Make-Before-Break Wavelength Defragmentation

Brigitte Jaumard, Hamed Pouya, David Coudert

Make-Before-Break Wavelength Defragmentation

Brigitte Jaumard[1], Hamed Pouya[1], David Coudert[2]
[1] *CSE, Concordia University, Montreal (Qc) H3G 1M8 Canada*
[2] *Université Côte d'Azur, Inria, CNRS, I3S, UNS, Sophia Antipolis, France*

Invited Paper

ABSTRACT

Future optical networks, in particular Software Defined Optical Networks (SDONs), are expected to provide reconfigurable services while maintaining an efficient usage of wavelength resources. In this paper, we propose a Make-Before-Break (MBB) wavelength defragmentation process which minimizes the bandwidth requirement of the resulting provisioning. We next compare the latter provisioning with a minimum bandwidth provisioning that is not subject to MBB.

The resulting solution process is thoroughly tested on various data and network instances. Numerical experiments show that, on average, the best seamless lightpath rerouting is never more than 5% away (less than 1% on average) from an optimal lightpath provisioning.

Keywords: Wavelength Defragmentation, Seamless Defragmentation, Make Before Break Rerouting, Routing and Wavelength Assignment, Fragmented Network, Network Reconfiguration.

1. INTRODUCTION

With the deployment of Reconfigurable Optical Add/Drop Multiplexer (ROADM) networks and newly developed Elastic Optical Networks (EONs), service providers (SPs) are able to reconfigure wavelengths or frequency slots of optical paths remotely. Indeed, as lightpaths are set up and torn down quite often, the bandwidth resources become fragmented. It results in higher blocking probability though there is enough capacity to satisfy new demands. Wavelength/spectrum defragmentation is then required in order to re-optimize the bandwidth usage.

Wavelength/spectrum defragmentation consists of three phases [1, 2]: *(i)* deciding when to conduct a reconfiguration ; *(ii)* designing a new lightpath provisioning with a given optimization objective (e.g., minimum bandwidth requirement or equivalently maximum Grade of Service (GoS), i.e., maximizing the number of granted requests), which ensures the most possible seamless defragmentation; and *(iii)* migrating from the current lightpath fragmented provisioning to the new optimized one, in the most seamless possible fashion. The focus of our paper is on the third phase. In particular, we study the minimum trade-off to be made in order to obtain a seamless migration to an optical lightpath provisioning that requires minimum bandwidth.

In the context of wavelength defragmentation, a network reconfiguration scheme that requires no disruption corresponds to *Make-Before-Break* (MBB) reachable wavelength provisioning, i.e., such that new lightpaths are first established, traffic in the old lightpaths is switched to the new lightpaths, and then bandwidth resource of old lightpaths are released. Within that context, an optimized provisioning is one that reduces significantly the bandwidth requirement, with ideally the smallest possible number of reroutings. Deciding whether there exists an MBB reconfiguration from a current fragmented provisioning to an optimized pre-computed one by rerouting on-going connections one after the other can be done in polynomial time [3]. An MBB reconfiguration can be easily defined using a *Move-To-Vacant* (MTV) algorithm, i.e., sequentially choosing a connection, finding a new path using spare resources and then switching the connection to its new path. The process goes on until the new state of the network satisfies the desired constraints, e.g., overall usage of resources or no vacant path is found for any of the demands. However, the rerouting order plays a significant role in how far we can go in terms of minimizing the bandwidth requirements.

The paper is organized as follows. In Section 2, we propose a reconfiguration framework in order to investigate the Make-Before-Break wavelength defragmentation problem. Section 3 proposes a nested decomposition optimization model and algorithm, called WDF_MBB and WDF_NCG respectively, which models/computes the minimum bandwidth Routing and Wavelength (RWA) provisioning that is reachable with Make-Before-Break. Extensive numerical results are presented in Section 4.

2. MBB WAVELENGTH DEFRAGMENTATION FRAMEWORK

We present in Section 2.1 an overview of a wavelength defragmentation framework for computing an MBB minimum bandwidth requirement re-provisioning. We next recall (Section 2.2) the concept of dependency graph, which allows the feasibility check of an MBB rerouting, and if not feasible, allows the identification of blocking elements. Lastly, we propose (Section 2.3) some rerouting operations that allow to reduce the number of blocking situations.

2.1. Overall Framework

We assume we are given a Wavelength-Division Multiplexing (WDM) optical network. Each connection request, if granted, is to be provisioned by a lightpath, i.e., the combination of a route and a wavelength, the same wavelength (so-called continuity constraints) all the way from the source to the destination. The framework we propose is as follows.

Initialization:
  Compute a maximum GoS RWA provisioning, i.e., grant the largest possible number of connections.

Step 1: Dynamic RWA Provisioning
  Repeat
    Free the resources of each terminating connection.
    Grant each incoming connection if there are spare resources to provision it, else deny it.
  Until a wavelength defragmentation is triggered

Step 2: Trigger Defragmentation
  When the deterioration of the GoS reaches a given threshold, trigger defragmentation.
  Let RWA^{FRAG} be the resulting fragmented RWA provisioning.

Step 3: Conduct Defragmentation
  Step 3.1. Compute RWA^{OPT}, a minimum bandwidth requirement RWA provisioning
  Step 3.2. Initialize $\text{RWA}^{\text{MBB-OPT}}$ with RWA^{OPT}
    If $\text{RWA}^{\text{MBB-OPT}}$ is MBB reachable from RWA^{FRAG}
      Reroute one request at a time following an MBB strategy
      Return to **Step 1**
    Else
      Identify some rerouting deadlocks, i.e., situation in which multiple requests are blocking each other such that neither request can be rerouted.
      Recompute a minimum bandwidth RWA provisioning with the deadlock avoidance constraints (see Section 3)
      Let $\text{RWA}^{\text{MBB-OPT}}$ be the new optimized RWA provisioning
      Return to **Step 3.2.**

2.2. Dependency Graph and Lightpath Rerouting Order

In order to check whether we can conduct an MBB rerouting, and if possible, in which order the lightpaths can be rerouted, we build the so-called dependency graph $G_{\text{D}} = (V_{\text{D}}, L_{\text{D}})$, introduced in [4]. A dependency graph is a directed graph that represents the rerouting order dependence between requests in order to migrate from the fragmented (RWA^{FRAG}) and the optimized (RWA^{OPT}) provisionings at the end of each defragmentation interval. Node and link sets of the dependency graph are defined as follows.

$$V_{\text{D}} = \{\pi = (p, \lambda) : \pi \text{ is a lightpath in RWA}^{\text{FRAG}}\}$$
$$L_{\text{D}} = \{(\pi, \pi') : \pi' \text{ needs to be rerouted before } \pi \text{ in order to reach RWA}^{\text{OPT}} \text{ with MBB}\}.$$

In other words, each link (π, π') defines the order of migration between two lightpaths, when a lightpath π' needs to be rerouted before another lightpath π in order to perform an MBB rerouting. We can then conduct an MBB rerouting of the requests if and only if G_{D} is a Directed Acyclic Graph (DAG), using the labeling of the nodes in any topological sorting[1] of G_{D}. Once G_{D} is not a DAG, we compute the set

---
[1]The topological sorting of a Directed Acyclic Graph (DAG) is a linear ordering of the nodes such that for every link (π, π'), node π comes before π' in the ordering

of strongly connected components. Any circuit in a strongly connected component defines a deadlock. We discuss in Section 3.2 how to translate deadlock into constraints within the optimization model that determines an RWA provisioning.

2.3. WDF_NCG **Algorithm: Reducing the Number of Deadlocks**

We now propose WDF_NCG Algorithm to modify the current RWA$^{\text{OPT}}$ in order to reduce the number of deadlocks, without modifying the GoS. Note that while reducing the number of deadlocks, we may increase the rerouting sequence length, i.e., nodes in the dependency graph. Details of WDF_NCG Algorithm follow.

Algorithm 1 Preprocessing Algorithm

Input: Dependency graph G_{D}
Input: $G_\lambda = (V, L_\lambda) \subseteq G = (V, L)$ where L_λ is the set of links which are not used in any lightpath using λ, for $\lambda \in \Lambda$.
Output: New wavelength assignment for some lightpaths involved in circuits
  1: Compute the strongly connected components (SCCs) of G_{D}
  2: Build the list Π of lightpaths, i.e, the set of nodes in G_{D}, involved in the SCCs
  3: Sort Π by non-increasing in-degree in G_{D}
  4: **for** every node (lightpath) $\pi = (p, \lambda) \in \Pi$ such that $p : v_s \rightsquigarrow v_d$ **do**
  5:     **for** every $\lambda' \in \Lambda$ **do**
  6:         $\hat{p} \leftarrow$ shortest path from v_s to v_d in $G_{\lambda'}$
  7:         **if** $\hat{p} \neq \emptyset$ and $|\hat{p}| \leq |p|$ **then**
  8:             Check if changing π to $(\hat{p}, \lambda')$ creates new circuits in G_{D}
  9:             **if** No new circuit is generated **then**
 10:                 Set $\pi = (\hat{p}, \lambda')$; Update G_λ and $G_{\lambda'}$; Go to step 1
     return Set of lightpaths

Algorithm 1 helps us break some of the circuits. If the resulting dependency graph is a DAG, it means that a seamless migration is found, and we can proceed with an MBB defragmentation. Otherwise, some deadlock constraints corresponding to circuits are added to the re-computation of an optimized RWA re-provisioning. We detail its computation in the next section.

3.   A NESTED DECOMPOSITION MODEL/ALGORITHM FOR COMPUTING RWA$^{\text{OPT}}$

In this section, we discuss how to compute an optimized RWA provisioning, called RWA$^{\text{OPT}}$, and how to modify it as little as possible so that it can be MBB reachable from RWA$^{\text{FRAG}}$. We discuss the optimization model in Section 3.1 and its solution in Section 3.2.

3.1. WDF_MBB **Model**

We now describe the WDF_MBB model that computes the minimum bandwidth RWA provisioning subject to the deadlock avoidance constraints identified in the dependency graph (if there is no deadlock constraints, WDF_MBB computes an optimized RWA provisioning). Since the number of circuits in a directed graph can be exponential, we do not introduce all possible circuits (deadlock situations).

The WDF_MBB model relies on the concept of wavelength configurations, where a configuration γ corresponds to a wavelength plan, for a given wavelength λ, i.e., the set of all lightpaths using a given wavelength (forcing all such lightpaths to be pairwise link disjoint). While in the RWA decomposition models, configuration (i.e., wavelength plan) can be defined for a generic wavelength and therefore the resulting model does not have any symmetry issues (the wavelength assignment is made a posteriori), this is unfortunately not possible here in order to be able to express the rerouting deadlock avoidance constraints.

Model WDF_MBB requires one unique set of decision variables z_γ for $\gamma \in \Gamma$. Each variable z_γ allows or not the selection of configuration γ in the optimal RWA provisioning output. Each configuration $\gamma \in \Gamma$ is formally defined by the following parameters:
B^γ = bandwidth requirement of configuration γ, as expressed by the number of links used in γ for

routing some connections ; a_{sd}^γ = number of lightpaths between node pair $(v_s, v_d) \in \mathcal{SD}$ in configuration γ ; $a_\pi^\gamma = 1$ if lightpath π is used in configuration γ, 0 otherwise.

Let D_{sd} be the number of lightpaths between node pair (v_s, v_d) in RWA$^{\text{FRAG}}$.

$$\min \quad \sum_{\gamma \in \Gamma} B^\gamma z_\gamma \quad (1) \qquad \sum_{\gamma \in \Gamma} a_{sd}^\gamma z_\gamma \geq D_{sd}, \qquad (v_s, v_d) \in \mathcal{SD} \quad (3)$$

$$\text{subject to:} \qquad\qquad \sum_{\gamma \in \Gamma} \left(\sum_{\pi \in c} a_\pi^\gamma \right) z_\gamma \leq |c| - 1 \qquad c \in C \qquad (4)$$

$$\sum_{\gamma \in \Gamma_\lambda} z_\gamma \leq 1 \qquad \lambda \in \Lambda \quad (2) \qquad z_\gamma \in \{0, 1\} \qquad\qquad \gamma \in \Gamma. \qquad (5)$$

Constraints (2) ensure that we select at most one configuration for a given wavelength λ. Constraints (3) enforce the demand constraints, with the left-hand side term computing the number of demand units provided by each configuration, and then summing over all configurations. Constraints (4) guarantees that the provisioning will not contain any of the circuits $c \in C$, where C denotes the set of embedded circuits (deadlocks). Constraints (5) define the domains of the variables.

3.2. WDF_NCG Algorithm

The WDF_MBB model proposed in the previous section has an exponential number of variables, and therefore is not scalable if solved using classical ILP (Integer Linear Programming) tools. Indeed, we need to use column generation techniques in order to manage a solution process that only requires an *implicit* enumeration of the wavelength configurations. Using a column generation technique consists in solving alternatively a restricted master problem (the WDF_MBB model in Section 3.1 with a very limited number of columns/variables) and the pricing problem (generation of a new wavelength configuration) until the optimality condition is satisfied (i.e., no wavelength configuration with a negative reduced cost).

Once the optimal solution of the LP (Linear Programming) relaxation ($z_{\text{LP}}^\star$) has been reached, we solve exactly the last restricted master problem, i.e., the restricted master problem of the last iteration in the column generation solution process, using a branch-and-bound method, leading then to an ε-optimal ILP solution ($\tilde{z}_{\text{ILP}}$), where $\varepsilon = (z_{\text{LP}}^\star - \tilde{z}_{\text{ILP}})/ z_{\text{LP}}^\star$. Branch-and-price methods can be used in order to find optimal solutions, if the accuracy (ε) is not satisfactory, see, e.g., [5, 6].

In order to solve the linear relaxation of model WDF_MBB, we started from the column generation CG++ algorithm in [7] and improve it with respect to the solution of the pricing problems: rather than using two formulations of the pricing problems, a link one with PP$_{\text{LINK}}$, and a path one with PP$_{\text{PATH}}$, we used only the PP$_{\text{PATH}}$, and we solved optimally using a column generation algorithm (rather than a very small set of pre-computed paths), hence resulting in a nested column generation framework.

Due to the lack of space, we omit the detailed description of PP$_{\text{PATH}}$, but it can be deduced from the one in [7], with the addition of a polynomial solvable pricing problem for generating the paths.

4. NUMERICAL RESULTS

We use four different networks, whose key characteristics (number of nodes and links, average node degrees) are described in Table 1. In order to generate an RWA provisioning, we use the framework described in Section 2. The first initial provisioning corresponds to the RWA provisioning obtained in Jaumard and Daryalal [9], with the objective of maximizing the Grade of Service (GoS), i.e., the number of granted requests. We then use a dynamic RWA process with ADD and DROP requests, using a random generator such that the probability of ADD and DROP is the same, i.e., 0.5. Source and destination of ADD and DROP requests are selected at random. Defragmentation is then triggered with a threshold on the deterioration of the GoS.

Table 1: Main characteristics of the networks (W = number of wavelengths)

Networks	$\mid V \mid$	$\mid L \mid$	Average degree	W
USA [7]	24	88	3.7	75
GER [7]	50	176	3.5	130
NTT [7]	55	144	2.6	42
CONUS [8]	60	158	2.6	50

We evaluate the compromise to be made on the minimization of the bandwidth requirement in order to get a seamless, i.e., Make-Before-Break, wavelength reconfiguration. We report the results in Table 2. No data instance requires more than 5 rounds of the iterative process of the WDF_NCG algorithm and six instances are solved in 1 round. NTT with threshold 25% requires the larger number of deadlock avoidance constraints, i.e., 11 additional constraints. The increase of the bandwidth requirement is very modest and rarely exceeds 1% for an optimized wavelength provisioning that is Make-Before-Break reachable.

Table 2: Bandwidth requirement compromise for a Make-Before-Break reachable optimized wavelength provisioning

	Defrag, trig. event	Bandwidth requirement (number of wavelength links)			$\text{RWA}^{\text{MBB\_OPT}}$ vs. RWA^{OPT}	RWA^{FRAG} vs. $\text{RWA}^{\text{MBB\_OPT}}$
		RWA^{FRAG}	RWA^{OPT}	$\text{RWA}^{\text{MBB\_OPT}}$		
NTT	10%	2,907	2,546	2,558	0.47%	12.0%
	15%	3,009	2,645	2,645	0.0%	12.1%
	20%	2,971	2,537	2,600	2.4%	12.5%
	25%	2,920	2,464	2,470	0.2%	15.4%
USA	5%	3,931	2,798	2,814	0.6%	28.4%
	10%	3,691	2,593	2,601	0.3%	29.5%
	15%	3,573	2,545	2,548	0.1%	28.7%
GER	10%	8,347	6,386	6,386	0.0%	23.5%
	15%	7,893	5,984	5,984	0.0%	24.2%
	20%	7,008	5,329	5,329	0.0%	24.0%
CONUS	25%	3,921	3,410	3,424	0.4%	14.5%
	30%	3,746	3,159	3,164	0.1%	15.7%
	35%	3,718	3,039	3,054	0.5%	22.3%

5. CONCLUSIONS

We proposed a very efficient algorithm for obtaining an MBB wavelength optimized provisioning that allows a seamless migration from a fragmented network provisioning to an optimized one.

Future work should consider investigating the best compromise between number of reroutings and bandwidth savings, as well as the triggering of the defragmentation events.

REFERENCES

[1] J. Wu, "A survey of WDM network reconfiguration: strategies and triggering method," *Computer Networks*, vol. 55, no. 11, pp. 2622–2645, 2011.

[2] H. Li and J. Wu, "Survey of WDM network reconfiguration: topology migrations and their impact on service disruptions," *Telecommunication Systems*, vol. 60, pp. 349–366, 2015.

[3] D. Coudert and J.-S. Sereni, "Characterization of graphs and digraphs with small process number," *Discrete Applied Mathematics*, vol. 159, pp. 1094–1109, July 2011.

[4] N. Jose and K. Somani, "Connection rerouting/network reconfiguration," in *Proceedings of IEEE/VDE Workshop on Design of Reliable Communication Networks - DRCN*, (Banff, Alberta, Canada), pp. 23 – 30, October 2003.

[5] C. Barnhart, E. Johnson, G. Nemhauser, M. Savelsbergh, and P. Vance, "Branch-and-price: Column generation for solving huge integer programs," *Operations Research*, vol. 46, pp. 316–329, May-June 1998.

[6] B. Jaumard, C. Meyer, and B. Thiongane, "On column generation formulations for the RWA problem," *Discrete Applied Mathematics*, vol. 157, pp. 1291–1308, March 2009.

[7] B. Jaumard and M. Daryalal, "Efficient spectrum utilization in large scale rwa problems," *IEEE/ACM Transactions on Networking*, vol. PP, pp. 1–16, March 2017.

[8] J. Simons, "CONUS network." Monarch Network Architects, `http://www.monarchna.com/topology.html`.

[9] B. Jaumard and M. Daryalal, "Efficient spectrum utilization in large scale RWA problems," *IEEE/ACM Transactions on Networking*, vol. 25, no. 2, pp. 1263–1278, 2017.