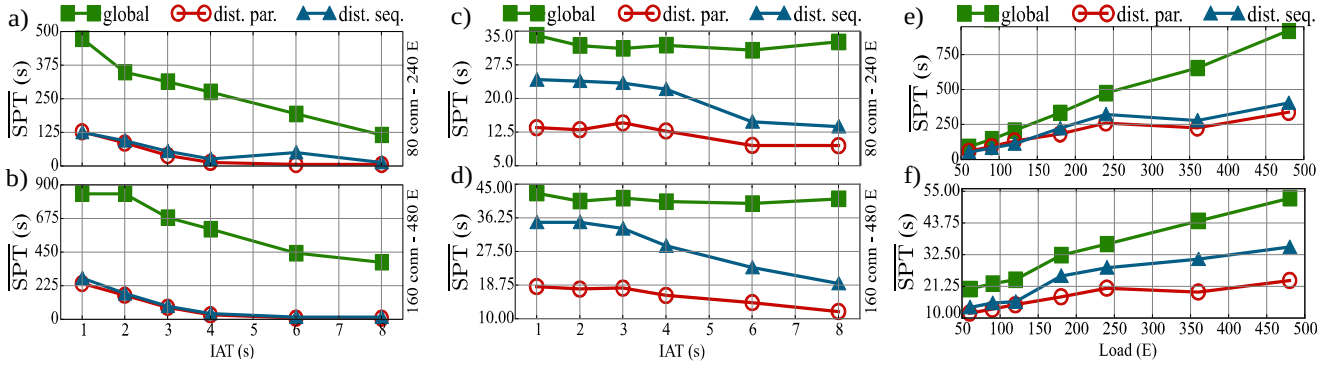

Fig. 4. High load scenario $\overline{SPT}$ for (a)(b) without and (c)(d) with debouncing for two load cases as a function of the IAT, resepectively. Fault recovery scenario $\overline{SPT}$ (e) without and (f) with debouncing as a function of the load.

On the one hand, Fig. 4(a) and (b) show the $\overline{SPT}$ as a function of the IAT without using the debouncing technique for 240 and 480 erlang, respectively. Again, the global strategy exhibits a poor performance against the two distributed strategies, which achieve similar $\overline{SPT}$. We observe a similar performance between the two modes of the distributed strategy in both load cases. However, small deviations, e.g. distributed sequential mode in Fig. 4 (a) shows an increase and then a decrease (IAT 4 to 8), may be generated due to the non-deterministic behavior intrinsic of the communication protocol to configure the equipment[6] and the typical process preemptive swap technique of the conventional operating systems used in the test-bed.

On the other hand, Fig. 4(c) and (d) show that debouncing drastically reduces the $\overline{SPT}$ for all the global strategy cases and for the distributed strategy low IAT cases. Between the two operating modes of the distributed strategy, the sequential mode further reduces the $\overline{SPT}$ as IAT increases compared to the parallel mode. Indeed, the sequential mode efficiently exploits the independent RAs jointly with the debouncing technique, whereas the parallel mode exhibits a small $\overline{SPT}$ reduction as IAT increases due to the few simultaneous hops per connection to be established. Thus, the sequential mode may be convenient in a high-load and high-IAT condition because it offers a reliable LSP establishment without presenting notable high $\overline{SPT}$ values.

D. Fault recovery scenario

This scenario considers a network fault recovery case where all connection requests arrive at the same time, thus IAT = 0. As in the high-load scenario 2, CD is considered to be high and $L \approx N$. Fig. 4(e) shows that when debouncing is not applied, the three modes exhibit a similar performance at low load. In particular, these low-load cases show a clear consistency with the benchmarking scenario for IAT=1, where the results are reported numerically in Tab. 2. However, for L around 200 E and higher, the distributed strategies offer significantly lower SPT compared to the global strategy. Nevertheless, when debouncing is applied in Fig. 4(f), all the $\overline{SPT}$ values are reduced one order of magnitude and a clear differentiation is observed among the different operation modes. However, even considering that the sequential mode increases the $\overline{SPT}$ in 10 s for the high-load cases compared to the parallel mode, it is yet a valuable choice for reliable LSP establishment as it may avoid optical power fluctuations [29] thanks to its inherently ordered sequence of configurations. Finally, it is also worthwhile to mention that the comparison of the strategies and operations in this scenario (with simultaneous arrivals) benefits the debouncing approach. Therefore, even if the results of the techniques with or without debouncing would not be directly comparable under this assumption, the experienced $\overline{SPT}$ showed similar trends as in the other scenarios.

E. Experimental application final remarks

In the three scenarios analyzed, the distributed-parallel mode with debouncing offers the best performance in terms of $\overline{SPT}$. In particular, this design for the SDN traffic loader application presents SPTs in the order of ten seconds (for three-hop connections) clearly outperforming the other alternatives. Although the global strategy exhibits the worst performance, it offers advantages in terms of simplicity in its implementation because it directly maps the decisions of the CM with a single agent (NA) into network configuration actions without the need to synchronize multiple agents. Besides, the inclusion of a global strategy in the discussion can be considered as information for comparison purposes, while not necessarily being a choice for deployment. On the other hand, the configuration of all the nodes in the lightpath at the same time by the distributed-parallel mode may require a feedback mechanism between nodes to slightly change the signal power across the path for final adjustments [28], [29]. A study that combines the proposed dynamic traffic allocation strategies together with signal power control strategies is left for future work.

Tab. 3: Main parameters and variables in the simulations

Acronym	Definition
W	Number of wavelengths to be configured by a WSS
$X_{WSS}(W)$	WSS response time to configure a batch of size W
W_{max}	Maximum batch size
λ	Conn. req. generation rate. $\lambda = 1 / IAT$

[6] The SDN application uses a REST interface over HTTP provided by the custom equipment. Therefore, it is subject to eventual network activity, HTTP overhead, and the "best-effort" policy of the TCP/IP stack.

VI. NETWORK MODEL

This section further analyzes the Service Provisioning Time (SPT) required to create and release LSPs. Note that the SPT is intrinsically related to the response time of the WSS devices to switch or block the required lightpath in the physical layer. As shown in Fig. 1(b), the time required to perform a change of attenuation in a WSS device is a function of the number of wavelengths for which the attenuation is being adjusted. Parameters and variables that are used to describe the algorithm and simulation assumptions are reported in Tab. 3.

From a network viewpoint, any LSP request – either setup or tear down - is handled by the SDN controller as follows. In the SDN controller, the Connection Manager (CM) finds a path, assigns a wavelength to it (RWA) and computes the appropriate signal power level for each link in the path in case this information in not precomputed out of the controller. Then, it issues a control command to the ROADM agent (RA) of every ROADM along the path. These commands contain the attenuation of the assigned wavelength, based on the signal power level computed by the CM. These signal power levels can be chosen to optimize the optical signal-to-noise ratio (OSNR) [34] for setup requests or block the assigned wavelength for tear down requests. Upon completion of all these setup (tear down) commands the lightpath is established (released). In this study, each optical node has a single RA which can only control one WSS device at a time. Even if the RA should be capable to configure several WSS (in the same node) at the same time, a limitation in the implementation of the RA used in our test-bed leads to this characteristic that we decided to maintain in this simulation analysis. If multiple control commands are sent to the same node in a short period of time, some of them may have to be queued, awaiting their service times. The policy of this queue is FIFO and it can be either implemented in the CM of SDN controller (as in Sec. IV) or at the optical node. For consistency, in this study we implemented the queue in the CM of SDN controller.

Relevant to the control-data plane interaction, special attention must be given to control commands that would require simultaneous adjustment of the attenuation of multiple wavelengths routed through the same WSS device. In this case, the CM has the option to issue a single control command to adjust all of them at once. Given the different types of requests to be included in the commands issued by the CM (e.g., single or multiple, setup or tear down) two policies can be defined:

1. Distinct Batch Service (DBS): control commands contain multiple requests of the same type (setup or tear down) for the same WSS device can be combined and executed simultaneously;
2. Single Batch Service (SBS): control commands contain multiple requests of both types (setup and tear down) for the same WSS device can be combined and executed simultaneously.

The DBS strategy is motivated by certain WSS models not supporting the joint increase and decrease of the attenuation in different parts of the spectrum. Moreover, non-adequate firmware implementations may lead to situations where a WSS device capable to perform an SBS operation is controlled by a legacy firmware implementation that does not allow this feature.

Since every arriving connection requests must be processed by the CM at the SDN controller, an awareness mechanism needs to keep track of the WSS state changes required in the optical network by each request. Therefore, the CM can use either an *apriori* scheduling mechanism or a real time rescheduling mechanism to assign the sufficient configuration time for establishing/releasing the LSP. In the latter, the controller keeps track of the number of wavelengths (W) that are being adjusted simultaneously for each WSS device, and updates the configuration time using the updated W ($X_{WSS(W)}$). With this solution, the CM must re-compute the exact time for each WSS device to be reconfigured at each new connection request. To avoid this CPU demanding task, the *a priori* scheduling mechanism computes the WSS time-to-reconfigure only once, by assuming that the WSS required reconfiguration time will not exceed a given predefined threshold $X_{WSS(W_{max})}$. As we will see later in this section, making use of a predefined threshold has the drawback of having to guess, ahead of time, what the WSS reconfiguration time is going to be and therefore presents an optimization problem to solve. In this study, we investigated the *a priori* scheduling mechanism taking three time constraints into account.

1. Given that the RA can only control one WSS at a time, the CM cannot overlap the configuration times of different WSSs belonging to the same node.
2. The CM needs to schedule the proper duration of the configuration time consistently with Fig. 1(b).
3. If the CM is set to use the distributed-sequential mode, the LSP's WSS devices must be serviced sequentially, starting from the source node and continuing down the path direction. If it is set to use the distributed-parallel mode, WSS devices of the different LSP's nodes can be configured simultaneously.

In order to comply with the second time constraint, the controller must take into account the number of wavelengths (W) that will be configured at once. Given that in the cases of DBS and SBS policies, the attenuation of multiple wavelengths must be adjusted simultaneously with a single command. Since the CM schedules configuration times ahead-of-time, it cannot predict the exact value of W nor can accurately determine the duration of the WSS configuration time. To circumvent this drawback, we propose to use a threshold W_{max}. This threshold is used for two purposes: first, it enables the CM to estimate the WSS configuration time by assuming that at most W_{max} wavelengths will be reconfigured, and second, it ensures that the controller will not allow more than W_{max} requests to be bundled together in the same command. Any additional request will be bundled in a distinct command for a subsequent WSS reconfiguration cycle. The right value for W_{max} must be chosen keeping in mind that an excessively large value will unnecessarily increase the estimated WSS configuration time, while an excessively small value will not allow to take full advantage of concurrently reconfiguring multiple wavelengths.

Algorithm 1: Scheduling Algorithm for Traffic Allocation

```
1: procedure SCHEDULE(W_max, SPT)
2:     SPT ← 0
3:     for all i ∈ request_nodes[...] do
4:         nEvent ← createnewevent
5:         nEvent_wc ← 1
6:         nEvent_type ← request_type
7:         nEvent_link ← request_links[i]
8:         nEvent_st ← request_time
9:         nEvent_et ← nEvent_st + X_WSS(W_max)
10:        if request_node[i] = null then
11:            go to line 15
12:        while Event ≠ null do
13:            if nEvent_st < Event_st then
14:                if nEvent_et ⩽ Event_st then
15:                    put nEvent here
16:                    exit while loop
17:                else if Event_wc < W_max then
18:                    if nEvent_link = Event_link then
19:                        if policy = DBS then
20:                            if nEvent_type = Event_type then
21:                                go to line 23
22:                        else if policy = SBS then
23:                            merge nEvent with Event
24:                            Event_wc ← Event_wc + 1
25:            if Event_st ⩽ nEvent_st ⩽ Event_et then
26:                nEvent_st ← Event_et
27:                nEvent_et ← nEvent_st + X_WSS(W_max)
28:        SPT ← nEvent_et
29:    return SPT
```

The effect of W_{max} on the average SPT can be studied by varying its value from 1 to the maximum number of wavelengths (fiber capacity) at different traffic loads. Then, an optimal W for each load can be numerically computed, which minimizes the average SPT. A simulation tool is used to compute these optimal W_{max}. The scheduling algorithm that is implemented to estimate the LSP's WSS configuration times is shown in Algorithm 1.

Algorithm 1 details the Scheduling Algorithm for Traffic Allocation and includes the following variables. Each arriving request contains a list of all nodes ($request_{nodes}$) and a list of all link IDs ($request_{link}$) along the computed path, the request type ($request_{type}$) and the request time ($request_{time}$). $request_{type}$ can be either setup or tear down and $request_{time}$ is the arrival time of the LSP requests. As already mentioned, for each LSP request one or more control commands are issued, one for each hop in the path. Let these commands to be referred to as *events*, which contain their type ($event_{type}$) being setup or tear down, link ID ($event_{link}$), start time ($event_{st}$), end time ($event_{et}$) and a wavelength counter ($event_{wc}$). The link ID is the index of the link (output fiber) to which the event is assigned, start time is the time that the control command is issued, end time is a conservative estimate for the WSS device to complete reconfiguration, and wavelength counter keeps track of how many wavelengths are being bundled together in the control command.

VII. Simulation Results

In this section, we first define the simulation assumptions. Then, we discuss the numerical results obtained through simulation.

A customized discrete-event simulator – with Algorithm 1 applied - is implemented to obtain the simulation results. A mesh network with 5 nodes and 8 fiber-pair is used (Fig. **2**(a)). Each fiber capacity is 80 wavelengths. In the discrete-event simulator, connection requests are generated by a Poisson process with a rate of λ = 1 / *IAT*. The source and destination nodes for each request are randomly chosen using a uniform distribution. Five shortest paths are computed using hop counts as the metric for each generated request. The routing algorithm checks the wavelength availability along the five computed paths and the shortest path with available wavelength is selected for that lightpath request. If there are two or more candidate paths of the same hop-count, the least loaded path is chosen, and the available wavelength with the lowest ID is reserved along the path. The routing algorithm executes the same procedure even in case the maximum number of computed paths for a request is less than five. Wavelength continuity is enforced along the lightpath and the request is blocked when none of the five paths has at least one common wavelength available. The connection duration (CD) is a random variable with an exponential probability distribution of average $\overline{CD}$. IAT is varied to obtain results at different offered loads. Setup and tear down configuration time for the WSSs is as in Fig. 1(b). Simulation is set to use the distributed-sequential and distributed-parallel modes, and to compute the average SPTs at each offered load, while W_{max} is varied from 1 to 80. Note that W_{max} limits the number of wavelengths that can be bundled to be configured at once in a WSS device. In the experimental results reported in section III.C, it is shown that WSS response times for both blocking and unblocking operations are the same. Moreover, note that the lightpath setup will require a given "number of hops" of unblocking operations which coincide with the number blocking operations required in its tear-down process. Therefore, it can be concluded that both the average SPT or the average lightpath tear down time are the same. Consequently, we decided to report only the average SPT.

The average SPT is shown in Fig. 5 and Fig. 6 for DBS and SBS policies, respectively. In each figure, the value of W_{max} is varied and the corresponding average SPT is reported, each curve represents a load (*IAT* = [0.2, 10]) with $\overline{CD}$ = 100 (sec), thus being the bottom curve the offered load with *IAT* of 0.2 and the top curve with *IAT* of 10.

Lines with circle marker in Fig. 5 and Fig. 6 show the minimum $\overline{SPT}$ against the optimal W_{max}. It can be seen that the optimal W_{max} increases with the offered load. The reason is that when the network resources are more utilized, more control commands can be issued for the same WSS device and executed at once. This effect promotes higher values for W_{max} at higher loads as they efficiently group together more operations per command.

Fig. 7(a) and (b) show the optimal W_{max} as a function of *IAT* for DBS and SBS policies, respectively. The lines from

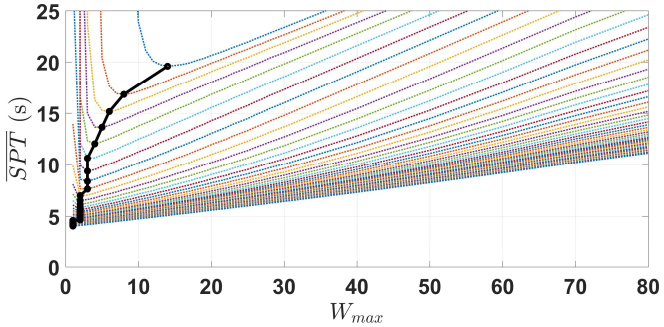

Fig. 5: $\overline{SPT}$ vs. W_{max} for DBS policy and the average CD of 100 (s).

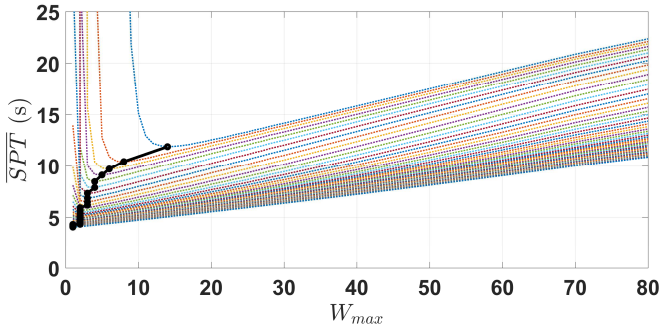

Fig. 6: $\overline{SPT}$ vs. W_{max} for SBS policy and the average CD of 100 (s).

bottom to top report the result when $\overline{CD}$ is set to be 200, 150, 100, 50, 20 and 10 (sec). Shorter CDs can free up resources (wavelengths) faster while the WSS configuration times remain unchanged. Because of this, there are more wavelengths available and the network can accept more LSP setup requests in any given time period. Therefore, it can be concluded that increasing W_{max} is the proper solution for handling higher arrival rates (i.e., low *IAT*s).

Fig. 8(a) and (b) show the minimum $\overline{SPT}$ as a function of *IAT* using the optimal W_{max} for DBS and SBS policies, respectively. The lines from bottom to top show the minimum $\overline{SPT}$ when $\overline{CD}$ is set to be 200, 150, 100, 50, 20 and 10 (sec). Since in the SBS policy requests of any type can be combined and configured simultaneously, we observe a notable performance gain of the SBS policy against the DBS policy.

Note that this simulation analysis explores high-dynamic scenarios with random source-destination pairs and executing the RWA in the CM. Therefore, it is possible to analyze the effect of the WSS response time from a network-wide viewpoint jointly with the blocking response. For $\overline{CD}$ times close to the SPT under these considerations,

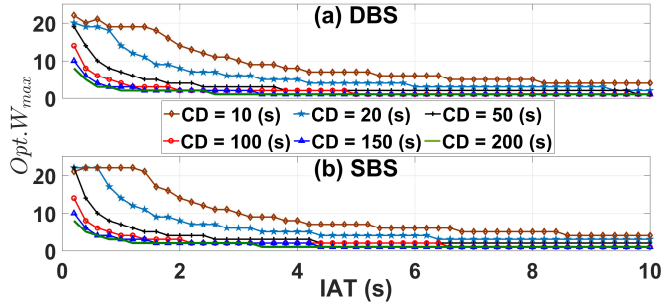

Fig. 7: *Wmax* vs. *IAT* (a) DBS and (b) SBS policies.

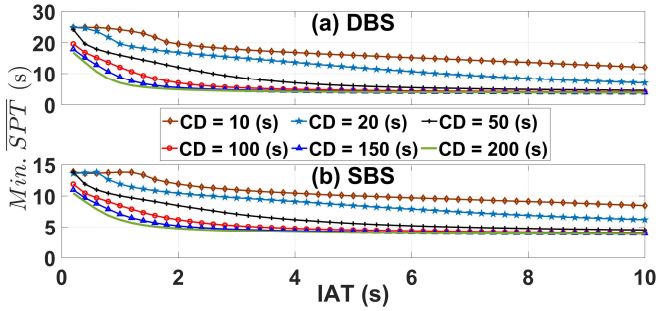

Fig. 8: Minimum $\overline{SPT}$ vs. *IAT* (a) DBS and (b) SBS policies.

the network performance becomes unstable being the blocking the major effect observed. Consequently, unstable behavior of both optimal W_{max} and minimum $\overline{SPT}$ is observed in Figs. 7 and 8 for CD values of 20 and 10 at high loads**.** Note that the characterization of the network behavior for low CD values without blocking issues is reported in the experimental part (Section V). However, given that usual values of CD will be in the order of minutes, the SDN controller can make use of these analyses by employing the optimal W_{max} in order to minimize the average SPT at each specific offered load.

Finally, the optimal W_{max} for the distributed-parallel mode is evaluated using the same procedure as distributed-sequential mode. Fig. 9(a) and (b) compare the minimum average SPT for the distributed-sequential (solid line) and parallel (dotted line) modes as a function of *IAT*s for DBS and SBS policies, respectively. The distributed-parallel mode yields better performance with both policies. However, given that both strategies follow a similar trend, we decided to present the average SPT for distributed-sequential mode only, as it provides a more reliable LSP establishment, as already mentioned in Section IV.

VIII. Summary and Future Work

This paper addressed LSP establishment response time in a metropolitan optical network test-bed in the presence of highly dynamic traffic scenarios. First, data-plane implementation and WSS device response time were reported. Then, three SDN-enabled strategies were reviewed and their experimental results discussed. The distributed strategy that uses multiple ROADM Agents in the SDN controller achieves data-plane configuration in two modes: a fast parallel mode suitable for fast failure recovery and a reliable sequential mode for usual EON dynamic traffic allocation. Moreover, a debouncing technique was

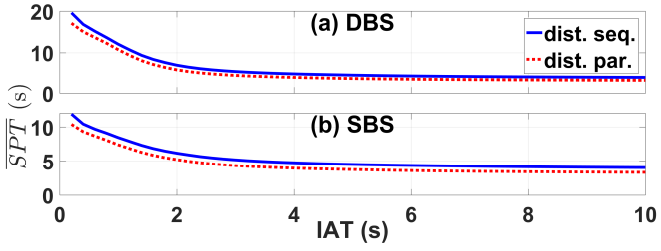

Fig. 9: Minimum $\overline{SPT}$ for distributed-sequential and distributed-parallel mode as a function of IAT for (a) DBS and (b) SBS policies and the average CD of 100 (s).

applied to further reduce the average SPT, especially in high-load cases. Subsequently, two batch service policies were considered aiming to reduce the average SPT. A simulation platform was used to estimate the average SPTs experienced when using the two policies. It was shown that there is an optimal number of wavelengths per batch that minimizes the average SPT. This number is a function of the network load.

Two specific aspects introduced in this paper can be explored as future work. On the one hand, different debouncing grouping times could be analyzed to investigate the trade-off between the number of LSPs installed simultaneously and the grouping time to wait for them. On the other hand, signal power control strategies [34] could be efficiently combined with the SDN distributed operation modes discussed in this paper.

ACKNOWLEDGMENT

Portions of this work were presented at ONDM 2016 [13]. This work was supported in part by NSF Grants No. CNS-1111329, CNS-1405405, CNS-1409849, and partially funded by the aid granted by the Spanish Ministry of Science and Innovation under the project SUNSET (FEDER-TEC 2014-59583-C2-2-R). The authors also thank the Brazilian Ministry of Communications, FUNTTEL/FINEP for the financial support under the project 100 GETH, and CNPq (grant 312047/2015-0).

REFERENCES

[1] J. Fernandez-Palacios, V. López, B. Cruz, and O. González de Dios, "Elastic Optical Networking: An Operators Perspective," Mo.4.2.1, *European Conf. on Optical Communication (ECOC), Cannes, France*, Sep. 2014.

[2] Finisar's ROADMs and wavelength selective switches (WSSs) https://www.finisar.com/roadms-wavelength-management

[3] 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 clouds architecture," in *IEEE Communications Magazine*, vol. 53, no. 4, pp. 159-165, April 2015.

[4] X. Jiny, Y. Li, D. Wei, S. Li, J. Gao, L. Xu, G. Li, W. Xu, and J. Rexford. "Optimizing Bulk Transfers with Software-Defined Optical WAN." *Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference*. ACM, 2016.

[5] Calient. Available at http://www.calient.net/

[6] Polatis. Available at http://www.polatis.com/

[7] A. Singla, A. Singh, K. Ramachandran, L. Xu, Y. Zhang, "Proteus: a topology malleable data center network." Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks. ACM, 2010.

[8] K. Chen, A. Singla, A. Singh, K. Ramachandran, L. Xu, Y. Zhang, X. Wen, and Y. Chen, "OSA: An optical switching architecture for data center networks with unprecedented flexibility," *USENIX NSDI*, 2012.

[9] Y. Fei, Y. Jayabal, Z. Lu, M. Razo, M. Tacca, A. Fumagalli, R. Hui, G. Galimberti, and G. Martinelli, "Handling race conditions among bidirectional LSP requests via WA-method-TLV in GMPLS WDM networks," *Networks and Optical Communications (NOC)*, 2014.

[10] M. Dallaglio, A. Giorgetti, N. Sambo, L. Velasco, and P. Castoldi, "Impact of Multi-wavelength sliceable transponders in Elastic Optical Networks," *Optical Fiber Communication Conference (OFC)*, 2015.

[11] S. Gringeri, N. Bitar, and T. J. Xia. "Extending software defined network principles to include optical transport," *IEEE Communications Magazine*, v. 51(3), pp. 32-40, 2013.

[12] D. Kreutz, F.M.V. Ramos, P.E. Verissimo, C.E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," *Proceedings of IEEE*, v.103, Jan. 2015.

[13] M. Garrich, A. Bravalheri, M. Magalhães, M. Svolenski, X. Wang, Y. Fei, A. Fumagalli, D. Careglio, J. Sole-Pareta, J. Oliveira, "Demonstration of dynamic traffic allocation in an SDN-enabled metropolitan optical network test-bed," *Optical Network Design and Modeling (ONDM),* Cartagena, Spain, May 2016.

[14] W. Lu, Z. Zhu, and B. Mukherjee, "On Hybrid IR and AR Service Provisioning in Elastic Optical Networks," *Journal of Lightwave Technology (JLT)*, vol.33, no.22, pp.4659-4670, Nov. 2015.

[15] A. Giorgetti, F. Paolucci, F. Cugini, P. Castoldi, "Dynamic Restoration With GMPLS and SDN Control Plane in Elastic Optical Networks [Invited]," *Journal of Optical Communications and Networking*, vol. 7, pp. A174-A182, 2015.

[16] X. J. Zhang, M. Birk, A. Chiu, R. Doverspike, M. D. Feuer, P. Magill, E. Mavrogiorgis, J. Pastor, S. L. Woodward, J. Yates, "Bridge-and-roll demonstration in griphon (globally reconfigurable intelligent photonic network)," *Optical Fiber Communication Conference (OFC)*, 2010.

[17] A. Mahimkar, A. Chiu, R. Doverspike, M. D. Feuer, P. Magill, E. Mavrogiorgis, J. Pastor, S. L. Woodward, J. Yates, "Bandwidth on demand for inter-data center communication," *ACM Workshop on Hot Topics in Networks*, 2011.

[18] S. J. B. Yoo, Y. Yin and K. Wen, "Intra and inter datacenter networking: The role of optical packet switching and flexible bandwidth optical networking," *Optical Network Design and Modeling (ONDM),* Colchester, 2012.

[19] N. Farrington, G. Porter, S. Radhakrishnan, H. H. Bazzaz, V. Subramanya, Y. Fainman, G. Papen, and A. Vahdat, "Helios: A hybrid electrical/optical switch architecture for modular data centers," *ACM SIGCOMM*, 2010.

[20] G. Wang, D. G. Andersen, M. Kaminsky, K. Papagiannaki, T. Ng, M. Kozuch, and M. Ryan, "c-Through: Part-time optics in data centers," *ACM SIGCOMM*, 2010.

[21] Y. Chen, S. Jain, V. K. Adhikari, Z.-L. Zhang, and K. Xu, "A First Look at Inter-Data Center Traffic Characteristics via Yahoo! Datasets," in INFOCOM, 2011.

[22] A. Roy, H. Zeng, J. Bagga, G. Porter, A.C. Snoeren, "Inside the social network's (datacenter) network," *ACM SIGCOMM,* New York, USA, pp. 123–137, 2015.

[23] M. Garrich, A. Bravalheri, M. Magalhães, H. Carvalho, J. Assine, H. Rusa, H. Yamamura, F. Hooft, U. Moura, J. Januário, M. Nascimento, L. Mariote, and J. Oliveira, "Pioneering hardware modeling and software design for optical infrastructure in the Autonomous Network project," *Optical Network Design and Modeling (ONDM),* Cartagena, Spain, May 2016.

[24] V.E.S. Parahyba, J.D. Reis, S.M. Ranzini, E.O. Schneider, E.S. Rosa, F.D. Simões, J.C.M. Diniz, L.H.H. Carvalho, E.P.L. Filho, J.C.R.F. Oliveira and J.R.F. Oliveira, "Performance against implementation of digital backpropagation for high-speed coherent optical systems." Electronics Letters, vol 51.14, pp. 1094-1096, 2015.

[25] J.C.M. Diniz, J. C. R. F. de Oliveira, E. S. Rosa, V. B. Ribeiro, V. E. S. Parahyba, R. da Silva, E. P. da Silva, L. H. H. de Carvalho, A. F. Herbster, and A. C. Bordonalli, "Simple feed-forward wide-range frequency offset estimator for optical coherent receivers." Optics express, vol. 19.26, pp. B323-B328, 2011.

[26] J. C. M. Diniz, S. M. Ranzini, V. B. Ribeiro, E. C. Magalhaes, E. S. Rosa, V. E. S. Parahyba, L. V. Franz, E. E. Ferreira and J. C. R. F. Oliveira, "Hardware-efficient chromatic dispersion estimator based on parallel Gardner timing error detector." Optical Fiber Communication Conference (OFC), 2013.

[27] E. Magalhães, M. Garrich, H. Carvalho, M. Magalhães, N. González, J. Oliveira, A. Bordonalli, and J. Oliveira "Global WSS-based equalization strategies for SDN metropolitan mesh optical networks," *European Conference on Optical Communication (ECOC)*, Cannes, France, 2014.

[28] H. Carvalho, E. Magalhães, M. Garrich, N. Gonzalez, M. Nascimento, F. Margarido, L. Mariote, A. Bordonalli, and J. Oliveira, "SDN Dual-optimization Application for EDFAs and WSS-based ROADMs," *Optical Fiber Communication Conference (OFC)*, Los Angeles, USA, 2015.

[29] J. Januário, M. Garrich, H. Carvalho, and J. Oliveira, "Experimental demonstration of SDN-enabled centralized strategies for cascaded WSS-based ROADMs", *European Conference on Optical Communication (ECOC)*, Valencia, Spain, Sep. 2015.

[30] M. Bjorklund, "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)," RFC 6020, Oct. 2010, Available from [accessed Jan 2016]: http://www.rfc-editor.org/info/rfc6020

[31] F. Belqasmi, R. Glitho, C. Fu, "RESTful web services for service provisioning in next-generation networks: a survey," *IEEE Communications Magazine*, v.49, n.12, pp.66-73, December 2011.

[32] Y. Sakurai, M. Kawasugi, Y. Hotta, S. Khan, H. Oguri, K. Takeuchi, S. Michihata, and N. Uehara, "LCOS-based 4x4 wavelength cross-connect switch for flexible channel management in ROADMs," *Optical Fiber Communication Conference (OFC)*, 2011.

[33] JDSU white paper "A Performance Comparison of WSS Switch Engine Technologies", May 2009, Available from [accessed Feb 2016]: http://nse.viavisolutions.com/ProductLiterature/wsscomp_wp_cms_ae.pdf

[34] X. Wang, Y. Fei, M. Razo, A. Fumagalli, M. Garrich, A. Andrade, M. Svolenski, H. Carvalho, "Effects of signal power control strategies and wavelength assignment algorithms on circuit OSNR in WDM networks," *Photonic Network Communications*, pp. 1-14, 2016.