

Practical Methods for Allocating and Assessing Resources in Flexgrid Networks

HUSSEIN CHOUMAN^{1,3}, LUAY ALAHDAB^{1,3}, RAFAEL COLARES^{2,3}, ANNIE GRAVEY^{*3}, PHILIPPE GRAVEY³, HERVÉ KERIVIN^{2,3}, AND MICHEL MORVAN^{1,3}

¹Institut Mines Télécom, IMT Atlantique, Brest, France

²Laboratory LIMOS, CNRS UMR 6158, Clermont-Ferrand, France

³The authors contributed equally to this work

*annie.gravey@ieee.org

Compiled July 4, 2023

This paper focuses on quantifying the efficiency of different methods used to allocate resources in flexgrid optical networks. These methods are based on a recently proposed Integer Linear Programming formulation of the Routing and Spectrum Assignment (RSA) problem that takes into account all possible paths and thus theoretically yields optimal solutions, whatever be the objective function. The paper advocates using a metric-based approach for assessing RSA methods preferably to the classical approach based on the blocking probability of dynamic demands, because of the long lifetime of optical paths and of the necessity of evaluating an operational network's state early enough before congestion. The main existing fragmentation metrics are extended to network level and a family of network remaining capacity metrics, more suited to assess congestion levels, are introduced. When demands are incrementally mapped, the latter decrease quite linearly, with a slope reflecting the quality of the RSA method. Remaining capacity values are used to compare several off-line methods where the demand sets are mapped either globally or one-by-one with a suitable ordering. In both on- and off-line cases, using the sum of the demands' maximum spectrum slice index (an original objective function proposed here), provides the best performance. Finally a method to anticipate a possible congestion, based on a combination of metrics computed on the actual and a reference network instance, is presented.

<http://dx.doi.org/10.1364/ao.XX.XXXXX>

1. INTRODUCTION

In a flexgrid based optical network, the spectrum is split into frequency slices; an optical channel is characterized by one or several contiguous slices dedicated to this specific channel on all links of the path it is routed on. On the one hand, this allows a network operator to support different rates and thus to more efficiently use the available spectrum; on the other hand, the contiguity (on each link) and continuity (over all the links of a path) constraints respectively yield the well-known spectrum "horizontal" and "vertical" fragmentation issues, both implying that some spectrum slices cannot be used over some links [1].

In flexgrid optical networks, the problem of resource allocation consists in establishing lightpaths (optical paths) that compete for spectrum resources. This is called Routing and Spectrum Assignment (RSA). Off-line RSA considers a static set of demands to be served simultaneously, whereas on-line RSA deals with dynamic demands, serving them one by one. The RSA problem is usually formulated as an Integer Linear Programming (ILP) optimization problem. Thus, mapping demands on

a certain network will depend on the selected objective function.

One objective of the present paper is to apply to the RSA methods introduced in [2] to both on-line and off-line realistic frameworks. As an operational transport network should accommodate all requested demands, blocking is to be avoided. Moreover, it is well known that demands in a transport network are long-lived; therefore, this paper assumes that once established to satisfy a demand, an optical path is never released. Obviously, this does not mean that optical networks are purely static objects as new demands have to be satisfied and old demands should sometimes be modified. However, in the context of always increasing traffic, optical resources have to be regularly upgraded, either by adding new links or by modifying transceivers technology. It is quite hard to compare the respective time scales of networks upgrading and optical paths lifespan. This is not part of the present paper.

A second objective of the paper is to explore what is the "best" method for mapping demands on a flexgrid network. As blocking has to be avoided, the quality of a given RSA method shall thus not be assessed by computing the blocking rate. This is

why different metrics are discussed to define new RSA methods or to assess their quality. Some of those are specifically designed to assess the “remaining capacity” of a partially filled flexgrid network.

A last objective of the paper is to propose a practical framework for monitoring the on-line mapping procedures within an operational network.

The rest of the paper is structured as follows: Section 2 describes new ILP methods and Section 3 focuses on “network health” metrics, including both fragmentation and remaining capacity metrics. Section 4 describes the experimental framework that is used to compare RSA methods (on-line mapping in Section 5, off-line mapping in Section 7) and assess the relevance of the new remaining capacity metrics (Section 6). Section 8 proposes a network health monitoring method and conclusions are drawn in Section 9.

2. NEW METHODS FOR MAPPING DEMANDS ON A NETWORK

The RSA problem consists of mapping a given set of traffic demands on an existing topology. As the emphasis of this study lies on the RSA problem, the modulation format used by the pair of transponders serving a demand has priorly been established. Moreover, no intermediate regeneration of the lightpaths is allowed. Thus each lightpath is associated with a pair of transponders characterized by a number of spectrum slices and a transmission reach.

A. Off-line RSA problem

An offline RSA instance is composed of

- an optical network, represented as an undirected, loopless, and connected graph $G = (V, E)$, where V corresponds to the set of network nodes and E to the set of optical fibers linking the network nodes,
- a fiber length (in kms) $\ell_e \in \mathbb{R}_+$ for each $e \in E$,
- an optical spectrum (*i.e.*, available frequency slices) $S = \{1, \dots, \bar{s}\}$, where \bar{s} denotes the number of slices to be considered per link,
- a multiset K of traffic demands, where each $k \in K$ is specified by a pair of origin and destination nodes $(o_k, d_k) \in V \times V$, a required number of slices $w_k \in \mathbb{N}_+$, and a maximum transmission reach $L_k \in \mathbb{R}_+$ (in kms).

A feasible solution to the RSA problem consists of providing for each traffic demand $k \in K$, a lightpath composed of an (o_k, d_k) -path P_k in G and a transmission channel $S_k \subseteq S$ such that the following conditions are satisfied:

- the length of path P_k does not exceeds the imposed transmission reach L_k ,
- the transmission channel S_k is composed of w_k consecutive slices (*i.e.*, spectrum contiguity), that is, $S_k = \{i, i+1, \dots, i+w_k-1\}$ where $1 \leq i \leq \bar{s} - w_k + 1$,
- the transmission channel S_k assigned to traffic demand k is the same along all edges of path P_k (*i.e.*, spectrum continuity), and
- within any given link $e \in E$, a slice cannot be allocated to more than one traffic demand (*i.e.*, non-overlapping spectrum).

The goal of the RSA problem is to find the best feasible solution according to a chosen objective function. The quality of a feasible solution depends on the selected objective function. In [1], the total number of transponders is minimized, as a way to minimize the cost necessary to satisfy all demands. In [3, 4], the volume of unsatisfied traffic is minimized. In [5], the sum of lengths of the paths carrying the demands is minimized, in order to limit the lengths of selected paths and thus minimize global spectrum occupancy and delay. Furthermore, assume that slices are indexed between 1 and the total number of slices on a link. Other works aim to limit fragmentation by minimizing the max slice-index in the network in [6], or the summation, over links, of the max slice-index of each link in the network [7]. In this work, we consider four different objective functions:

- Total Slice Occupancy (*TSO*): minimize the total number of slices used across the network,
- Total Path Length (*TPL*): minimize the sum of selected path lengths,
- Maximum slice index (*MSI*): minimize the highest used frequency slice index,
- Total Slice Index (*TSI*): minimize the sum of the demands' maximal used slice index.

Objective function *TSO*, *TPL*, and *MSI* have already been considered in the literature (see [8] and the references therein) whereas objective function *TSI* is newly proposed in this work.

Notice that for a given objective function, there might exist several different optimal solutions inducing – possibly very – different mappings. In order to distinguish between these solutions we may combine the proposed objectives. More precisely, among the set of optimal solutions for a given objective, we can look for the one that optimizes a second metric. Such procedure (known as Lexicographic Method in multiobjective optimization) provides Pareto-optimal solutions. We denote by $O_i - O_j$ the combination of objectives O_i and O_j which is obtained by searching for a solution optimizing O_j within the set of optimal solutions for O_i , where O_i and O_j are distinct objectives functions.

B. ILP Formulation

The majority of ILP formulations devised to solve the offline RSA problem are based on either an edge-path model or an edge-node model (see [8] and the references therein.) The former model is usually characterized by an exponential number of variables (*i.e.*, for each demand any feasible path or lightpath is associated with a variable), a drawback bypassed by only considering a precomputed subset of paths for each demand but at the price of losing the overall optimality. The latter model is compact in terms of the number of variables and constraints. However, it suffers on the one hand, from handling the routing aspect in a less intuitive and more involved way and on the other hand, from being incomplete (*i.e.*, its feasible set is a superset of the set of feasible solutions to the offline RSA problem) which makes it unable to handle some objective functions.

The ILP formulation used in this paper was first introduced in [2] as a compact extended formulation, based on an edge-node model, that has proved to be stronger than formulations known in the literature [2, 9]. It is based on the formulation given in [9], the first complete formulation for the RSA problem, and consists of constructing a directed graph $G' = (V, A')$ from G by replacing each edge by two opposite arcs, inheriting the length

of the edge, and of replacing the three sets of binary variables considered in [9] by the set of binary variables $\{f_a^{sk} : s \in S, k \in K, a \in A'\}$ where $f_a^{sk} = 1$ if and only if demand k is routed through arc a and s is the last slice of the channel assigned to k .

Replacing each variable in the model of [9] by a linear function of the f -variables provides a multi-commodity flow formulation where each commodity corresponds to a demand. This formulation is then strengthened by adding the *disaggregated reach inequalities*

$$\sum_{a \in A} l_a f_a^{sk} - L_k \sum_{a \in \delta^+(o_k)} f_a^{sk} \leq 0 \quad \forall k \in K, s \in S \quad (1)$$

and the *disaggregated flow conservation inequalities*

$$\sum_{a \in \delta^+(o_k)} f_a^{sk} - \sum_{a \in \delta^-(o_k)} f_a^{sk} = 0 \quad \forall k \in K, s \in S, v \in V \setminus \{o_k, d_k\}, \quad (2)$$

where $\delta^+(o_k)$ and $\delta^-(o_k)$ are the sets of arcs in A' having o_k as their tail and head, respectively. The ILP formulation thus obtained requires $2|E||K|\bar{s}$ variables and is valid for every objective function considered for the RSA problem in the literature.

The four objective functions considered in this paper can be expressed as linear functions of the f -variables as follows:

- *TSO*: $\min \sum_{k \in K} w_k \sum_{a \in A'} \sum_{s \in S} f_a^{sk}$,
- *TPL*: $\min \sum_{a \in A'} \ell_a \sum_{k \in K} \sum_{s \in S} f_a^{sk}$,
- *MSI*: $\min \max_{a \in A} \max_{k \in K} \sum_{s \in S} s f_a^{sk}$ which can be trivially linearized by adding a single variable and $2|E|\bar{s}$ inequalities,
- *TSI*: $\min \sum_{k \in K} \sum_{a \in \delta^+(o_k)} \sum_{s \in S} f_a^{sk}$.

It is worth noticing that solving the ILP formulations associated with these four objective functions using branch-and-cut frameworks may be more or less time-consuming. In fact, as illustrated in [2], the spectrum-related objective functions (*i.e.*, MSI and TSI) tend to require way longer than the routing-related ones (*i.e.*, TSO and TPL) to converge towards optimal solutions. Convergence times for MSI and TSI are further discussed in Section 7.

3. NETWORK HEALTH METRICS

The present Section focuses on deriving quantitative “network health” indicators and highlighting the relationship between spectrum fragmentation and these indicators. The goal of such indicators is to address a network operator’s requirement: assuming that a given set of demands has to be mapped on a certain network, is it possible to assess how much spectrum resources are wasted and to gain insight on how much additional traffic could be supported once the initial set of demands is mapped?

A naive way of assessing the remaining capacity of a network on which an initial set of demands has already been mapped, is to submit a set of supplementary demands to be mapped till one demand is blocked, and to compute the number of supplementary demands that have been successfully mapped. However, as different supplementary sets would yield different amounts, it is necessary to consider a large number of supplementary sets in order to statistically characterize the remaining capacity of the network.

In order to avoid the above cumbersome method we propose using network health metrics that characterize the global state of the network.

After discussing state-of-the-art results regarding fragmentation metrics, which assess the fragmentation of links or paths, new network level metrics are introduced as natural extensions of existing link/path level metrics; these new metrics aim to characterize either the global fragmentation within the network or the network’s remaining capacity.

A. Existing fragmentation Metrics

External Fragmentation (*EF*) [10] is defined for each link as the ratio of the maximum number of free contiguous slices to the total number of free slices. The Utilization Entropy [11] of a link is computed by dividing the total number of slice usage status changes for all pairs of neighboring slices by the total number of slices in the link-1. Shannon’s Entropy (*SE*) [12] considers how slices are distributed on a link as follows: $SE = -\sum_{i=1}^N D_i / D \log(D_i / D)$ where D is the total number of frequency slices in the spectrum, D_i is the number of slices of the i th block of either contiguous free or contiguous used slices and N is the number of such blocks. Access Blocking Probability (*ABP*) [13] is another link-level metric; it computes the ratio between the number of possible demands that can be placed on sets of contiguous free slices on a certain link and the number that could be placed if all free slices were contiguous.

In general, a link level metric (*e.g.* EF, SE, and ABP) can easily be extended at the path level by considering the path as an aggregated link where a spectrum slice of a given index s is free if and only if the spectrum slices with index s of the all the links belonging to the path are free. The path-level metric is then computed by applying the original metric definition to this aggregated link. For example, such a method has been applied to the SE metric [14].

In [11], UE was also extended at path level, but using a different approach, *i.e.* by first computing for each slice the number of status changes between two consecutive links of the paths and dividing by the number of links-1 and then averaging this ratio over all slices.

Wasted-Unusable-Free Ratio (*WUFR*) [15] is metric that is directly defined as a path level metric: it the ratio of the number of wasted slices (those that are only free on some, but not all, of the links of a path) and unusable slices (those free on all links of a path but belonging to a block of free slices that is smaller than the number of slices requested by a transponder) to the number of free slices.

B. Alternative path-level metrics

New path-level metrics, aiming at characterizing the remaining capacity on a path, are introduced here, by extending the approach described in [15].

B.1. Wasted and accessible slices

Let $p_{m,i}$ be the i^{th} path between the $(o-d)$ pair m . On each link of $p_{m,i}$, there are used slices and free slices, the latter being partitioned into wasted and accessible slices as explained in Section 3.A; let us define $W_{m,i}$ (respectively $A_{m,i}$) as the number of wasted (respectively accessible) slices on path $p_{m,i}$. Metric $\bar{W}_{m,i}$ (respectively $\bar{A}_{m,i}$) is $W_{m,i}$ (respectively $A_{m,i}$) divided by the number of links of the path $p_{m,i}$. Dimensionless metrics are obtained by dividing the above metrics by the number \bar{s} of available slices on the links.

B.2. Number of supplementary transponders

On a given path, accessible slices are grouped into “spectrum-blocks”, that is, sets of continuous and contiguous accessible slices separated by either wasted or used slices. If $p_{m,i}$ satisfies transponder’s reach and has at least one spectrum-block j of size larger than n , then it is defined as a n -feasible path. The number of supplementary transponders with granularity n that could be accommodated on $p_{m,i}$ is:

$$X_{m,i}(n) = \sum_{j=1}^J \left\lfloor \frac{\bar{A}_{m,i,j}}{n} \right\rfloor \quad (3)$$

where $\lfloor x/n \rfloor$ denotes the integer division of x by n and J denotes the total number of spectrum-blocks in a n -feasible path $p_{m,i}$.

Fig.1 illustrates the above concepts by computing the remaining capacity metrics \bar{W}_{BE} , \bar{A}_{BE} and $X_{BE}(n)$ on path B-C-D-E between nodes B and E with 3-slice and 6-slice transponders. There are 18 used slices and 36 free slices, which are partitioned into $W_{BE} = 9$ wasted slices (in blue) and $A_{BE} = 27$ accessible slices (in orange). As there are 3 links on the path and $\bar{s} = 18$, $\bar{W}_{BE} = 3$ and $\bar{A}_{BE} = 9$. Dimensionless metrics (respectively 16.7% and 50%) are obtained if we divide by $\bar{s} = 18$. The accessible slices of the path are grouped into two spectrum-blocks: spectrum-block₁, $\bar{A}_{(BE,1)} = 3$, can accommodate one 3-slice transponder, while spectrum-block₂, $\bar{A}_{(BE,2)} = 6$, can accommodate either two 3-slice transponders or a single 6-slice transponder.

Links	Path-hole ₁			Path-hole ₂				
[B-C]	[Diagram]			[Diagram]				
[C-D]	[Diagram]			[Diagram]				
[D-E]	[Diagram]			[Diagram]				
Accessible and Wasted Slices								
W_{BE}	6	0	0	3	0	0	Total	
A_{BE}	0	9	0	18	0	0	27	
Divided by 3 (number of links)								
\bar{W}_{BE}	2	0	0	1	0	0	$\rightarrow W_{BE}/\bar{s}$ 17%	
\bar{A}_{BE}	0	3	0	6	0	0	$\rightarrow A_{BE}/\bar{s}$ 50%	
Supplementary Transponders								
Total								
$X_{BE}(n=3)$	0	1	0	2	0	0	3	
$X_{BE}(n=6)$	0	0	0	1	0	0	1	

Fig. 1. Remaining capacity metrics for Path_{BE}.

C. Network State Metrics

Being either link level or path level, the previously described metrics do not provide any quantitative insight on the whole network’s remaining capacity (i.e. the network’s capability to host future demands while taking into account the granularity of transponders and network’s wasted resources).

In the following, several ways of characterizing the global state of the network are considered.

C.1. Total number of holes

This metric is the sum of the number of holes on all the links (a spectrum hole on a link is a set of contiguous free slices).

C.2. Network level fragmentation metrics

For a given path-level metric, the following methodology is followed. For a given demand, all feasible paths, among the five shortest paths computed for the (o-d) pair of the demand are considered and the value of the path-level metric is computed for each feasible path; then a (o-d)-level metric is obtained by averaging the previously computed path-level metric values.

The network level metric is then computed as the average overall $(o-d)$ pairs.

Note that both versions of network-level UE metric that were proposed by the authors of [11] differ from the above one. Indeed, in the first one, they followed the same approach to derive a path-level metric by considering all network links in an arbitrary order, while the second one was based on averaging the path-level metrics on the shortest path between all $(o-d)$ pairs.

Note also that the followed methodology also differs from the one used in [16], where the average is made on all network links. Averaging on paths instead of links enables capturing the spectrum continuity issue and gives a more representative weight to the links belonging to many paths compared to those that are rarely used.

C.3. Network remaining capacity

The previous metrics focus only on fragmentation but do not address whether or not the network is congested. In the following, alternative metrics focus on the remaining capacity in the network.

For $(o-d)$ pair m , let I_m denote the total number of paths that can be used for m and $I_m(n)$ denote the number of n -feasible paths ($I_m(n) \leq I_m$). For each path i of $(o-d)$ pair m and path-level remaining capacity metric $RC_{m,i}$ (either $\bar{W}_{m,i}$, $\bar{A}_{m,i}$, or $X_{m,i}(n)$), network level metrics are obtained by first averaging $RC_{m,i}$ over the number of feasible paths I (which can be I_m or $I_m(n)$ depending on the considered $RC_{m,i}$ metric), and then averaging over the M $(o-d)$ pairs. A network level remaining capacity metric is thus defined as:

$$RC = \left(\sum_{m=1}^M \frac{\sum_{i=1}^I RC_{m,i}}{I} \right) / M. \quad (4)$$

When $RC_{m,i} = X_{m,i}(n)$, the corresponding remaining capacity metric shall be called $X(n)$.

4. EXPERIMENTAL FRAMEWORK

A simple framework has been selected in order to compare the various RSA methods. Intermediate regeneration of the lightpaths is not considered, which implies that each lightpath is associated to a pair of transponders. For the sake of simplicity and in order to focus on the RSA problem considered in this paper, network design and the optical feasibility of given paths have been abstracted: the modulation format used by the transponder(s) serving a given demand is assumed to be defined prior addressing the RSA problem and each transponder T is simply characterized by a bit rate, the number of occupied spectrum slices and a transmission reach $T(br, ss, tr)$.

Therefore, a given demand to the optical transport network is expressed as a number of spectrum slices to be carried between an o-d pair. There may be several distinct demands between a given (origin, destination) pair). The present paper considers transmission systems operating in the extended C-band, corresponding to a 4-THz spectrum with a nominal number of slices that equals 320 (i.e., 12.5 GHz per slice). The previous assumptions imply that between each o-d pair, there should exist at least one path shorter than the maximum reach allowed by the transponder. Specifically, in all experiments presented in this paper, the following three transponders $T3, T5, T6$ are considered:

- $T3 = T(100Gbit/s, 3, 3000km)$;
- $T5 = T(200Gbit/s, 5, 1500km)$;

- $T6 = T(400\text{Gbit}/s, 6, 600\text{km})$.

Two widely used network topologies are considered, namely the 17-node German network (Fig.2) and a modification of the NSF topology (M-NSF), which is based on the 9-Node NSF topology with 13 links [17] (Fig.3). The modification on the original NSF topology consists in dividing its link lengths by 2 to make it compatible with the transmission reach of the considered transponders.

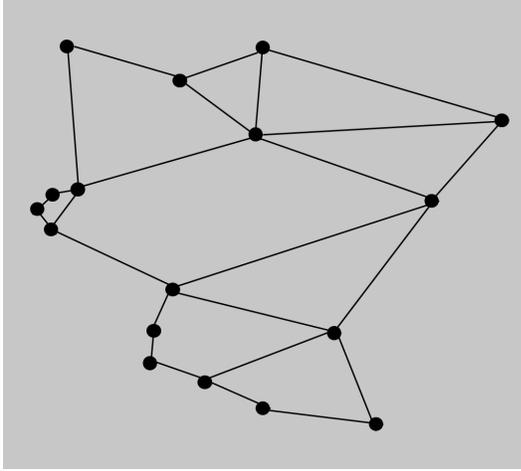


Fig. 2. 17-nodes German network topology

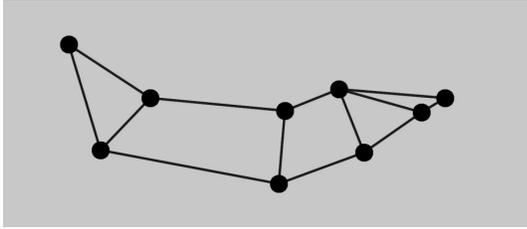


Fig. 3. 9-nodes M-NSF network topology

A set of demands is generated according to the following procedure:

- the total number of initial demands in a set is fixed (200); each demand is considered to be bi-directional as both directions are reserved when a demand is mapped;
- for each initial demand :
 - the number of slices is chosen in $\{3, 5, 6\}$ with equal probability;
 - an o-d pair is uniformly selected within the $N(N - 1)$ possible pairs;
 - when the reach associated to an initial demand is not compatible with the shortest path for the selected o-d pair, the demand is split into 2, or more, actual demands with a larger reach, (i.e. using a different transponder); this process stops as soon as the reach of the actual demand is large enough. For example, if the shortest path for the o-d pair is 1000 km, a 6-slices demand ($400\text{Gbit}/s$) shall be split into two 5-slices demands ($200\text{Gbit}/s$), while if the shortest path is 2000 km, the same demand shall be split into four 3-slices demands ($100\text{Gbit}/s$).

This procedure implies that the final number of demands may slightly vary from one set to another.

To run simulations that use the above optimization model, the number \bar{s} of available slices on each link needs to be specified. For example, a fiber with a 4-THz spectrum has a nominal number of slices that equals 320 (i.e., 12.5 GHz per slice). However, it would be better to decrease this number in order to decrease both the number of variables and constraints in the ILP model and consequently, the time necessary to obtain an optimal solution. Let an *admissible link configuration* be a number \bar{s} of available slices on each link that guarantees no blocking demands and no links whose proportion of used slices exceeds a given threshold T (e.g., $T = 80\%$). For a given s value of the number of slices on each link, let $P_{used}(s)$ denote the maximum proportion of used slices over all links in the configuration that minimizes $\sum_{k \in K} \sum_{a \in \delta^+(o_k)} \sum_{s \in S} f_a^{sk}$; the number of blocked demands for that configuration is $N_{blocked}(s)$. After checking that $P_{used}(320)$ is smaller than T and that $N_{blocked}(320) = 0$, Algorithm 1 is used to heuristically derive an *admissible link configuration*.

Algorithm 1. Heuristic for defining \bar{s}

```

1: procedure DEFINE  $\bar{s}(320, 40)$   $\triangleright \bar{s}$  decreased by steps of 40
2:    $\bar{s} \leftarrow 320$ 
3:   while  $N_{blocked}(\bar{s}) = 0$  and  $P_{used}(\bar{s}) < T$  do
4:      $\bar{s} \leftarrow \bar{s} - 40$ 
5:   return  $\bar{s}$   $\triangleright \bar{s}$  is an admissible link configuration

```

On a typical set of 40 demands, using Algorithm 1 with $T = 80\%$, a value for \bar{s} of 120 can be selected for both Germany and M-NSF networks.

In transmission networks, demands are not set up and broken very often; indeed, in many operational networks, new demands are added till the network is considered congested and is then reorganized, possibly adding new resources or considering new transponders. This is why we consider in our study that once a demand is mapped, the resources it occupies are unavailable for any other demand. Blocking occurs when a new demand cannot be mapped on the partially filled network (i.e. not enough available contiguous slices on a sufficiently short path).

As each actual demand corresponds to a given number of allocated spectrum slices, an interesting indicator of a partially filled network is the number of *mapped slices*, i.e. the total number of slices corresponding to the set of demands currently mapped on the network.

5. ON-LINE MAPPING

In the present Section, some of the tools introduced in Sections 2 and 3 are used to design on-line mapping methods; these are then compared with several existing ones. One method, which is shown to be among the most efficient ones, is then selected to be used as a reference for on-line mapping in the rest of the paper.

A. RSA Heuristic methods based on fragmentation metrics

First, the approach described in [11] and [15] that respectively used the UE and WUFR metrics to derive RSA heuristics, is extended to other path-level fragmentation metrics presented in Section 3, namely EF, SE and ABP. Then, new RSA heuristics based on network-level metrics are introduced. In the following, heuristics RSA methods based on path-level and network-level

metrics are named MetricName-P and MetricName-N respectively.

Lastly, two new RSA heuristics involving the $X(n)$ functions introduced in Section 3.C are presented as X-P and X-N. For these heuristics, for a demand of n slices, all possible mappings are considered on all feasible paths (among a pre-computed set of shortest paths) for the $(o - d)$ pair of the demand.

- using X-P, only the paths corresponding to the $(o - d)$ pair m of the demand are considered and the mapping that maximizes $X_{m,i}(n)$ over all possible mappings over all feasible paths i is selected.
- using X-N, all $(o - d)$ pairs are considered and the mapping that maximizes the average value of $X(n)$ over all paths and all $(o - d)$ pairs is selected as described in Equation (4).

B. ILP methods for On-line RSA

Four original On-Line Optimal (ONO) RSA methods based on the formulation and the four objective functions introduced in Section 2 are named accordingly ONO-ObjectiveName.

C. On-line RSA performance

The performance of the different on-line RSA methods is assessed by applying each of them to both Germany and M-NSF networks, and by considering 128 sets of demands in each case. As explained in Section 4, the number of demands per set is not constant. For the German network, the number of demands (resp. slices) in a given set varies from 207 to 217 (resp. 950 to 1034). For the NSF network, these figures are respectively 243 to 249 for demands and 1082 to 1174 for slices. Starting from an empty network, demands are mapped one by one till the first blocking occurs, and the number of allocated slices is recorded.

Two classical on-line RSA heuristics are considered as benchmarks:

- FAR-FF (Fixed Alternative Routing - First Fit)[18], which selects the shortest available path among a set of pre-computed paths list for the $(o - d)$ pair; then, the slices are selected as first fit; FAR-FF is thus fragmentation unaware.
- LLR-FF-ACC (Least Loaded Route - First Fit - Accessible) [19], which selects the least loaded path, i.e. the one providing the highest number of accessible slices for the $(o - d)$ pair, and then select the slices as first fit; LLR-FF-ACC is thus fragmentation aware.

The results regarding the heuristic methods are first presented in Fig.4 and Fig.5 that show for each network topology and RSA method, using box-plot representation [20], the distribution of allocated slices obtained with the 128 demand sets. In the chosen box-plot representation, whiskers are drawn within the 1.5 Inter-quartile range value.

Even though the number of mapped slices strongly depends on the actual set of demands, several trends may be noted when comparing these methods:

- in most cases, metric-based methods provide significantly better results when operated at network rather than path level, the only exception being X-N in the German network case, which is not quite as good as X-P;
- for both networks, X-P provide the best results among the path level-based methods;

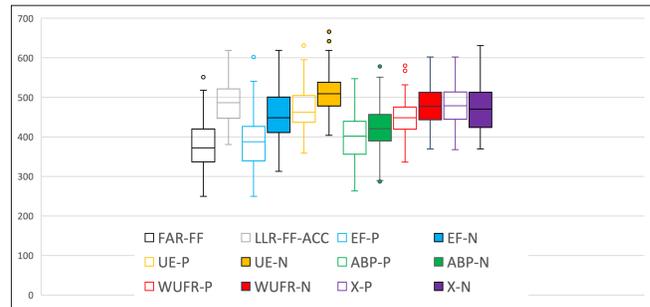


Fig. 4. Total mapped slices distribution among 128 demand sets with the on-line RSA Heuristics (German network)

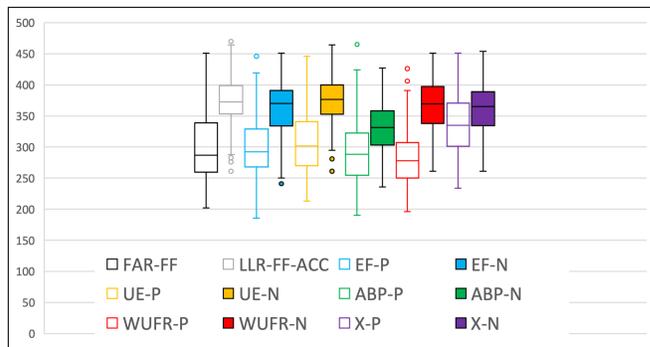


Fig. 5. Total mapped slices distribution among 128 demand sets with the on-line RSA Heuristics (M-NSF network)

- regarding the average mapped-slices numbers, the best values are obtained by UE-N in both cases although LLR-FF-ACC is a close second, especially in the M-NSF network case.

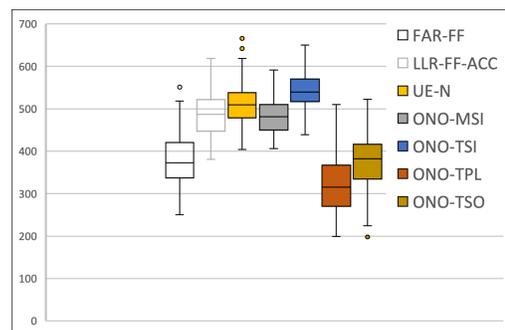


Fig. 6. Total mapped slices distribution among 128 demand sets using ILP with four different objectives and three heuristics as benchmarks (German network)

The results achieved with the ILP formulation with the four objectives are then presented in Fig. 6 and Fig. 7. In both figures, these results are bench-marked with the classical FAR-FF and LLR-FF-ACC methods and with the best heuristics, namely UE-N.

- ONO-TSI provides the best results for both networks, in front of UE-N and LLR-FF-ACC, the figures being very close in the M-NSF case;
- ONO-MSI is next, with a slightly worse performance;

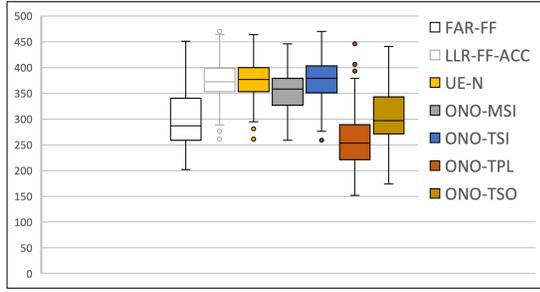


Fig. 7. Total mapped slices distribution among 128 demand sets using ILP with four different objectives and three heuristics as benchmarks (M-NSF network)

- the other methods provide the worst performances, with ONO-TSO performing globally better than FAR-FF and ONO-TPL;

Based on these results, ONO-TSI is selected to be used as a reference for on-line mapping in the rest of the paper and for assessing network health evolution in the next Sections.

6. RELEVANCE OF REMAINING CAPACITY METRICS

According to the definition of the remaining capacity metric $X(n)$, and assuming, to simplify the discussion, that the total number of slices is an integer multiple of n , this metric exhibits the following features:

- $nX(n) = \bar{s}$ in an empty network;
- $X(n)$ decreases when a new demand is mapped;
- if $X(n) = 0$, it is impossible to satisfy any demand requesting the mapping of n slices;

In order to facilitate computing $X(n)$, the number of considered paths for a given $(o - d)$ pair has been limited to the 10 (respectively 5) shortest ones (in terms of physical length) for the German network (respectively for the M-NSF network). Note that this procedure should preserve the three above mentioned features of $X(n)$.

In the present Section, the behavior of $X(n)$ is assessed in order to know to which extent it could provide useful information on the network health once a demand set has been mapped and on the performance of the RSA method used to achieve this mapping.

To this end, the evolution of $X(n)$ during the on-line experiments of Section 5 is first analysed. Then, another experiment is introduced in order to correlate the values of $X(n)$ in a partially filled network with the numbers of slices that can be mapped on top of the initial mapping.

A. Evolution of $X(n)$ metric

Among the algorithms used in Section 5, the emphasis is put on those based on ILP formulation using objectives TSI, MSI, TPL and TSO. As shown previously, the two former are representative of well-performing RSA on-line methods, while the latter present significantly worst performances. How $X(n)$ for $n = 1, 3, 5, 6$ varies during the 128 experiments of mapping demands one by one is then analysed.

An example for a specific set of demands is given in Fig. 8 that shows the evolution of $X(1)$ values as a function of the

number of mapped slices for one of the 128 demand sets when using the four RSA methods. Linear regressions are included, and the corresponding determination coefficient R^2 is provided. The linear fits of $X(1)$ are quite good and computed slopes differ

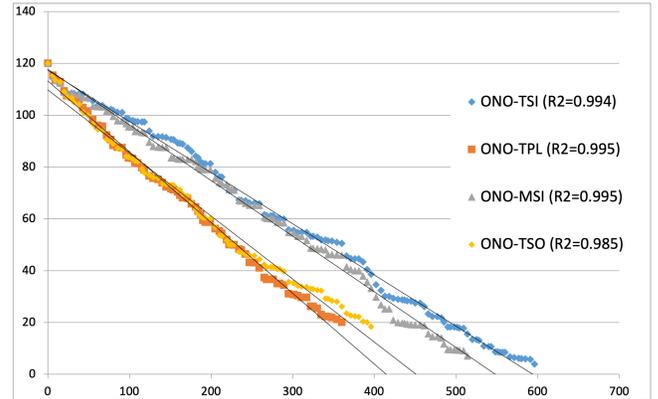


Fig. 8. Evolution of $X(1)$ vs the number of mapped slices for German network (set 2)

significantly between the well- and bad-performing methods. Moreover, for the latter, the first blocking occurs at larger $X(1)$ values than for the former.

Linear fits of the $X(n)$ values for n in $\{1, 3, 5, 6\}$ are then shown, using box-plot representation, in Fig. 9 for the Germany network and in Fig. 10 for the M-NSF network. The linear regression slopes displayed are those of $nX(n)$, in order to ease the comparison between the different values of n . The focus is here on objectives TSI and TSO, as the results achieved with objective MSI and TPL are close to those obtained with TSI and TSO, respectively.

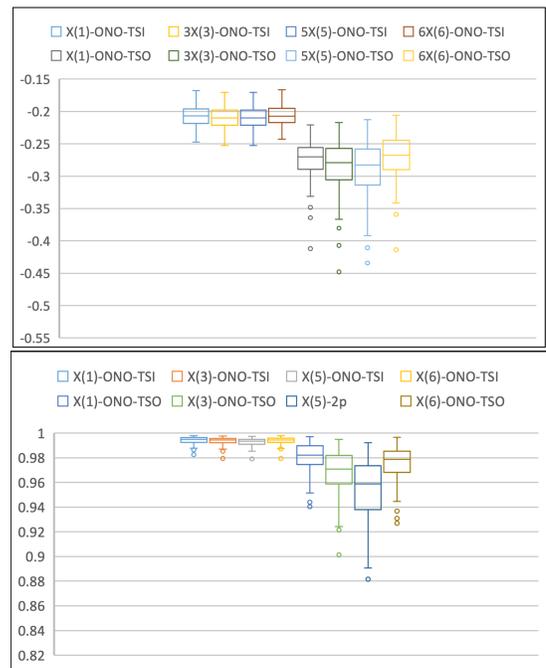


Fig. 9. Distribution of the slope (top) for $nX(n)$ and the R^2 coefficient of determination (bottom) for $X(n)$ among 128 demand sets for the German network

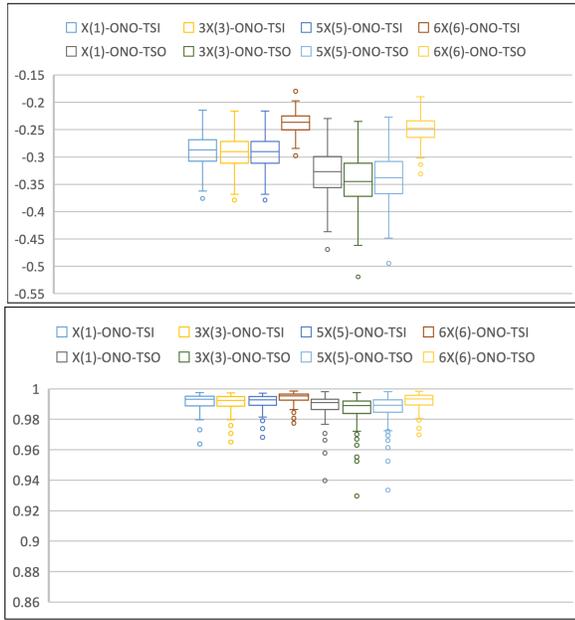


Fig. 10. Distribution of the slope (top) for $nX(n)$ and the R^2 coefficient of determination (bottom) for $X(n)$ among 128 demand sets for the M-NSF network

- In the German network, the slopes of the $nX(n)$ linear regression vary within ranges of values that are nearly identical for all values of n ; the quality of the linear fits is quite similar for all values of n for objective TSI and better than the one achieved with objective TSO, for which a quality degradation from $n = 1$ to $n = 5$, and an improvement for $n = 6$ are observed.
- in the M-NSF network, for $n = 1, 3$ or 5 , the distribution of the slopes as well as the quality of the linear fits are very close; for $n = 6$, the slope is less pronounced than for the smaller n values. For this last case, it is worth noting that there is a feasible path for the $n = 6$ transponder only for 17 out of the 34 (o, d) pairs, which indeed makes it difficult to compare $X(6)$ with the other $X(n)$.

It can be concluded from the above analysis that the $X(n)$ value computed after a significant number of demands has been mapped (i.e. the network is “partially filled”) can be used to compare different RSA methods. This is illustrated by Fig. 11, which displays, for the four ILP objectives, the distribution of $X(1)$ values computed after mapping the 40 first demands of the 128 demand sets. Clearly, ranking the ILP objectives according to the $X(1)$ distributions is equivalent to ranking them using the method described in Section 5 (see Figs. 6 and 7).

B. Correlation between $X(n)$ and blocking

The previous results show that, comparing the $X(n)$ values achieved with different RSA algorithms applied to many demand sets, provides a useful insight on their relative performances. This shall be used in Section 7 to compare off-line RSA methods. However, one may wonder whether the mere knowledge of the obtained $X(n)$ for different mappings of a given set of demands may be used to compare the quality of these different mappings: if a mapping yields a larger $X(n)$ value than another one, does it ensure that more slices can be mapped on top of the first mapping than on top of the second one ?

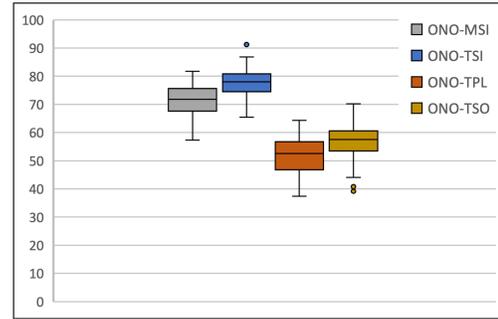


Fig. 11. Distribution of $X(1)$ values computed after mapping 40 demands using ILP formulation with four different objectives (German network)

Another experiment is then introduced to try answering the above question. Firstly, an initial set of demands is mapped, resulting in remaining capacity metrics values. Five initial sets, generated according to the procedure described in Section 4, are considered; the first 40 demands of these sets are mapped using various RSA methods and the resulting $X(n)$ values are computed. The numbers of slices corresponding to the initial 40 demand sets vary between 183 and 196 for the German network and between 170 and 190 for the modified-NSF one. It was chosen to map 40 demands, roughly corresponding to a bit less than 200 mapped slices since using on-line mapping, blocking typically occurs between 300 and 600 mapped slices, depending on the RSA method.

Secondly, the demands of the 128 random sets used in Section 5 are mapped one by one, using the ONO-TSI method, on top of the initial mapping, till the first blocking; the total number of mapped slices are then computed for all the sets of additional demands.

For the German and modified-NSF networks, 75 and 52 different initial mappings were respectively generated by using different RSA methods, based on the algorithms and objective functions presented in Section 2. As the present section’s objective is to assess the correlation between the initial $X(n)$ values and the ability to serve additional demands on top of this initial mapping, the methods used for the latter are not detailed.

In the following the focus is put on $X(1)$ but similar results, not shown here, have been obtained with the other $X(n)$ metrics. The results regarding the German network are presented for four of the five sets in Fig. 12 depicting for each set the median, the 95th and the 5th percentiles. For all these sets two distinct regions roughly separated by the $X(1) = 60$ line are observed:

- in both regions, there is some variation of the number of additionally mapped slices but it is much more pronounced in the low $X(1)$ region;
- in the high $X(1)$ region, the minimum number of additional slices that can be mapped is close to 200;
- in the low $X(1)$ region, this number may be close to zero in the worst cases;
- results vary slightly from one set to the other, in particular, lower results in the high $X(1)$ region for set 4 and a lesser variation between high and low $X(1)$ for set 2.

The results for set 5, not shown here, are in line with these observations.

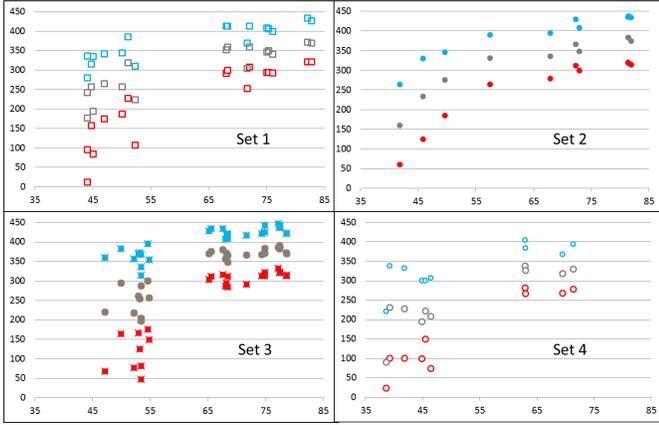


Fig. 12. Additional mapped slices versus initial $X(1)$ values. 4 initial mappings over the German network, 128 additional demand sets. Median (grey), 95th percentile (blue), 5th percentile (red)

The corresponding results for the M-NSF network shown in Fig. 13 present globally the same features, with more variations between demand sets.

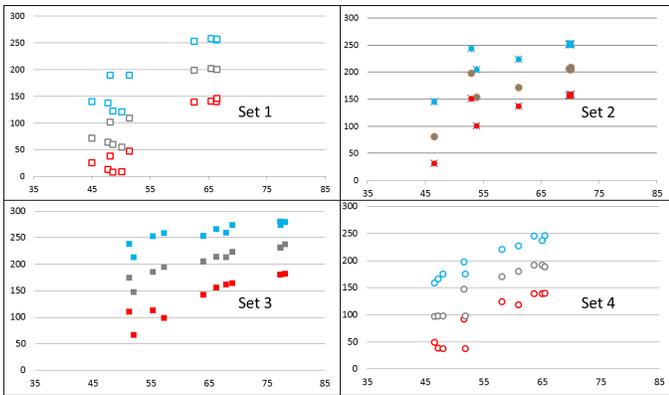


Fig. 13. Additional mapped slices versus initial $X(1)$ values. 4 initial mappings over the M-NSF network, 128 additional demand sets. Median (grey), 95th percentile (blue), 5th percentile (red)

The following conclusions can be drawn from this analysis:

- any mapping presenting a *high* $X(n)$ ensures that a large number of additional slices can be mapped, for any of the considered random set;
- conversely, for a mapping in the *low* $X(n)$ region, the number of additional slices that can be mapped can be very large or very small, depending on the considered random sets;
- when two initial mappings are considered, one in each region, the number of additional slices that can be mapped on the *high* $X(n)$ initial mapping is statistically significantly larger than the one that can be mapped on the *low* $X(n)$ initial mapping.

7. OFF-LINE MAPPING PERFORMANCE

The off-line RSA problem consists of selecting paths and allocating spectral resources to a set of demands. Section 2 has

presented a RSA-algorithm and different objectives functions, which may be optimized using this algorithm. In the present section, different mappings obtained using these objective functions are compared by using the $X(n)$ metrics. As previously, the German and M-NSF networks are considered.

A straightforward way to use the RSA algorithm is to apply it to the overall demand set. By construction, this process guarantees that the obtained mapping will optimize the objective function. This method is called *Off-line Optimal* (OFO). Using OFO with objective functions TPL and TSO, the convergence time is quite small; it is longer for TSI and it is always much longer for MSI. These general results regarding TSI and MSI are illustrated in Table 1: three sets of respectively, 30, 40, and 50 demands are mapped the M-NSF network case. A maximum computation time has been fixed to two hours; when an optimum solution is not found before reaching this limit, the relative gap between upper and lower bounds is indicated.

Table 1. OFO-TSI and OFO-MSI computation times for M-NSF instances.

Number of demands	OFO-TSI	OFO-MSI
30	47.41s	1453.51s
40	148.23s	7200s (1.07%)
50	4205.89s	7200s (9.99%)

The above results show that OFO may not be applied with TSI and MSI to realistically large instances, due to the complexity of the RSA problem. Therefore, a less complex, alternative approach consisting in applying the RSA algorithm sequentially by mapping the demands one by one is introduced; this method is called *Off-line Heuristic* (OFH). OFH provides potentially sub-optimal mappings; on the other hand, obtaining mappings with OFH is always quite rapid. Therefore, it is to be expected that for a (large) operational network, when it is not possible to compute an optimal mapping, heuristic methods such as OFH should be considered to map sets of demands. OFH is similar to the method used for solving the on-line RSA problem, but presents a major difference because in the off-line problem all demands are known and the order in which they are processed can be chosen. It is to be expected that the final mapping (and consequently the $X(n)$ values) will depend on the selected ordering. Among the many possible demands permutations, some specific orderings are considered in the following and the impact of the ordering is assessed for both OFH and OFO.

Let us define the (o, d) distance as the length of the shortest path (in terms of physical length) between o and d . The requested capacity for a demand of s slices between o and d is defined as the product between s and the minimum number of hops on the considered paths between o and d (10 paths for Germany and 5 for M-NSF). Six orderings are then specified:

- Slice-based ordering - largest to smallest (SBLS): demands are ordered by decreasing number of requested slices; when demands request the same numbers of slices, they are sorted by decreasing (o, d) distances;
- Slice-based ordering - smallest to largest (SBSL): demands are ordered by increasing number of requested slices; when demands request the same numbers of slices, they are sorted by increasing (o, d) distances;

- Distance-based ordering - longest to shortest (DBLS): demands are ordered by decreasing (o, d) distances;
- Distance-based ordering - shortest to longest (DBSL): demands are ordered by increasing (o, d) distances;
- Capacity-based ordering - largest to smallest (CBLS): demands are ordered by decreasing capacity value; when demands request the same capacity, they are sorted by decreasing (o, d) distances;
- Capacity-based ordering - smallest to largest (CBSL): demands are ordered by increasing capacity value; when demands request the same capacity, they are sorted by increasing (o, d) distances;

Regarding the objective functions, the four objective functions already used in the previous Sections and some composite functions following the approach presented in Section 2 are considered. As seen previously, the *spectrum unaware* objectives functions TPL and TSO yield significantly worst performances than the *spectrum aware* objectives functions TSI and MSI. However, it is interesting to investigate to which extent a combination of a primary *spectrum unaware* objective followed by a secondary *spectrum aware* objective could improve the quality of the obtained mapping. For the sake of simplicity, the only results that are presented are those obtained with composite objectives TSO-TSI and TPL-MSI. Also for the sake of simplicity, the focus is put on the $X(1)$ metric, as Section 6.A shows that the various $X(n)$ metrics evolve quite similarly.

Considering the six selected orderings, both simple and selected composite objective functions, and using either OFO or OFH, the 40 demands of each of the five demand sets introduced in Section 6.B are mapped. Whenever a mapping is obtained, $X(1)$ is computed; then, mean and standard deviation of $X(1)$ computed over the six considered orderings are computed and reported in Table 2. Note that, due to a limit set on convergence time, OFO mappings with MSI in the experiments over the German network are not obtained.

A. Assessing OFO performance

Table 2 shows that in the OFO case, the demand ordering has a very limited impact on $X(1)$ when spectrum aware objective functions (TSI, MSI) are used either as single or as a secondary objective. With objectives TPL and TSO, the $X(1)$ value depends slightly more on the demand ordering, but this doesn't affect the comparison between the different objectives.

Regarding single objective functions, in the German network TSI outperforms both TPL and TSO, as expected. In the M-NSF network, where MSI results are available, the results obtained with this objective are close to those obtained for TSI but never higher. TPL and TSO are behind TSI and MSI but the discrepancy is lower than in the German case and significantly varies with the demand set.

Considering composite objectives TSO-TSI and TPL-MSI, it appears that complementing a spectrum-unaware primary objective with a secondary one which is spectrum-aware greatly improves the performance in comparison with the single objective case. In the German network, the $X(1)$ values achieved with composite objectives remain lower than those achieved with TSI. In the M-NSF case, TSO-TSI provides an higher $X(1)$ value for one set, while TPL-MSI yields significantly lower values than those achieved with MSI.

To summarize, when using OFO, TSI provides the best performance (which is consistent with the results obtained in the on-line case) regardless of the ordering of the demands which has an insignificant impact on the obtained metrics values.

B. Assessing OFH performance

Table 2 shows that in the OFH case, regardless on demand ordering, objective functions TPL and TSO yield significantly lower results than the four other ones, in nearly all instances, the only exception being M-NSF network with set 1 and objective TSO. This shows that spectrum-aware objective functions should be preferred in OFH, either as single objective function, or as a secondary objective function.

However, as demand ordering has a significant impact on $X(1)$ which is quite pronounced with objectives TSI and MSI, especially for the German network, it is necessary to assess the impact of demand ordering on the four spectrum-aware OFH mapping methods (TSI, MSI, TSO-TSI, TPL-MSI). This is done by ranking the obtained $X(1)$ values in the considered 40 instances where OFH is applied (five demand sets, four spectrum aware objective functions on two networks). The obtained rankings are displayed in Fig. 14.

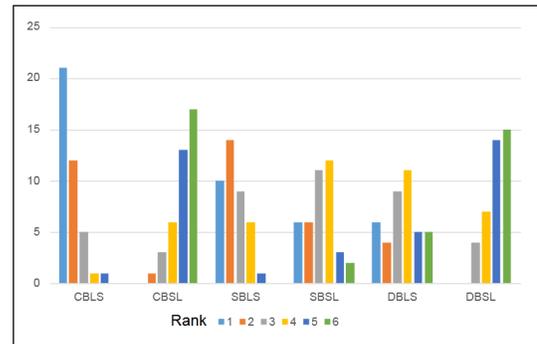


Fig. 14. Obtained rankings of the $X(1)$ values. 6 ordering of demands, 5 sets of 40 demands, 4 spectrum-aware OFH mapping methods (TSI, MSI, TSO-TSI, TPL-MSI) for both German and M-NSF networks.

Fig. 14 shows that CBLS, being ranked first or second in 33 instances performs quite well. CBSL and DBSL are badly ranked in most cases. SBLS is also correctly ranked in general (first or second in 24 instances) while the performances of SBSL and DBLS depend on the considered instance. These results may be interpreted as follows:

- serving first the demands that potentially require the largest amount of spectral resources (CBLS) is generally the best strategy; and the opposite one (CBSL) is logically the worst one;
- serving first the demands with the largest (o, d) distance (DBLS) is smarter than doing the contrary (DBSL) but as this a spectrum unaware strategy, its efficiency is still poor;
- grouping the demands according to the numbers of requested slices makes sense as it likely limits fragmentation; starting with the largest numbers of requested slices (SBLS) seems more appropriate but the opposite (SBSL) also performs reasonably well.

Based on the previous results, CBLS is used to compare more precisely the performances obtained with objective functions

Table 2. Mean and standard deviation of $X(1)$ over the 6 considered orderings, for 5 sets of 40 demands

German Network						
Objective	TSI	TPL	MSI	TSO	TSO-TSI	TPL-MSI
	Mean/StD	Mean/StD	Mean/StD	Mean/StD	Mean/StD	Mean/StD
Set1 OFO	79.8/0.1	47.7/2.4	./.	48.9/2.3	69.6/0.2	66.3/0.2
Set1 OFH	73.5/4.1	50.2/2.6	68.2/4.5	53.8/1.2	65.8/2.7	62.3/2.4
Set2 OFO	81.5/0.1	45.4/2.0	./.	50.5/1.7	77.6/0.1	57.4/0.2
Set2 OFH	74.6/4.4	44.2/4.3	69.4/4.9	54.2/2.6	71.9/2.0	53.7/2.4
Set3 OFO	77.4/0.1	52.8/2.1	./.	53.9/1.4	71.1/0.3	66.1/0.6
Set3 OFH	72.0/3.3	53.3/1.8	67.8/4.7	55.6/5.2	68.3/2.6	64.9/3.3
Set4 OFO	71.4/0.1	36.3/1.6	./.	46.2/1.1	62.8/0.2	40.8/0.6
Set4 OFH	63.8/5.9	37.5/1.4	59.2/7.6	48.7/1.9	59.1/3.8	40.1/1.7
Set5 OFO	82.8/0.1	44.4/1.9	./.	47.0/2.4	77.2/0.6	55.2/0.6
Set5 OFH	73.3/2.2	45.9/2.4	69.2/4.2	51.6/1.7	68.3/1.3	52.2/1.3

M-NSF Network						
Objective	TSI	TPL	MSI	TSO	TSO-TSI	TPL-MSI
	Mean/StD	Mean/StD	Mean/StD	Mean/StD	Mean/StD	Mean/StD
Set1 OFO	78.9/0.1	65.4/0.7	78.9/0.9	70.5/1.4	80.8/0.0	70.6/0.3
Set1 OFH	74.6/1.8	66.1/1.3	73.5/2.1	72.1/2.7	79.5/1.2	69.2/1.8
Set2 OFO	69.9/0.2	48.6/2.8	69.2/0.2	52.2/1.5	69.7/0.2	62.0/0.2
Set2 OFH	65.5/3.6	52.0/2.9	60.3/6.0	56.9/4.6	67.5/2.9	58.8/2.7
Set3 OFO	78.2/0.1	49.3/1.2	77.1/0.5	55.2/2.7	75.4/0.1	55.3/0.5
Set3 OFH	71.9/3.7	50.6/1.8	68.4/2.6	65.7/2.1	73.2/2.1	55.1/1.5
Set4 OFO	65.5/0.1	47.8/1.2	64.6/0.2	50.4/2.2	65.0/0.1	53.4/0.4
Set4 OFH	61.8/3.9	49.6/2.4	58.1/5.2	56.3/2.8	63.8/1.6	54.1/1.8
Set5 OFO	66.4/0.0	47.7/1.2	66.4/0.2	50.7/1.6	59.3/0.6	52.1/0.4
Set5 OFH	62.5/3.8	45.2/2.0	58.2/3.8	52.1/1.7	58.3/1.5	52.2/0.4

TSI, MSI, TSO-TSI and TPL-MSI. This is done by comparing the $X(1)$ values obtained with the CBLS ordering, for five sets of 40 demands and with the four spectrum-aware OFH mapping methods on both considered networks. The obtained results are displayed in Fig. 15.

Fig. 15 shows that in the German network, the four methods are identically ranked for the five sets, the method using TSI being ranked first. The methods using TSI, MSI and TSO-TSI provide close values with all sets while the results of the TPL-MSI method depend strongly on the considered set. As seen in previous studies, M-NSF results are less contrasted: TPL-MSI yields always the lowest values, while TSI performs the best in four of the five sets and TSO-TSI is best in the last one.

To summarize, when using OFH, a method based on the CBLS ordering and on objective function TSI provides a consistently good performance; slightly less efficient methods are SBLs ordering and objectives functions MSI or TSO-TSI.

C. Comparing OFO and OFH

The previous results show that for both OFO and OFH, using TSI as an objective function provides the best results; for OFH, it is also necessary to focus on the CBLS ordering. It now seems worth comparing OFO and OFH.

This is done for a subset of the previously considered cases; in

those four cases (3 related to the German network, one to the M-NSF network), the $X(1)$ values obtained after mapping 40 initial demands with either OFO or OFH using objective function TSI and CBLS ordering, are compared; then, on top of each set of two partially filled networks, ONO-TSI is used to map one by one the demands of the 128 sets described in Section 4 till a first blocking occurs; this allows comparing the respective numbers of additionally mapped slices. The obtained results are reported in Table 3 and Fig. 16.

Table 3. OFO versus OFH $X(1)$ ratios after the initial mapping of 40 demands and ratios of average of additionally mapped slices

Network	Germany			M-NSF
	1	3	5	
Initial set of demands	1	1.04	1.09	1
$X(1)_{OFO}/X(1)_{OFH}$	1.04	1.04	1.08	1.02
Average of additional mapped slices ratio (OFO/OFH)	1.06	1.04	1.08	1.02

The OFO over OFH $X(1)$ ratio for four different instances in Table 3 varies between 1.00 and 1.09. This suggests that OFO

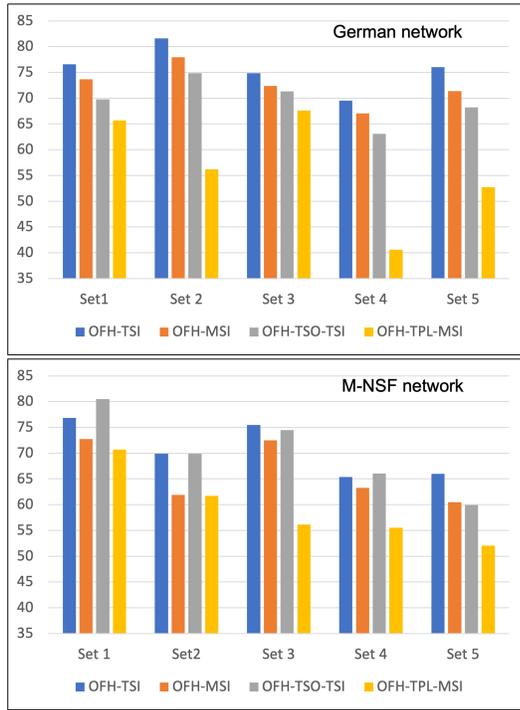


Fig. 15. $X(1)$ values obtained with CBLs ordering. 5 sets of 40 demands, 4 spectrum-aware OFH mapping methods (TSI, MSI, TSO-TSI, TPL-MSI) for both German (top) and M-NSF (bottom) networks.

works slightly better than OFH. This is confirmed by the ratios of average of additionally mapped slices over the partially filled networks. These last values are in the same range as the $X(1)$ ratio, with quite similar set-to-set variations.

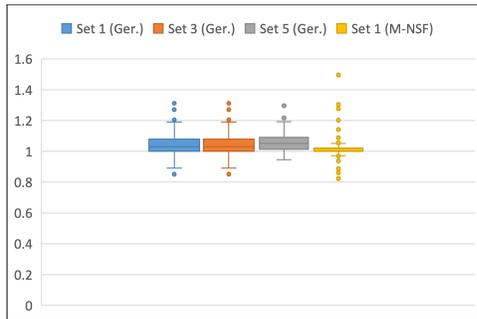


Fig. 16. Distribution of the additionally mapped slice ratio between OFO and OFH initial mapping with objective TSI among 128 additional demand sets, for 4 initial sets of 40 demands (3 for German and 1 for M-NSF network)

It is seen on the global distribution of the additionally mapped slice ratios shown in Fig. 16 that, although OFO statistically allows mapping more slices than OFH, this is not always the case.

To summarize, the above results show, as expected, the superiority of OFO over OFH when objective function TSI and ordering CBLs are used; however, their performances are quite close. From a practical point of view, this can be very interesting in operational networks due to the complexity issue of OFO. Intermediate approaches, where the ordered demands are served

in OFH by blocks (with a size providing a suitable computation time) instead of being handled one by one, could possibly allow approaching even closer the performance of OFO. This is outside the scope of the present paper.

The results in Table 3 also confirm the pertinence of $X(1)$ as an indicator of the quality of a given mapping, as small differences in $X(1)$ also correspond to small differences in average on the number of additional demands that can be accommodated on a partially filled network. Possible methods for using the $X(n)$ metrics to predict network congestion are discussed in the next section.

8. USING METRICS TO PREDICT CONGESTION

Based on the evolution of $X(n)$ as a function of the mapped slices observed in Section 6.A, the present Section investigates whether monitoring $X(n)$ when performing on-line mapping can serve to predict an incoming network congestion. As $X(n)$ decreases when demands are mapped successively, a possible way to predict such congestion could be to compare its value with a given threshold.

A. Applying $X(n)$ metric to predict congestion in on-line mapping: basic scenario

The scenario first considered, similar to the one studied in Section 6.A, assumes that all demands are served one by one using the same RSA algorithm and that once mapped, a demand remains permanently.

Fig. 17 shows the distribution of the last value taken by $3X(3)$ before the first blocking, computed over 128 different demand sets applied to the German network for the 4 on-line RSA methods ONO-MSI, ONO-TSI, ONO-TPL and ONO-TSO. $X(3)$ has been chosen as congestion indication metric but results would be quite similar for any of the other $X(n)$ metrics.

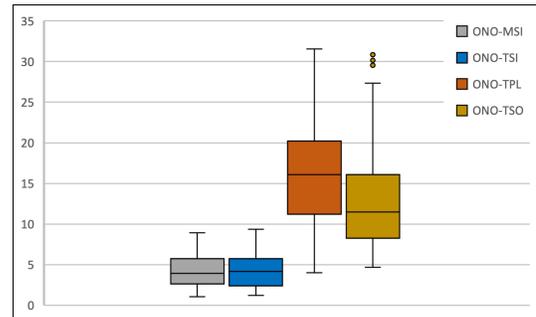


Fig. 17. Distribution of last value of $3X(3)$, before the first demand blocking among 128 demand sets for the German network

As seen in Fig. 17, the last $X(3)$ value before blocking are globally higher and more widely spread for a non spectrum-aware method (ONO-TSO, ONO-MSO) than for a spectrum-aware method (ONO-TSI, ONO-MSI). Therefore, the threshold value should be significantly higher for the former. This is illustrated in Fig. 18 that represents the distribution of the number of additional mapped slices after $X(3)$ becomes lower than 15: looking at the 5% percentile, it is seen that with ONO-MSI and ONO-TSI, it is possible to add at least 100 slices whereas the corresponding numbers of additional slices for ONO-TPL and ONO-TSO are less than 50.

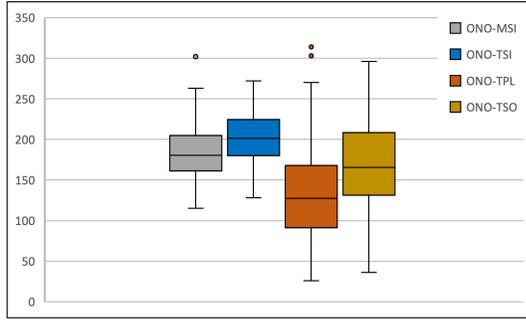


Fig. 18. Distribution of the number of additional mapped slices after $X(3)$ becomes lower than 15, among 128 demand sets for the German network

In order to gain some insight on the pertinence of such a threshold-based approach, the focus is put on ONO-TSI and ONO-TSO methods; Fig. 19 shows the number of additional slices that can be mapped once $X(3)$ has crossed a given threshold; it is computed over 128 demand sets for the German network. With Fig. 19, it is seen that the number of additional mapped slices behaves quite linearly as a function of the threshold value set for $X(3)$ for both TSI and TSO. Thus, it may be expected that monitoring the evolution of $X(3)$ should allow warning the network operator that the number of slices that can be surely mapped becomes too low.

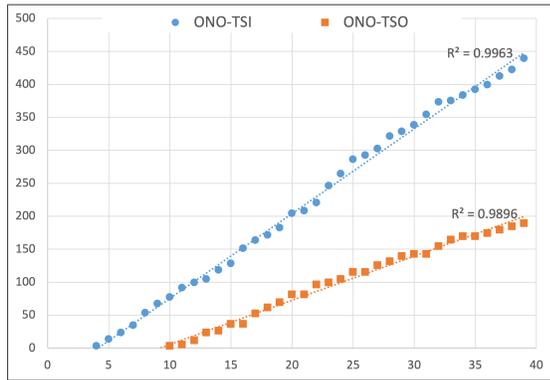


Fig. 19. Minimum number of additional mapped slices vs $X(3)$ threshold

The above results suggest that a threshold on $X(n)$ could indeed provide an estimation of the minimum additional number of slices that could be served before congestion occurs. However, a threshold on the number of already mapped slices could also be used in this simple scenario as the relationship between $X(n)$ and the number of mapped slices has been shown in Section 6.A to be close to linear, especially for spectrum aware RSA methods.

B. Applying $X(n)$ metric to predict congestion in on-line mapping: real-life scenario

In an operational network, some demands may be released and thus, in the absence of rerouting of the remaining demands, the spectrum could suffer from some excess fragmentation even if the demands are mapped using a spectrum-aware algorithm such as ONO-TSI. Other operations, e.g. demand rerouting in response to fibre or equipment failure, could also lead to significant discrepancies with the ideal scenario considered in the

previous Section. Using a simplistic threshold-based approach (either on $X(n)$ or on the number of mapped slices) cannot thus be directly applied.

In order to take account of such situations, it is proposed to use a threshold based approach augmented with a comparison between the actual $X(n)$ value ($X^{\text{actual}}(n)$) and the value it would take in a reference scenario ($X^{\text{ref}}(n)$), based on a well-performing RSA algorithm. For this purpose, OFH-TSI with CBLs ordering is selected as the well-performing RSA algorithm since Section 7 has shown that it provides a good trade-off between performance and scalability. Let

$$\Delta X(n) = X^{\text{ref}}(n) - X^{\text{actual}}(n)$$

Whenever the set of mapped slices is modified, $X^{\text{actual}}(n)$ and $X^{\text{ref}}(n)$ are computed. Algorithm 2 is then applied.

Algorithm 2. Network Health monitoring algorithm

- 1: **procedure** MONITORING($(\Delta X/X)_{\text{thr}}$, $X_{\text{thr}}(n)$)
- 2: **if** $\Delta X(n)/X^{\text{ref}}(n) > (\Delta X/X)_{\text{thr}}$ **then**
- 3: Remap demands
- 4: **if** $X^{\text{ref}}(n) < X_{\text{thr}}(n)$ **then**
- 5: Send warning : Impending Congestion

Algorithm 2 states that when the relative difference between $X^{\text{actual}}(n)$ and $X^{\text{ref}}(n)$ is higher than a given threshold ($(\Delta X/X)_{\text{thr}}$), a demand re-mapping with the reference algorithm is triggered. Introducing such a threshold avoids too frequent re-mappings. If $X^{\text{ref}}(n)$ becomes lower than a threshold ($X_{\text{thr}}(n)$) the re-mapping may still be performed but in addition, a message is generated to warn the operator about a congestion risk that could require specific measures, such as activating or deploying new resources.

It is worth noting that $X^{\text{actual}}(n)$ and $X^{\text{ref}}(n)$ can easily be computed and interpreted by a centralized network controller in order to trigger the requested actions. In particular, in large networks, computing $X(n)$ is much simpler than simulating the mapping of a set of random additional demands.

The choice of the threshold values introduced in the above procedure depends on the actual use case, in particular the considered topology and operator's policy. For example, in the case of the German network, plausible values may be derived from the $X(n)$ behaviour commented in Section 6.A:

- $(\Delta X/X)_{\text{thr}} = 5\%$, as the comparison between OFO-TSI and OFH-TSI in Table 3 shows that relative differences of a few percent in $X(n)$ indeed translates into significant performance differences;
- $X_{\text{thr}}(n) = 40/n$, as Fig. 19 shows that, using on-line mapping with TSI objective, a threshold of 40 on $3X(3)$ enables to map (in 95% of the considered cases) about 100 additional slices, which represent about 20% of the total mapped slices; therefore such a threshold value could be used to call for new resources.

9. CONCLUSION AND PERSPECTIVES

This work addressed two key aspects of spectral resource management in flexgrid optical networks, namely their allocation and how to assess the quality of this allocation.

An exact RSA method, recently proposed by some of the authors [2], has been applied to various scenarios. As it takes

into account all possible paths, it theoretically yields optimal solutions, whatever be the objective function.

Scenarios design and results evaluation are important features of the present work. The classical approach based on a dynamic demand scenario in which the blocking probability is the performance metric has not been considered.

This raised the issue of metrics to assess the quality of an RSA at the network level. A first group of metrics was built by extending the main existing fragmentation metrics, originally defined at either link or path-levels, to $(o - d)$ pair and network levels. However, in order to evaluate how far from congestion is the network, an alternative approach has been proposed by introducing a family of network remaining capacity metrics $X(n)$, where n is the number of spectral slices requested by a hypothetical demand. When plotted against the total number of spectrum slices of the mapped demands, it has been shown that all $X(n)$ metrics decrease quite linearly, with a slope that varies with the actual demand sets, among a range that depends on the chosen objective. These newly defined metrics have been used to design heuristic methods to perform on-line RSA.

The proposed RSA methods and newly defined metrics have been applied to practical scenarios relying on the German network topology and on a second one derived from the 9-node NSF topology.

Among all on-line methods, the most efficient one is based on an ILP formulation that minimizes *the sum of the demands' maximal used slice index* (TSI). In particular, it is more efficient than the one based on minimizing *the max slice index among all links* (MSI) that was used in previous works.

The $X(1)$ values are used to compare several off-line methods, with either single or combined objectives, where the demand sets are mapped either globally (OFO) or one-by-one with different orderings (OFH). As in the on-line case, minimizing TSI was shown to be the best performing objective for most instances for both OFO and OFH. Using the best objective and a suitable ordering, OFH has been shown to perform almost as well as OFO in several small instances. This could be important for the application to large instances, for which OFO is not applicable contrary to OFH. Using a routing-oriented primary objective followed by a spectrum-oriented secondary one, also improves the scalability.

Finally, the paper discusses how a remaining capacity metric such as $X(n)$ can be used to monitor network health and thus trigger corrective actions such as spectrum defragmentation and/or deployment of new resources. A procedure that could be applied in an operational context is suggested.

A first point that could deserve further investigations in more sophisticated scenarios would be to identify the best usage of the many $X(n)$ metrics as the reported results do not show significant differences related to the chosen n value.

Physical layer limitations were accounted for in the ILP formulation by using a maximum reach constraint depending on the requested amount of spectrum slices (that is implicitly on the modulation format). There should be no major difficulty to replace this simple approach with an interface between the present optimization software and a more accurate tool such as GNPpy [21].

A natural extension of the present work would be to adapt the proposed RSA methods to space-division multiplexing and/or multi-band WDM.

10. FUNDING AND ACKNOWLEDGMENTS

FUNDING

This work has been partly founded by the French Agence Nationale de la Recherche within the framework of the FLEXOPTIM research project (ANR- 17-CE25-0006).

ACKNOWLEDGMENTS

When most of this work was carried out, Annie Gravey and Philippe Gravey were with IMT Atlantique, Brest, France.

11. REFERENCES

REFERENCES

1. D. Amar, E. Rouzic, N. Brochier, J.-L. Auge, C. Lepers, N. Perrot, and S. Fazel, "How problematic is spectrum fragmentation in operator's gridless network?" in *ONDM*, (2014), pp. 67–72.
2. C. Rafael, K. Hervé, and W. Annegret, "An extended formulation for the constrained routing and spectrum assignment problem in elastic optical networks," in *Proceedings of the Joint ALIO/EURO International Conference 2021-2022 on Applied Combinatorial Optimization*, (2022), pp. 5–10.
3. M. Ruiz, M. Pióro, M. Żotkiewicz, M. Klinkowski, and L. Velasco, "Column generation algorithm for RSA problems in flexgrid optical networks," *Photonic Netw. Commun.* **26** (2013).
4. M. Klinkowski, M. Pióro, M. Żotkiewicz, M. Ruiz, and L. Velasco, "Valid inequalities for the routing and spectrum allocation problem in elastic optical networks," in *ICTON*, (2014), pp. 1–5.
5. M. Żotkiewicz, M. Pióro, M. Ruiz, M. Klinkowski, and L. Velasco, "Optimization models for flexgrid elastic optical networks," in *ICTON*, (2013), pp. 1–4.
6. K. Walkowiak, R. Goścień, M. Klinkowski, and M. Woźniak, "Optimization of multicast traffic in elastic optical networks with distance-adaptive transmission," *IEEE Commun. Lett.* **18**, 2117–2120 (2014).
7. A. Ghallaj, R. Romero Reyes, M. Ermel, and T. Bauschert, "Optimizing spectrum allocation in flex-grid optical networks," in *Photonic Networks; 18. ITG-Symposium*, (2017), pp. 1–8.
8. B. M. Luis, "A branch and cut algorithm for the routing and spectrum allocation problem," Ph.D. thesis, Universidad de Buenos Aires, Argentina (2022).
9. Y. Hadhbi, H. Kerivin, and A. Wagler, "A novel integer linear programming model for routing and spectrum assignment in optical networks," in *FEDCSIS*, vol. 18 (2019), pp. 127–134.
10. D. Amar, E. Rouzic, N. Brochier, J.-L. Auge, C. Lepers, and N. Perrot, "Spectrum fragmentation issue in flexible optical networks: analysis and good practices," *Photonic Netw. Commun.* **29** (2015).
11. X. Wang, Q. Zhang, I. Kim, P. Palacharla, and M. Sekiya, "Utilization entropy for assessing resource fragmentation in optical networks," (2012).
12. P. Wright, M. C. Parker, and A. Lord, "Simulation results of Shannon entropy based flexgrid routing and spectrum assignment on a real network topology," in *ECOC*, (2013), pp. 1–3.
13. D. Amar, E. Rouzic, N. Brochier, E. Bonetto, and C. Lepers, "Traffic forecast impact on spectrum fragmentation in gridless optical networks," in *ECOC*, (2014), pp. 1–3.
14. P. Wright, M. C. Parker, and A. Lord, "Minimum- and maximum-entropy routing and spectrum assignment for flexgrid elastic optical networking [invited]," *J. Opt. Commun. Netw.* **7**, A66–A72 (2015).
15. F. Pederzoli, D. Siracusa, A. Zanardi, G. Galimberti, D. L. Fauci, and G. Martinelli, "Path-based fragmentation metric and RSA algorithms for elastic optical networks," *IEEE/OSA J. Opt. Commun. Netw.* **11**, 15–25 (2019).
16. L. H. Bonani, M. Forghani-elahabad, and M. L. F. Abbade, "Network fragmentation measure in elastic optical networks," in *2019 SBFoton International Optics and Photonics Conference (SBFoton IOPC)*, (2019), pp. 1–5.

17. P. Pavon-Marino, S. Azodolmolky, R. Aparicio-Pardo, B. Garcia-Manrubia, Y. Pointurier, M. Angelou, J. Sole-Pareta, J. Garcia-Haro, and I. Tomkos, "Offline impairment aware RWA algorithms for cross-layer planning of optical networks," *J. Light. Technol.* **27**, 1763–1775 (2009).
18. X. Wan, L. Wang, N. Hua, H. Zhang, and X. Zheng, "Dynamic routing and spectrum assignment in flexible optical path networks," in *2011 Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference*, (2011), pp. 1–3.
19. Z. Zhu, W. Lu, L. Zhang, and N. Ansari, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," *J. Light. Technol.* **31**, 15–22 (2013).
20. Y. Benjamini, "Opening the box of a boxplot," *The Am. Stat.* **42**, 257–262 (1988).
21. A. Ferrari, M. Filer, E. Le Rouzic, J. Kandrát, B. Correia, K. Balasubramanian, Y. Yin, G. Grammel, G. Galimberti, and V. Curri, "Gnpy: an open source planning tool for open optical networks," in *2020 International Conference on Optical Network Design and Modeling (ONDM)*, (2020), pp. 1–6.