

# First Experimental Demonstration of ABNO-driven In-Operation Flexgrid Network Re-Optimization

L. Velasco<sup>1\*</sup>, F. Paolucci<sup>2</sup>, Ll. Gifre<sup>1</sup>, A. Aguado<sup>3</sup>, F. Cugini<sup>2</sup>, P. Castoldi<sup>2</sup>, and V. López<sup>3</sup>

<sup>1</sup> Optical Communications Group (GCO), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain.

<sup>2</sup> Consorzio Nazionale Interuniversitario per le Telecomunicazioni (CNIT), Pisa, Italy.

<sup>3</sup> Telefónica Investigación y Desarrollo (TID), Madrid, Spain.

\*e-mail: lvelasco@ac.upc.edu

**Abstract:** The after failure repair optimization problem is used to experimentally validate in-operation planning using the ABNO architecture to control a flexgrid optical network. A back-end PCE is used to solve this computationally-intensive planning problem.

© 2014 Optical Society of America

**OCIS codes:** (060.4251) Networks, assignment and routing algorithms; (060.4256) Networks, network optimization

## 1. Introduction

In-operation network planning consists in making network resources available by reconfiguring and/or re-optimizing the network on demand and in real-time [1]. Although the concept of in-operation planning is general and can be applied to any network, we focus on flexgrid-based optical core networks [2]. The flexgrid technology split the optical spectrum into multiple equally-sized frequency *slices*, where a variable number of consecutive slices (*frequency slot*) can be allocated to each connection depending on the requested bandwidth and the used modulation format.

As a use case of network re-optimization, in this paper we face the after failure repair optimization (AFRO) problem, defined in [3]. When a fiber cut occurs, connections affected are restored; when the link is repaired not only those connections that were restored, but also any other connection that might use the repaired link, can be re-routed. In addition, to increase restorability, multipath restoration can be applied [4] dividing original connections into several parallel disaggregated connections (*sub-connections*). Although multipath (MP) restoration has benefits, it provides poor resource utilization and it is spectrally inefficient. Hence, re-optimizing the network just after a link is repaired, becomes essential. To that end, the AFRO problem needs to be enhanced with the capability of merging sub-connections; we call it as MP-AFRO.

To implement in-operation planning, a control plane architecture automating connection provisioning is needed. In that regard, the IETF is currently defining the Application-Based Network Operations (ABNO) architecture [5] based on functional elements, such as the active stateful Path Computation Element (PCE) [6]. As for the PCE, we assume a front-end/back-end (fPCE/bPCE) architecture; the bPCE is a planning tool prepared to solve optimization problems [7], such as MP-AFRO, whereas the fPCE delegates some complex computations to the back-end.

## 2. Multi-Path Restoration and After Failure Repair Optimization

For illustrative purposes, Fig. 1 reproduces a small flexgrid network topology where several connections are currently established; in particular, the route of connections *P1* and *P2* is shown. The spectrum usage is also provided in Fig. 1, where the spectrum allocation for all five established connections is specified. Three snapshots with the state of the network are shown: the link 6-7 has failed in Fig. 1a; multipath restoration in Fig. 1b and connection *P2* has been split into two parallel sub-connections *P2a* and *P2b*; failed link 6-7 has been repaired in Fig. 1c, and re-optimization has been performed by solving MP-AFRO.

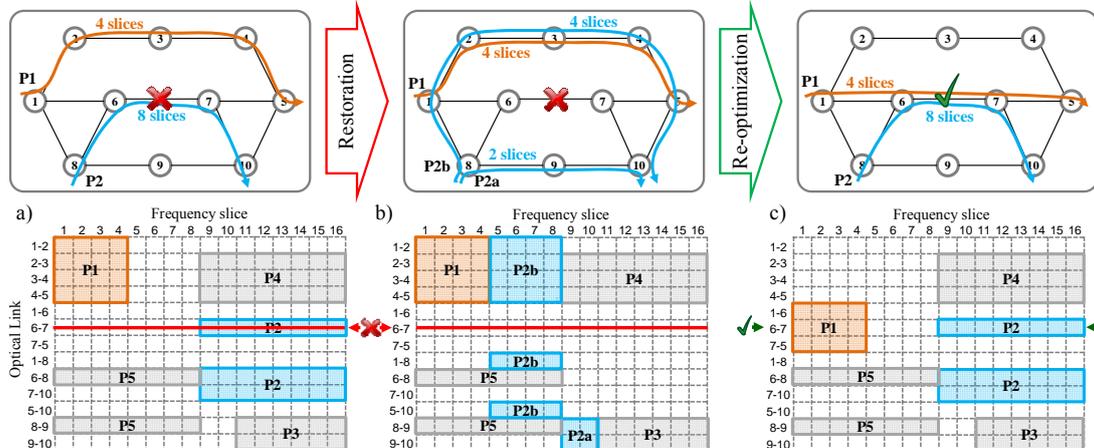


Fig. 1. An example of multi-path restoration and after failure repair optimization. 6.25 GHz frequency slices are used.

It is clear, in view of the example, that multipath restoration allows increasing restorability, in particular when no enough contiguous spectrum can be found along a single path, as happened when restoring  $P2$ . Nonetheless, this benefit is at the cost of an increased resource usage, not only as a result of using (not shortest) parallel routes, but also because the spectral efficiency is degraded when connections are split. For instance, a 400 Gb/s aggregated flow can be conveyed on one single 100 GHz connection or on four parallel 37.5 GHz sub-connections (150 GHz in total). For this very reason, resource utilization can be improved by applying MF-AFRO, i.e. by re-routing established connections on shorter routes and by merging parallel sub-connections to achieve better spectrum efficiency. Fig. 1c illustrates an example of such re-optimization, where connection  $P1$  has been rerouted using a shorter route that includes the repaired link, whilst sub-connections  $P2a$  and  $P2b$  have been merged on a single connection conveying the originally requested bandwidth.

The MP-AFRO problem can be formally stated as follows:

- **Given:** a) a network topology  $G(N, E)$  defined as a set of optical nodes  $N$  and a set of fiber links  $E$ ; b) an optical spectrum divided into frequency slices of a given width; c) the set of slices used by non-selected connections; d) a set of connections and sub-connections candidate for re-optimization,  $CL$ .
- **Find:** the route and spectrum allocation for connections in  $CL$ , merging sub-connections serving each original connection.
- **Objective:** maximize the bitrate served and minimize the total number of used frequency slices.

In the next section, we focus on devising the workflow that need to be performed in the network control plane so as to be able to request solving the MP-AFRO problem and deploy the obtained solution.

### 3. Proposed re-optimization workflow

To deal with network re-optimization, we consider a control plane based on the ABNO architecture, which includes a fPCE responsible of computing provisioning requests and dealing with network data plane, and a bPCE capable of performing complex computations to solve optimization problems.

We assume that an operator in the Network Management System (NMS) triggers the MP-AFRO workflow after a link has been repaired. To that end, the NMS issues a service request towards the ABNO Controller. The diagram in Fig. 2 reproduces the re-optimization sequence involving ABNO components. When the request from the NMS arrives at the ABNO controller, re-optimization is requested by sending a PCReq message (labeled as 1 in Fig. 2) to the fPCE. Upon receiving the request, the fPCE collects relevant data to be sent to the bPCE in the form of a PCReq message containing a Global Concurrent Optimization (GCO) request (2).

In light of the MP-AFRO problem statement, and assuming that the network topology and the current state of the resources has been distributed by other means, the information to be included in the GCO request is the set of connections candidate for re-optimization,  $CL$ . Therefore, an algorithm to find the candidate connections in the Label Switched Paths (LSP) database is needed. Our algorithm selects the connections that have been previously split or whose shortest path traverses the repaired link. Information regarding candidate connections is sent in a PCReq message containing the GCO request. Each candidate connection is sent as an individual request, identified by a RP object; the end points and original bandwidth is also included. In addition, for each individual sub-connection related to the original connection, its current route and spectrum allocation is specified using a RRO object (the Label sub-object specifies spectrum allocation with the tuple  $\{n, m\}$ ). Aiming at finding an optimal solution for the entire problem, individual requests are grouped together using a SVEC object. The desired network-wide GCO related criterion, such as “MP-AFRO”, is specified by means of the OF object. Finally, it is important to specify that the repaired link should/must be used for re-optimization. To that end, we extended current standards adding an IRO object.

Upon receiving the PCReq, the bPCE runs the specified algorithm. To solve the MP-AFRO problem, we implemented the heuristic algorithm in Table 1, which provides a good trade-off between complexity and optimality. The algorithm maximizes the amount of bitrate that is served (note that only part of the original bitrate could be restored), while minimizing the amount of used resources. Resources related to the route and spectrum allocations computed for each connection in the  $CL$  set, are temporally reserved in a copy of the local traffic engineering database (TED), so subsequent computations consider those resources in use. *MaxIter* iterations on the randomly sorted  $CL$  list are done and the best solution is eventually returned.

The solution of the MP-AFRO problem is encoded in a PCRep (3); each individual request is replied specifying the bitrate that could be served and the route and spectrum allocation of the connections related to each request in a list of ERO objects. Note that the solution might entail merging several existing sub-connections to create one or more new connections. The Order TLV is included in RP objects to indicate the order in which the solution needs to be implemented in the data plane.

Upon receiving the PCRep message with the solution from the bPCE, the fPCE compares the contents of the ERO objects received with the connections currently in its LSP-DB. In case updates are received, the fPCE sends the corresponding PCUpd/PCInit messages towards source GMPLS controllers (4) in the order specified. When the solution has been completely implemented i.e., when the corresponding PCRpt messages are received, the fPCE replies the completion of the requested operation to the ABNO controller using a PCRep message (5), which eventually informs the NMS.

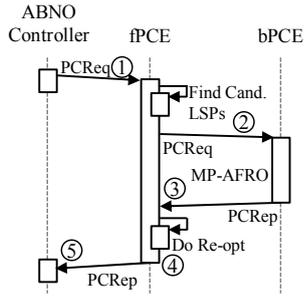


Fig. 2. Re-optimization sequence

Table 1. MP-AFRO Heuristic	
<b>INPUT</b>	$TED, CL, maxIter$
<b>OUTPUT</b>	$bestSol$
1:	$bestSol \leftarrow \emptyset; sol \leftarrow \emptyset; TED2 \leftarrow TED$
2:	<b>for</b> $i = 1..maxIter$ <b>do</b>
3:	$sort(CL, random)$
4:	<b>for each</b> $C$ in $CL$ <b>do</b>
5:	$deallocate(TED, C)$
6:	$L \leftarrow multipathRSA(TED, C)$
7:	<b>if</b> $L = \emptyset$ <b>then break</b>
8:	$allocate(TED, L)$
9:	$sol \leftarrow sol \cup L$
10:	$TED \leftarrow TED2$
11:	<b>if</b> $f(sol) > f(bestSol)$ <b>then</b>
12:	$bestSol \leftarrow sol$
13:	<b>return</b> $bestSol$

▶ RP object
▶ END-POINT object
▶ BANDWIDTH object
▼ RECORD_ROUTE object (RRO)
SUBOBJECT: Unnum Iface ID: 10.10.0.8:3
SUBOBJECT: Label Control
Type: SUBOBJECT_LABEL (3)
$n = 1, m = 1$
Label: 6a 00 00 01 01 00 00 00
SUBOBJECT: Unnum Iface ID: 10.10.0.9:2
SUBOBJECT: Label Control
SUBOBJECT: IPv4 Prefix: 10.10.0.10/32
▶ RECORD_ROUTE object (RRO)

Fig. 4. Request for P2 in PCReq (2)

▶ RP object
▶ BANDWIDTH object
▼ EXPLICIT_ROUTE object (ERO)
SUBOBJECT: Unnum Iface ID: 10.10.0.8:2
SUBOBJECT: Label Control
Type: SUBOBJECT_LABEL (3)
$n = 4, m = 4$
Label: 6a 00 00 04 04 00 00 00
SUBOBJECT: Unnum Iface ID: 10.10.0.6:3
SUBOBJECT: Label Control
SUBOBJECT: Unnum Iface ID: 10.10.0.7:2
SUBOBJECT: Label Control
SUBOBJECT: IPv4 Prefix: 10.10.0.10/32

Fig. 5. Solution for P2 in PCRep (3)

No.	Time	Source	Destination	Info
1	17 8.097681	172.16.104.2	172.16.101.3	PATH COMPUTATION REQUEST
2	18 8.097845	172.16.101.3	172.16.50.2	PATH COMPUTATION REQUEST
3	19 8.199351	172.16.50.2	172.16.101.3	PATH COMPUTATION REPLY
4	20 8.220174	10.0.0.49	10.0.0.1	Unknown Message (11). PCUpd
5	21 8.223918	10.0.0.1	10.0.0.49	Unknown Message (10). PCRpt
6	22 11.281405	10.0.0.49	10.0.0.8	Unknown Message (11).
7	23 11.285044	10.0.0.8	10.0.0.49	Unknown Message (10).
8	24 16.639926	172.16.101.3	172.16.104.2	PATH COMPUTATION REPLY

Fig. 3. Exchanged PCEP messages

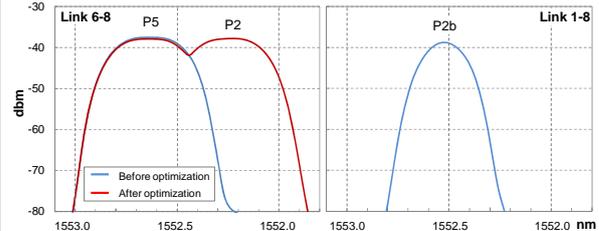


Fig. 6. Optical spectrum before and after AFRO in links 6-8 and 1-8.

#### 4. Experimental validation

Experiments have been carried out on a distributed field trial set-up connecting premises in Telefonica (Madrid, Spain), CNIT (Pisa, Italy), and UPC (Barcelona, Spain) through IPsec tunnels. Telefonica's ABNO controller was implemented in Java, CNIT's fPCE and GMPLS controllers as well as UPC's bPCE were implemented in C++ for Linux. All components communicate by exchanging PCEP messages.

We implemented the network and reproduced the scenario presented in Fig. 1. The capture in Fig. 3 shows meaningful PCEP messages exchanged. The ABNO controller (IP: 172.16.104.2) starts the network re-optimization workflow, as requested by the NMS, by sending a PCReq message (1) to the fPCE (IP: 172.16.101.3). Next, the fPCE computes the list of candidate connections (i.e., P1 and P2) and delegates the computation to the bPCE (IP: 172.16.50.2) requesting a GCO computation within a PCReq message (2). In our example, the request includes two RRO objects related to connection P2: P2a (route 8-9-10,  $m=1$ ) and P2b (8-1-2-3-4-5-10,  $m=2$ ). Fig. 4 shows part of PCReq message (2). When the bPCE receives the PCReq message, it selects and runs the algorithm with the code specified in the incoming OF object; the MP-AFRO algorithm in our case. The MP-AFRO algorithm solves the optimization problem and decides to re-route P1 through the shortest route that uses the repaired link and to merge back P2a and P2b into the original P2 connection. The solution is coded in the PCRep message (3) that is sent back to the fPCE. Fig. 5 details the response related to connection P2, where just one ERO object, specifying the new route and spectrum allocation (route 8-6-7-10 and spectrum allocation  $n=4, m=4$ ), is sent.

Upon PCRep message arrival, the fPCE starts updating connections by sending PCUpd messages towards the involved GMPLS controllers (IPs: 10.0.0.X). The controllers operate on a flexgrid test-bed derived from [8], including 100Gb/s polarization multiplexed quadrature phase shift keying (PM-QPSK) optical signals and bandwidth variable cross-connects. Fig. 6 shows the optical spectrum on links 6-8 and 1-8 before and after the re-optimization is performed. PCRpt responses are generated when the requested actions have been performed (4). When the solution computed by the bPCE has been completely deployed, the fPCE confirms the ABNO controller the end of the requested re-optimization by sending a PCRep message (5).

#### 5. Conclusions

A use case of in-operation network planning has been experimentally demonstrated on a distributed test-bed connecting premises in Telefonica (Madrid), CNIT (Pisa), and UPC (Barcelona). After a link was repaired, network re-optimization was requested from the NMS. The ABNO architecture controlled a flexgrid-based optical network, where the PCE architecture consisted of a fPCE and a bPCE. The ABNO controller is in charge of initiating the MP-AFRO workflow, requesting re-optimization to the fPCE, which delegates complex computations to the bPCE. The overall re-optimization has been successfully completed in around eight seconds, mainly due to the configuration time of the bandwidth variable cross-connects. The relevant PCEP messages were shown and its contents analyzed. Note that since network dynamicity frequently derives into optimal use of resources, this use case can easily be extended to be triggered by any other event.

#### References

- [1] L. Velasco et al., IEEE Communications Magazine, vol. 52, 2014.
- [2] M. Jinno et al., IEEE Communications Magazine, vol. 47, 2009.
- [3] M. Ruiz et al., in Proc. OSA OFC, 2014.
- [4] A. Castro et al. in Proc. IEEE RNDM, 2012.
- [5] D. King et al., IETF draft, work in progress, 2013.
- [6] E. Crabbe et al., IETF draft, work in progress, 2013.
- [7] L. Gifre et al., in Proc. OSA OFC, 2014.
- [8] F. Cugini et al., IEEE/OSA J. Lightwave Tech., vol. 30, 2013.