Routing and Wavelength Assignment of Scheduled Lightpath Demands

Josué Kuri, *Student Member, IEEE*, Nicolas Puech, Maurice Gagnaire, *Member, IEEE*, Emmanuel Dotaro, *Member, IEEE*, and Richard Douville

***Abstract*—In this paper, we present algorithms that compute the routing and wavelength assignment (RWA) for scheduled lightpath demands in a wavelength-switching mesh network without wavelength conversion functionality. Scheduled lightpath demands are connection demands for which the setup and teardown times are known in advance. We formulate separately the routing problem and the wavelength assignment problem as spatio-temporal combinatorial optimization problems. For the former, we propose a branch and bound algorithm for exact resolution and an alternative tabu search algorithm for approximate resolution. A generalized graph coloring approach is used to solve the wavelength assignment problem. We compared the proposed algorithms to an RWA algorithm that sequentially computes the route and wavelength assignment for the scheduled lightpath demands.**

***Index Terms*—Branch and bound (B&B), combinatorial optimization, graph coloring, routing and wavelength assignment (RWA), scheduled demands, tabu search (TS).**

I. Introduction

THE DEVELOPMENT in recent years of optical devices and systems for key networking functions (e.g., amplification, regeneration, multiplexing/demultiplexing, switching, etc.) has fostered the development of wavelength-division multiplexing (WDM) networks. Standardization bodies such as the International Telecommunications Union (ITU) have developed a set of standards that define the architecture of WDM optical transport networks (OTN) [1]. These networks are expected to be reconfigurable so that connections (commonly called *lightpaths*) can be set up, maintained, and torn down using either a management plane [2] or a control plane [3], [4]. The decision on whether to use a management plane or a control plane to instantiate the lightpaths depends on factors such as the characteristics of the traffic demands (e.g., if demands are static or scheduled, a management plane is enough; if demands arrive in an unexpected manner, a control plane would be necessary) and the approach of the operator to control its network (e.g., an historical telco carrier is probably more familiar with a management plane. Conversely, an Internet service provider (ISP) may be more familiar with a multiprotocol label switching (MPLS)-like control plane). Moreover, both a management plane and a control plane are likely to coexist in some optical transport networks.

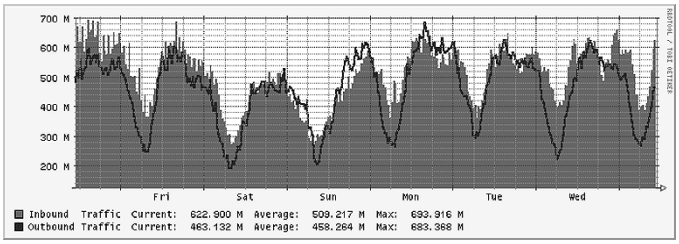

Fig. 1. Traffic on New York—Washington link of the Abilene backbone network from April 3, 2003 to April 10, 2003.

A service that could be provided by an OTN is the optical virtual private network (OVPN). An OVPN client company may request a set of static lightpaths to satisfy its minimal connectivity and capacity requirements from the provider. The client may also request some *scheduled* lightpaths to increase the capacity of its network at specific times on certain links, for example, between headquarters and production centers during office hours or between data centers during the night, when backup of databases is performed. Finally, the unexpected peaks of traffic demands in the client's network could be borne by using dynamically established lightpaths. This example of an OVPN shows that the services offered by the optical network operator can lead to three different types of lightpath demands: static, scheduled, and unexpected. We believe that when OTNs become a reality, at least for some years, most of the demands will be for static and scheduled lightpaths. The reason is that the traffic load in a transport network is fairly predictable because of its *periodic* nature. Fig. 1 gives an indication of this phenomenon. The figure shows the traffic on the New York–Washington link of the Abilene backbone network [5] for a typical week. A similar periodic pattern was observed on all the other links of the network in the same period. It can be argued that what is observed on a link is not necessarily an indication of the end-to-end traffic load profile and that the traffic load on a research network may be very different from the traffic load on a commercial network. However, the figure is an evidence of the link between the intensity of communication among humans using the network (greater during working hours), and the network traffic load.

In this paper, we deal with scheduled lightpath demands (SLDs). An SLD is a connection demand represented by a tuple $(s, d, n, \alpha, \omega)$, where s and d are the source and destination nodes of the demand, n is the number of requested lightpaths,

Manuscript received January 2, 2003; revised June 16, 2003. This paper was supported in part by Alcatel R & I, Marcoussis, France, under Grant CI 578. The work of J. Kuri was supported in part by CONACyT under Grant 122688.

J. Kuri, N. Puech, and M. Gagnaire are with the Department of Computer Science and Networks, École Nationale Supérieure des Télécommunications, GET Télécom Paris, Paris, France (e-mail: kuri@enst.fr; npuech@enst.fr; gagnaire @enst.fr).

E. Dotaro and R. Douville are with Alcatel R & I, Transport Network System and Architecture Unit, Marcoussis, France (e-mail: Emmanuel.Dotaro@ alcatel.fr; Richard.Douville@ alcatel.fr).

Digital Object Identifier 10.1109/JSAC.2003.816622

TABLE I
EXAMPLE OF THREE SCHEDULED LIGHTPATH DEMANDS (SLDS)

No.	s	d	n	α	ω
δ_1	2	8	2	08:00	14:40
δ_2	3	7	3	11:00	13:00
δ_3	5	6	2	17:00	19:30

and α, ω are the setup and teardown dates of the demand. Table I shows an example of an SLD set. Note that a demand for $n > 1$ lightpaths may exist if the requested rate is greater than the nominal rate of a lightpath (typically 2.5 Gb/s or 10 Gb/s).

The traffic model based on SLDs is different from the static and dynamic-random lightpath traffic models previously considered in the literature [6]. Static traffic means that all the demands are known in advance and do not change over time, whereas dynamic-random traffic means that the arrival and holding times of demands are random. The SLD traffic model is *deterministic* because all the demands are known in advance and is *dynamic* because it takes into account the evolution of the traffic load in the network over time.

It may happen that in a given set of SLDs, some of the demands are not simultaneous in time. For example, SLDs δ_1 and δ_3 in Table I are not simultaneous. Because of this time disjointness, a same network resource could be used to satisfy several demands at different times. In other words, the time disjointness of SLDs can be taken into account in order to minimize the number of network resources required to satisfy the set of SLDs. This is precisely the scope of the algorithms proposed in this paper: they take into account the time disjointness of SLDs in order to minimize the number of required network resources.

Section II describes in detail the problem under consideration. Section III presents spatio-temporal combinatorial optimization models used to represent separately the routing problem and the wavelength assignment (WA) problem. This section also presents the time correlation criterion used to characterize the sets of SLDs. Section IV describes the algorithms proposed to solve the resulting combinatorial optimization problems. In Section V, the algorithms are experimentally evaluated and compared with an alternative algorithm that sequentially computes a routing and wavelength assignment (RWA) for SLDs. Finally, Section VI presents some conclusions and future work directions.

II. DESCRIPTION OF THE PROBLEM

In the problem considered in this paper, we assume the following.

1) The network has an arbitrary connex topology.
2) The links are bidirectional.
3) There is one optical fiber per link direction.
4) Multiple channels can be simultaneously realized in an optical fiber using WDM.
5) There is no wavelength conversion functionality in the optical switches of the network nodes.

We also define the following terms.

- *WDM channel.* Refers to the use of a wavelength on a directed physical link (an optical fiber) connecting two nodes. For example, the use of wavelength $\lambda_{1550\ \mathrm{nm}}$ on the directed link going from node 4 to node 7 represents a WDM channel that will be denoted by $(\lambda_{1550\ \mathrm{nm}}, 4, 7)$.
- *Wavelength.* Refers to the use of a wavelength in *at least* one optical fiber in the network.

To illustrate the difference between a WDM channel and a wavelength, suppose that we have a network where we require WDM channels $(\lambda_{1550\ \mathrm{nm}}, 1, 2), (\lambda_{1580\ \mathrm{nm}}, 1, 2), (\lambda_{1550\ \mathrm{nm}}, 2, 3)$, and $(\lambda_{1580\ \mathrm{nm}}, 2, 3)$. In this case, four WDM channels and two wavelengths are required ($\lambda_{1550\ \mathrm{nm}}$ and $\lambda_{1580\ \mathrm{nm}}$).

In informal terms, the RWA problem under consideration is the following: given a network and a set of SLDs, find the routing in the network for the SLDs which meets an optimality criterion. Then, for this routing solution, find the assignment of wavelengths to lightpaths that minimizes the number of required wavelengths, while satisfying the wavelength continuity constraint (no wavelength conversion exists in the optical switches). Different optimality criteria may be considered in the routing subproblem, for instance, the minimization of the number of WDM channels or the minimization of the congestion (the number of lightpaths in the most loaded link). WDM channel minimization is particularly important in opaque WDM networks where a transmitter/receiver (Tx/Rx) is required for each channel (the Tx/Rx often represent the dominant cost of the network). On the other hand, congestion minimization is important when the number of wavelengths in the network is limited. In fact, the congestion is a lower bound on the number of required wavelengths. In this paper, we consider both versions of the routing subproblem. We denote the WDM channel minimization version of the routing subproblem by R_{ch} and the congestion minimization version by R_{cg}.

The time disjointness (if any) among demands is taken into account to minimize either the number of required WDM channels or the congestion. This point is illustrated in Fig. 2. The figure shows two possible routing solutions for the three SLDs of Table I. In solution 1, the (physically) shortest path is used for each SLD. The number of required WDM channels is 18 and the congestion is five. In solution 2, the second shortest path is used for SLD δ_1, while the shortest path is still used for SLDs δ_2 and δ_3. With this change, the four WDM channels used on fibers $(1, 5)$ and $(5, 6)$ by SLD δ_1 in the morning, are *reused* by SLD δ_3 in the afternoon. In this way, the number of required WDM channels is 14, instead of 18, and the congestion is three. The number of necessary wavelengths is five in the first solution and three in the second one.

It is worth to note that the algorithms proposed in this article to solve the RWA problem under consideration are intended to be used as part of an off-line centralized tool and not as an on-line distributed RWA algorithm.

III. MATHEMATICAL MODEL

We separately address the routing subproblem and the WA subproblem. Sections III-A and III-B describe the spatio-temporal model for each subproblem. Section III-C characterizes the instances of the RWA problem.

A. Routing

We first present the notations used in the model of the routing subproblem.

- $G = (V, E, w)$ is an edge-weighted undirected graph with vertex set $V = \{v_1, v_2, \ldots, v_N\}$, edge set $E = \{e_1, e_2, \ldots, e_L\}$, and weight function $w: E \rightarrow \mathbb{R}^+$. The graph represents a physical network. The set V corresponds to the network nodes, the set E to the bidirectional links interconnecting the nodes and the function w to the physical length or cost of the links (e.g., set by a network administrator).
- $N = |V|, \quad L = |E|$ are, respectively, the number of nodes and links in the network.
- $\Delta = \{\delta_1, \delta_2, \ldots, \delta_M\}$ is the set of M SLDs, where $\delta_i = (s_i, d_i, n_i, \alpha_i, \omega_i)$ is a tuple representing the SLD number i; $s_i, d_i \in V$ are the source and destination nodes of the demand, n_i is the number of requested lightpaths, and α_i and ω_i are the setup and teardown dates of the demand.
- (G, Δ) is a pair representing an instance of the routing subproblem.
- $P_{k,i}, \quad 1 \leq k \leq K_{\max}, \quad 1 \leq i \leq M$ represents the kth path in G from s_i to d_i. For the purposes of this paper, we compute the $K_{\max}$ physically shortest paths for each demand using the algorithm defined in [7]. However, the paths might be defined according to any other criterion. Moreover, the paths do not need to be necessarily neither link disjoint nor node disjoint. In Fig. 2, the physically shortest path corresponds to the first alternate path $(k = 1)$ and the second shortest path to the second alternate path $(k = 2)$.
- $\pi_{\rho,\Delta} = (P_{\rho_1,1} P_{\rho_2,2} \ldots P_{\rho_M,M}), \quad \rho \in \{1, \ldots, K_{\max}\}^M$ is called an *admissible routing solution* for Δ. ρ is an M-dimensional vector whose elements can take a value between 1 and $K_{\max}$. The element ρ_i indicates the path (among the $K_{\max}$) selected for the SLD δ_i. All the lightpaths of an SLD are routed through the same path (i.e., no bifurcated routing is used). An admissible routing solution is fully characterized by ρ.
- $\Pi_\Delta = \{\pi_{\rho,\Delta}, \rho \in \{1, \ldots, K_{\max}\}^M\}$ is the set of admissible routing solutions for Δ. There are $|\Pi_\Delta| = (K_{\max})^M$ admissible routing solutions in the set (assuming that the $K_{\max}$ paths exist for each SLD. Otherwise, $|\Pi_\Delta| \leq (K_{\max})^M$).
- $C_{\text{ch}}: \Pi_\Delta \rightarrow \mathbb{N}, \quad C_{\text{cg}}: \Pi_\Delta \rightarrow \mathbb{N}$ are the cost functions that compute, respectively, the number of WDM channels and the congestion in an admissible routing solution $\pi_{\rho,\Delta}$. In such a solution, a path is defined for each SLD $\delta \in \Delta$. Each lightpath of an SLD requires one WDM channel on every fiber spanned by its defined path. The same WDM channel can be used by several lightpaths, provided that they occur at different times. To formalize functions $C_{\text{ch}}: \Pi_\Delta \rightarrow \mathbb{N}$ and $C_{\text{cg}}: \Pi_\Delta \rightarrow \mathbb{N}$, we define the following additional notations.
- $\theta = (\theta_{ij})$ is a $\{0,1\}^{M \times M}$ upper triangular matrix; $\theta_{ij}, i \leq j$, indicates whether the SLDs δ_i and δ_j overlap in time $(\theta_{ij} = 1)$ or not $(\theta_{ij} = 0)$. By definition $\theta_{ii} = 1, 1 \leq i \leq M$, and $\theta_{ij} = 0$ for $i > j$. This matrix expresses the temporal relationship between the SLDs.
- $\beta = (\beta_{ij}) = \text{diag}(n_i)$ is the diagonal matrix, where $\beta_{ii} = n_i, 1 \leq i \leq M$, i.e., β_{ii} is the number of lightpaths required by the SLD δ_i.
- $\gamma^{\pi_{\rho,\Delta}} = \left(\gamma_{ij}^{\pi_{\rho,\Delta}}\right)$ is a $\{0,1\}^{L \times M}$ edge-path incidence matrix; $\gamma_{ij}^{\pi_{\rho,\Delta}}$ indicates whether edge $i \in E$ is part of path $P_{\rho_j,j}$ in routing solution $\pi_{\rho,\Delta} (\gamma_{ij}^{\pi_{\rho,\Delta}} = 1)$ or not $(\gamma_{ij}^{\pi_{\rho,\Delta}} = 0)$. For the sake of simplicity, we note γ instead of $\gamma^{\pi_{\rho,\Delta}}$. This matrix describes the physical routing of the SLDs for a given ρ.
- $\eta = \theta \cdot \beta \cdot \gamma^T = (\eta_{ij})$ is a $\mathbb{N}^{M \times L}$ matrix; η_{ij} indicates the number of time-overlapping lightpaths on link e_j between SLD δ_i and SLDs $\delta_k, \forall k > i$.

Thus, the C_{ch} cost function is defined by

$$C_{\text{ch}}(\pi_{\rho,\Delta}) = \sum_{j=1}^{L} \max_{1 \leq i \leq M} \eta_{ij} \tag{1}$$

and the C_{cg} function is defined by

$$C_{\text{cg}}(\pi_{\rho,\Delta}) = \max_{1 \leq j \leq N} \max_{1 \leq i \leq M} \eta_{ij}. \tag{2}$$

Let us consider the example of Fig. 2 to illustrate the definition of the cost functions. The ρ vector for solution 1 is $(1\ 1\ 1)$,

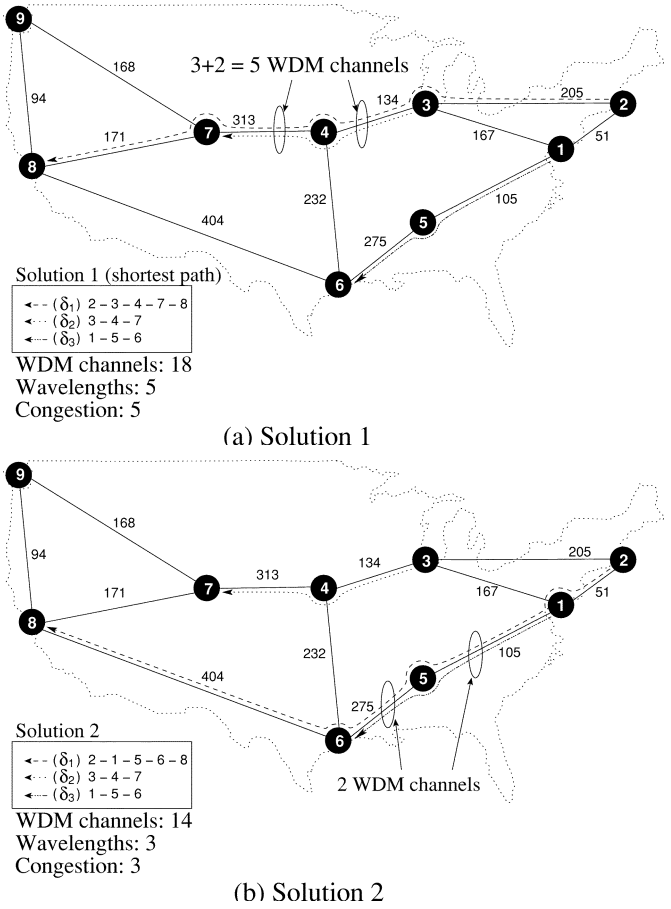

Fig. 2. Two alternative routing solutions for the SLDs of Table I.

i.e., the shortest path is used for each demand. Matrices θ and β are

$$\theta = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \beta = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 2 \end{pmatrix}.$$

The edge-path incidence matrix for ρ is[1]

$$\gamma = \begin{array}{c} \\ e_{23} \\ e_{34} \\ e_{15} \\ e_{56} \\ e_{47} \\ e_{78} \end{array} \begin{array}{c} \begin{array}{ccc} P_{1,1} & P_{1,2} & P_{1,3} \end{array} \\ \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \end{array}.$$

The corresponding η matrix is[1]

$$\eta = \theta \cdot \beta \cdot \gamma^T = \begin{array}{c} \\ \delta_1 \\ \delta_2 \\ \delta_3 \end{array} \begin{array}{c} \begin{array}{cccccc} e_{23} & e_{34} & e_{15} & e_{56} & e_{47} & e_{78} \end{array} \\ \begin{pmatrix} 2 & 5 & 0 & 0 & 5 & 2 \\ 0 & 3 & 0 & 0 & 3 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 \end{pmatrix} \end{array}.$$

Finally, the costs of this solution with C_{ch} and C_{cg} are

$$\begin{aligned} C_{\text{ch}}(\pi_{\rho,\Delta}) &= \sum_{j=1}^{L} \max_{1 \le i \le M} \eta_{ij} \\ &= 2 + 5 + 2 + 2 + 5 + 2 = 18 \end{aligned}$$

and

$$C_{\text{cg}}(\pi_{\rho,\Delta}) = \max_{1 \le j \le N} \max_{1 \le i \le M} \eta_{ij} = 5.$$

The R_{ch} version of the routing subproblem is formulated as the following combinatorial optimization problem

minimize

$$C_{\text{ch}}(\pi_{\rho,\Delta}) \tag{3}$$

subject to

$$\pi_{\rho,\Delta} \in \Pi_{\Delta} \tag{4}$$

that is, we are interested in finding an admissible routing solution $\pi_{\rho,\Delta}$ that minimizes the number of required WDM channels for the set of demands Δ.

The R_{cg} version of the routing subproblem is formulated as

minimize

$$C_{\text{cg}}(\pi_{\rho,\Delta}) \tag{5}$$

subject to

$$\pi_{\rho,\Delta} \in \Pi_{\Delta} \tag{6}$$

that is, we search an admissible routing solution that minimizes the congestion.

B. Wavelength Assignment

The solution to the routing subproblem is an admissible routing solution $\pi_{\rho,\Delta}$ that defines a route on the physical network for each SLD in Δ; it constitutes an instance of the WA subproblem.

[1]Only the edges used by at least one SLD in the solution are represented in this matrix.

The solution to the WA subproblem defines the wavelength assigned to each lightpath of the SLDs in Δ for the chosen $\pi_{\rho,\Delta}$ so that the number of used wavelengths is minimal and 1) each lightpath uses the same wavelength on all the fibers spanned by its route (wavelength continuity constraint) and 2) any two time-simultaneous lightpaths whose routes have at least one fiber in common use different wavelengths.

The WA problem in networks with one fiber per link direction assuming the wavelength continuity constraint and the static traffic model reduces to a well-known graph vertex coloring problem [6]. We generalize this model for the WA problem using the SLD traffic model [8]. The following notations are used:

$$\psi = \gamma^T \cdot \gamma = (\psi_{ij})$$

is an $\mathbb{N}^{M \times M}$ matrix; ψ_{ij} is the number of edges on routing solution $\pi_{\rho,\Delta}$, where SLD δ_i overlaps with SLD δ_j (remember that γ is a simplified notation for $\gamma^{\pi_{\rho,\Delta}}$); μ_i is the set of lightpaths belonging to the SLD δ_i; $\mu_i^j, 1 \le j \le n_i$, denotes the jth lightpath of SLD δ_i

$$G_c = (V_c, E_c)$$

is a graph, where $V_c = \cup_{i=1}^{M} \{\mu_i\}$ and $E_c = \{(\mu_p^i, \mu_q^j) \,|\, \mu_p^i \neq \mu_q^j, \psi_{pq} > 0, \theta_{pq} = 1\}$. This graph is called the conflict graph associated to the considered WA problem instance.

Once the graph vertex coloring problem is solved, the color associated to each vertex in G_c represents the wavelength to be assigned to the lightpath represented by this vertex.

C. Characterization of Problem Instances

The time correlation among the SLDs in Δ has a significant incidence on the cost of a solution of an instance of the RWA problem under consideration. In fact, a set of SLDs with significant time disjointness should ease the reuse of WDM channels and, as a consequence, lead to a smaller number of required channels. We characterize the sets Δ according to the time correlation among demands. In order to define this correlation, let

$$\varepsilon = \left(\bigcup_{i=1}^{M} \{\alpha_i\} \right) \cup \left(\bigcup_{i=1}^{M} \{\omega_i\} \right) \tag{7}$$

be an ordered set of $T = |\varepsilon|$ values $\varepsilon_1 < \varepsilon_2 < \cdots < \varepsilon_T$ (of course, $T \le 2M$ since some α_i and some ω_i might be the same) and

$$B_i = \{j \in \{1, \ldots, M\} \,|\, [\varepsilon_i, \varepsilon_{i+1}] \subseteq [\alpha_j, \omega_j]\} \tag{8}$$

is the set of SLD indexes j such that the SLD is active (at least) over time period $[\varepsilon_i, \varepsilon_{i+1}]$.

The time correlation of a SLD set is given by the formula

$$\tau(\Delta) = \frac{\sum_{i=1}^{T-1} \sum_{j \in B_i : |B_i| > 1} n_j (\varepsilon_{i+1} - \varepsilon_i)}{\sum_{i=1}^{M} n_i \cdot (\omega_i - \alpha_i)}. \tag{9}$$

Note that only $|B_i| > 1$ index sets must be considered in the formula, since they correspond to time overlapping of at least two SLDs.

A value $0 \le \tau(\Delta) \le 1$ close to 0 means weak time correlation and a value close to 1, strong time correlation.

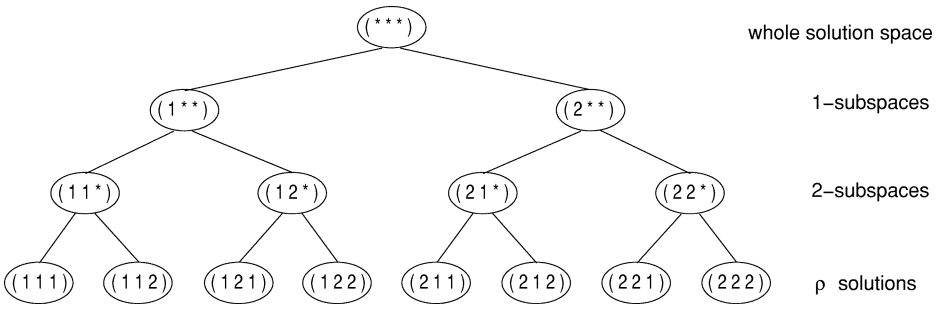

Fig. 3. Enumeration tree of solutions to a problem with $M = 3$ and $K_{\max} = 2$. Nodes correspond to m-subspaces and leaves to solutions ρ.

IV. Algorithms

In this section, we propose algorithms to find solutions to the the routing and the wavelength subproblems formulated in Section III.

A. Routing

1) Branch and Bound: We first describe a branch and bound (B&B) algorithm that computes optimal solutions[2] to instances of the $R_{\rm ch}$ or the $R_{\rm cg}$ version of the routing subproblem stated in Section III-A. We only describe the B&B algorithm that solves $R_{\rm ch}$. The B&B algorithm that solves $R_{\rm cg}$ is obtained by replacing $B_{\rm ch}$ and $C_{\rm ch}$ (see below) by $B_{\rm cg}$ and $C_{\rm cg}$.

The solution of a problem with a B&B algorithm can be described as a search through a dynamically built enumeration tree. As its name suggests, an enumeration tree is a structure used to enumerate all the feasible solutions of the problem. The root node of the tree corresponds to the whole solution space, the remaining nodes are solution subspaces and the leaves are feasible solutions to the problem. The aim of the algorithm is to find a feasible solution (i.e., a leaf) for which the cost is minimal[3] without *explicitly* enumerating all the solutions. A description of B&B algorithms' principles can be found in [9].

We first define some terms. Recall that an admissible routing solution $\pi_{\rho,\Delta}$ can be fully characterized by the vector $\rho \in \{1 \ldots K_{\max}\}^M$. Indeed, the value $\rho_i (1 \leq i \leq M, 1 \leq \rho_i \leq K_{\max})$ indicates that the ith SLD is routed through the ρ_ith path between nodes s_i and d_i. Hereafter, we refer to a vector ρ as a solution. Furthermore, we define an *m-solution-subspace* ρ^m, or simply an *m-subspace*, as a subspace of solutions wherein only the paths for the first $m, (m < M)$ SLDs are defined, whereas the paths of the remaining $\delta_i (m < i \leq M)$ SLDs are undefined. The following example illustrates this definition. Consider a set $\Delta = \{\delta_1, \delta_2, \delta_3\}$ of $M = 3$ SLDs and $K_{\max} = 2$ possible paths for each demand. The vectors $(1\ 1\ 1), (1\ 1\ 2), (1\ 2\ 1), (1\ 2\ 2), \ldots, (2\ 2\ 2)$ represent the set Π_Δ of admissible routing solutions $\pi_{\rho,\Delta}$ for Δ. Vectors $(1\ *\ *)$ and $(2\ *\ *)$ are 1-subspaces, since only the path of the first SLD is defined (undefined paths are indicated with an asterisk). Solutions $(1\ 1\ 1), (1\ 1\ 2), (1\ 2\ 1)$, and $(1\ 2\ 2)$ belong to the $(1\ *\ *)$ 1-subspace. We denote this type of relationship by $\rho \in \rho^m$. We can build an *enumeration tree* of all the solutions to the routing subproblem in the following manner: the root node corresponds to the 0-subspace (i.e., the whole solution space). Child nodes are derived from the root node, each child corresponding to one of the $K_{\max}$ 1-subspaces. $K_{\max}$ child nodes are in turn derived from each 1-subspace. This second level is the enumeration of all the possible 2-subspaces. Subsequent levels of child nodes are derived until the ρ solutions are enumerated (i.e., we have reached the leaves of the tree). We refer to the derivation of a node as a *branching* operation. Fig. 3 illustrates the enumeration tree for the above example. The root node corresponds to the 0-subspace $(*\ *\ *)$. The level 1 and level 2 nodes are the enumeration of the 1-subspaces and the 2-subspaces, respectively. Finally, the leaves of the tree correspond to the ρ solutions.

Let $\Pi_{m,\Delta}$ be the set of m-subspaces (the nodes of the mth level in the enumeration tree). We define $B_{\rm ch}: \Pi_{m,\Delta} \rightarrow \mathbb{N}$ as a *lower bound function* that determines the total number of WDM channels (as in the $C_{\rm ch}$ function) required by the m SLDs for which the routing is defined in a $\rho^m \in \Pi_{m,\Delta}$ m-subspace. The function is a lower bound on the number of channels required by the solutions belonging to a m-subspace, i.e., $C_{\rm ch}(\pi_{\rho,\Delta}) \geq B_{\rm ch}(\pi_{\rho^m,\Delta}), \forall \rho \in \rho^m$. Furthermore, this lower bound increases as a function of m, i.e., $B_{\rm ch}(\pi_{\rho^j,\Delta}) \geq B_{\rm ch}(\pi_{\rho^i,\Delta})$ for $j > i$ and $\rho^j \in \rho^i$.

An initial solution must be provided.[4] This solution is considered at the beginning of the algorithm as the current best or *incumbent solution* ρ_i.

The dynamically developed tree contains at the beginning the root node, which is assigned to an initially empty pool of live nodes. In each *iteration* of the B&B algorithm, a node is selected from the pool using a selection strategy. Branching is then performed on the selected node to create its $K_{\max}$ children. In the case that the created nodes are leaves of the tree (i.e., admissible routing solutions), their cost are compared with $C_{\rm ch}(\pi_{\rho_i,\Delta})$, i.e., to the cost of the incumbent solution. If the cost of a created solution is smaller than $C_{\rm ch}(\pi_{\rho_i,\Delta})$, this created solution becomes ρ_i. Otherwise, if the nodes created by the branching are not leaves, the $B_{\rm ch}$ function is used to compute the bound associated to each of these nodes. The nodes with a bound $B_{\rm ch}(\pi_{\rho^m,\Delta}) < C_{\rm ch}(\pi_{\rho_i,\Delta})$ are stored in the pool of live nodes. Conversely, the nodes with a bound $B_{\rm ch}(\pi_{\rho^m,\Delta}) \geq C_{\rm ch}(\pi_{\rho_i,\Delta})$ are discarded since, at this point, we know that all the solutions derived from these nodes (subspaces) have a cost higher than or

[2]Solutions that satisfy the optimality condition (3) or (5) of $R_{\rm ch}$ and $R_{\rm cg}$, respectively.

[3]Maximization problems can be solved with B&B algorithms with small changes.

[4]The $k = 1$ path for all the SLDs is a candidate solution. However, as indicated later, the use of meta-heuristics to obtain an initial solution may a better alternative for the problem under consideration.

equal to $C_{\rm ch}(\pi_{\rho_i,\Delta})$. The operation of discarding a node is referred to as fathoming or *bounding*. The algorithm ends when the pool of live nodes is empty. The final solution is ρ_i.

By bounding a node, the associated solution subspace is eliminated from the search. It is, thus, important to bound nodes as close as possible to the root node since, as such, large portions of the complete solution space are not explored. Two factors with direct incidence on the bounding efficiency are the selection of a good initial solution and the strategy used to select nodes from the pool of live nodes. A good initial solution can be obtained, for instance, with the tabu search (TS) meta-heuristic algorithm described below. For the selection of nodes from the pool of live nodes, several strategies such as best first search (BeFS), breadth first search (BFS), or depth first search (DFS) exist [9], [10].

The time complexity of the B&B algorithm is $O(K_{\max}^M)$. Indeed, if no bounding is performed at all, the whole set of admissible routing solutions Π_Δ must be explicitly explored. This exponential complexity precludes the use of the algorithm when solving problem instances with a large number of demands (M).

2) Tabu Search (TS): The major drawback of the B&B algorithm is its exponential time complexity. An alternative approach to solve combinatorial optimization problems in general is the use of meta-heuristic algorithms. These algorithms explore only a fraction of the solution space but do not guarantee that the obtained solutions are optimal (with respect to the optimization model). However, it has been observed in practice that they provide solutions of "good quality" (i.e., solutions with a cost close to the optimal one). Since only a fraction of the solution space is explored, it is possible to tackle problem instances of large size.

We propose a TS meta-heuristic algorithm to find approximate solutions to the $\mathrm{R}_{\rm ch}$ and $\mathrm{R}_{\rm cg}$ versions of the routing subproblem. As for the B&B algorithm, we only describe the TS algorithm that solves $\mathrm{R}_{\rm ch}$. The TS algorithm that solves $\mathrm{R}_{\rm cg}$ is obtained by replacing $C_{\rm ch}$ by $C_{\rm cg}$.

TS is an iterative meta-heuristic algorithm used for combinatorial optimization problems. The algorithm explores the space of solutions until either a number of iterations is reached or a specific cost criterion is satisfied. The exploration starts with an *initial solution* computed by another algorithm (e.g., a random solution if nothing better can be prepared). This solution is, at the beginning of the algorithm, the current solution. At each iteration, TS computes a set of solutions (called a *neighborhood*) derived from the current solution through *perturbations* applied to this solution. The solution in the neighborhood with the best cost is selected as the new current solution. In order to prevent the algorithm from cycling along the same series of current solutions, a *tabu list* is maintained. The list contains a number of last visited solutions that cannot be chosen as long as they belong to the list. The solution selected at each iteration as the current one is inserted in the list and stays there for a given number of iterations. This mechanism allows the algorithm to escape from the local minima encountered during the search. A detailed description of TS can be found in [11].

Though there are no mathematical proofs of the convergence of TS to a global optimum, the algorithm is widely used in practice since it usually provides solutions with a cost "close" to the optimal one. Furthermore, it allows combinatorial optimization problems of realistic size to be tackled.

In order to design a TS algorithm, three problem-specific elements must be defined, namely: 1) an initial solution; 2) a cost function to evaluate the solutions generated by the algorithm; and 3) a perturbation procedure to generate a neighborhood of solutions from a current one. In our proposed TS algorithm [12], the initial solution is the ρ vector whose components are all equal to one. Function $C_{\rm ch}$ is used as the cost function. To generate a new solution from a current solution $\pi_{\rho,\Delta}$, we use the following perturbation procedure.

1) Generate a random number i, uniformly distributed in the interval $[1,\ldots,M]$.
2) Generate a random number j, uniformly distributed among the elements of the set $\{1,\ldots,K_{\max}\}\backslash\{\rho_i\}$.
3) Replace ρ_i by j.

This procedure creates a new solution from a current one $\pi_{\rho,\Delta}$ by randomly selecting a demand from Δ and randomly modifying the selected route for this demand. For example, a solution whose ρ vector is $(2,1,1,3)$ can be "perturbed" according to this procedure and leads to the solution $(2,3,1,3)$.

B. Wavelength Assignment

In order to solve the graph vertex coloring problem on the conflict graph G_c defined in Section III-B, we used the greedy algorithm described in [13], which computes approximate solutions. We selected the algorithm for its simplicity.

C. An (alternative) Sequential RWA Algorithm

The algorithms proposed in this paper are based on a combinatorial optimization approach wherein all the demands are simultaneously considered, so that the time disjointness among them is globally exploited. An alternative to solve the same RWA problem is to use a sequential algorithm that computes the routing and wavelength assignment sequentially, that is, demand by demand. In a sequential algorithm, the path and wavelength(s) assigned to a particular demand depend on the state of the network at the moment the routing and wavelength assignment decision is taken. Moreover, what may seem a good decision for a given demand may have a negative impact on the global cost. However, sequential algorithms are in general computationally simpler than combinatorial optimization based algorithms. We present a simple sequential RWA algorithm called sRWA that we use in Section V as a reference to evaluate the performance of the algorithms of Sections IV-A and IV-B.

The sRWA algorithm works in two phases. In the first one, the SLDs in Δ are sorted by descending order of weights $w_i = n_i \cdot h_{1,i}$, where n_i is the number of lightpaths of SLD δ_i and $h_{1,i}$ is the number of hops in path $P_{1,i}$. The reason for processing the SLDs with the longest path (in terms of hops) and the largest number of lightpaths first is explained in [14]. Intuitively, demands with long paths and large size are harder to establish than demands with short paths and small size because more unallocated wavelengths must be found on more links. In the second phase, the path and the wavelength(s) are computed for each demand, one by one, according to the order defined in the first step. We use fixed-alternate routing and first fit (FF) wavelength assignment [6]. In fixed-alternate routing, k alternate paths are

computed in advance for each source/destination pair in the network (in the sRWA algorithm proposed here, the paths are not necessarily link-disjoint). When a demand from s to d is processed, the k paths between s and d are ranked according to the value computed by the FF algorithm for each path (see below) and the path with the lowest value is selected. If more than one path have the same lowest value, the physically shortest path is selected. In FF wavelength assignment, the wavelengths in an optical fiber are sorted by ascending order of wavelength (e.g., $\lambda_{1550.0\text{ nm}} < \lambda_{1550.1\text{ nm}} < \lambda_{1550.2\text{ nm}} \ldots$). When searching for an available wavelength in the fibers along a path, a lower-numbered wavelength is considered before a higher numbered one. A wavelength is available in a path if it has not been previously assigned to a SLD that is time overlapping with the SLD currently being processed.

V. Experimental Evaluation

In this section, we experimentally evaluate the algorithms proposed in the previous sections. We use the following acronyms to refer to these algorithms:

- $\mathrm{B\&B_{ch}}$ The branch and bound routing algorithm for WDM channel minimization (Section IV-A1).
- $\mathrm{TS_{ch}}$ The TS routing algorithm for WDM channel minimization (Section IV-A2).
- $\mathrm{TS_{cg}}$ The TS routing algorithm for congestion minimization (Section IV-A2).
- GGC The greedy vertex coloring algorithm (Section IV-B).
- sRWA The alternative sequential RWA algorithm (Section IV-C).

We implemented the B&B algorithm using a library of parallel search algorithms called ZRAM [15]. The use of the library allowed individual problem instances to be solved on a cluster of 40 Sun Ultra-SPARC 5 computers with 128 MB of RAM each, running the Solaris 5.8 operating system. The library was particularly useful when dealing with problem instances with more than 10^{18} potential solutions ($M = 30, K_{\max} = 4$).

The TS algorithm was implemented in C. We used the pseudorandom number generator proposed in [16] to implement the perturbation procedure. For the GGC algorithm, we used an implementation in C provided by the authors. The sRWA algorithm was implemented in Perl. These three programs were executed on a Sun Fire 280 R computer with 1 GB of RAM running the Solaris 5.8 operating system.

Fig. 4 shows an undirected graph G representing the physical network considered in all the experiments of this section. The network corresponds to a hypothetical U.S. backbone network of 29 nodes and 44 links. A weight, representing the physical length, is associated to each edge of the graph (weights not shown in the figure).

We considered two classes of sets Δ: a class S_w of sets Δ with a weak time correlation $\tau(\Delta) \approx 0.01$ and a class S_s of sets Δ with a strong time correlation $\tau(\Delta) \approx 0.8$. The SLDs in a set Δ were randomly generated, being constrained only by the time correlation defined for the class the set Δ belongs to.

A. Quality of (approximate) TS Routing Solutions

Since the approximate solutions computed with TS are not guaranteed to be optimal, we first need to assess the quality of these approximate solutions with respect to the optimal solutions computed with B&B. We use the $\mathrm{B\&B_{ch}}$ and $\mathrm{TS_{ch}}$ algorithms for this comparison (WDM channel minimization). For a given problem instance (G, Δ), we want to know the quality loss $(C_{\mathrm{ch}}(\pi_{\rho_{\mathrm{TS}},\Delta}) - C_{\mathrm{ch}}(\pi_{\rho_{\mathrm{BB}},\Delta}))/(C_{\mathrm{ch}}(\pi_{\rho_{\mathrm{BB}},\Delta})) \cdot 100$, of the cost $C_{\mathrm{ch}}(\pi_{\rho_{\mathrm{TS}},\Delta})$ computed with $\mathrm{TS_{ch}}$ with respect to the cost $C_{\mathrm{ch}}(\pi_{\rho_{\mathrm{BB}},\Delta})$ computed with $\mathrm{B\&B_{ch}}$.

We generated 60 sets Δ of each class (S_w and S_s) with $M = 30$ SLDs each. For the $\mathrm{TS_{ch}}$ algorithm, we used the following initial settings: 3000 iterations, a neighborhood of 100 solutions, and a tabu list of 114 elements. Table II shows the average, maximum, and standard deviation, over the 60 sets Δ, of the quality loss for each class using these initial settings. Though the average values are small (3.75% in the worst case), the maximum values were as large as 16.22% in some cases.

By carefully analyzing the execution of TS in these cases, we noted that the algorithm was frequently trapped in some local minima. Therefore, we decided to increase the size of the neighborhood up to 200 and the size of the tabu list up to 20 times the neighborhood size. Additionally, we implemented a *diversification* function. This function works by applying multiple perturbations to the current solution if no cost improvement has been obtained after a large number of iterations (see [11]); hence, the search effort is shifted from a region of the solution space to another. Table III shows the results obtained after the implementation of these modifications. Note that the average and maximum values decrease as does the standard deviation.

This experiment illustrates the importance of designing both exact and approximate algorithms for a same optimization problem. By comparing approximate and exact solutions on small-size problem instances, it is possible to determine the quality of the approximate solutions and adjust the approximate

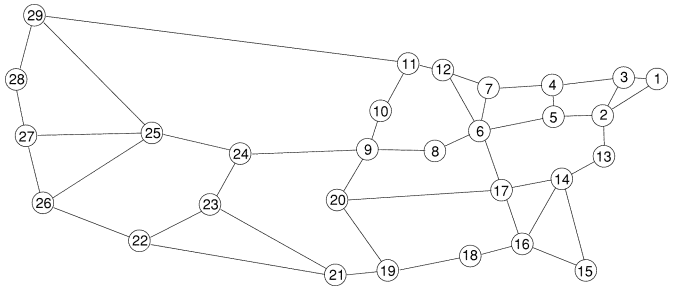

Fig. 4. Graph G representing the physical network.

TABLE II
Quality Loss for Different Values of $K_{\max}$

Class	Quality loss (%) (Avg / Max / Std Dev)		
	$K_{\max}$		
	2	3	4
S_w	1.20 / 8.76 / 1.96	2.13 / 15.12 / 2.73	3.75 / 16.22 / 3.96
S_s	0.61 / 7.46 / 1.38	1.50 / 9.83 / 1.94	1.70 / 10.08 / 2.26

TABLE III
Quality Loss for Different Values of $K_{\max}$ (Modified $\mathrm{TS_{ch}}$)

Class	Quality loss (%) (Avg / Max / Std Dev)		
	$K_{\max}$		
	2	3	4
S_w	0.70 / 3.98 / 0.98	0.59 / 8.14 / 1.61	1.13 / 12.88 / 2.02
S_s	0.52 / 4.90 / 1.02	0.35 / 6.58 / 1.04	0.56 / 10.08 / 1.55

TABLE IV
COMPARISON OF THE AVERAGE NUMBER OF WDM CHANNELS (ROUTING METRIC) COMPUTED WITH sRWA AND $\mathrm{TS_{ch}}$

Class	sRWA	TS_{ch}			Gain of TS_{ch} *vs.* sRWA		
		$K_{\max}=2$	$K_{\max}=3$	$K_{\max}=4$	$K_{\max}=2$	$K_{\max}=3$	$K_{\max}=4$
S_w	606.34	518.11	497.51	485.93	14.55 %	17.94 %	19.85 %
S_s	678.17	628.53	597.98	581.63	7.31 %	11.82 %	14.23 %

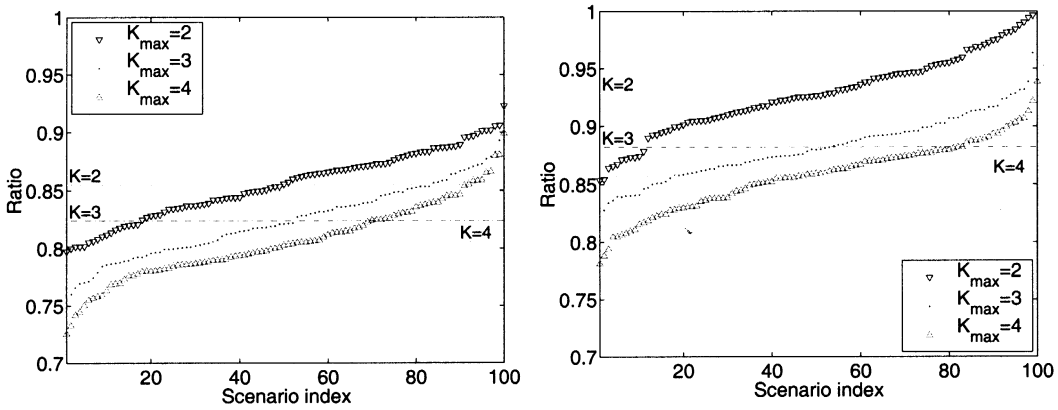

Fig. 5. Ratio of the number of WDM channels computed by $\mathrm{TS_{ch}}$ to the number of WDM channels computed by sRWA (small values are better).

TABLE V
COMPARISON OF THE AVERAGE NUMBER OF WAVELENGTHS (WAVELENGTH ASSIGNMENT METRIC) COMPUTED WITH sRWA AND $\mathrm{TS_{cg}}/\mathrm{GGC}$

Class	sRWA	TS_{cg}/GGC			Gain of TS_{cg}/GGC *vs.* sRWA		
		$K_{\max}=2$	$K_{\max}=3$	$K_{\max}=4$	$K_{\max}=2$	$K_{\max}=3$	$K_{\max}=4$
S_w	12.52	9.49	9.49	9.41	24.20 %	24.20 %	24.84 %
S_s	16.08	11.65	11.16	11.23	27.54 %	30.59 %	30.16 %

algorithm parameters to get solutions closer to the optimal ones.

B. Gain With Respect to Sequential Algorithm

We evaluated the gain obtained with the combinatorial optimization based algorithms when compared with the (sequential) sRWA algorithm. We generated 100 sets Δ of each class (S_w and S_s) with $M = 500$ SLDs each.

We first evaluate the gain in terms of WDM channels (routing metric) obtained by $\mathrm{TS_{ch}}$ with respect to sRWA. For $\mathrm{TS_{ch}}$, we used the settings described in the previous subsection. Furthermore, we considered $k = 10$ paths for the sRWA algorithm.

Table IV shows the average number of WDM channels computed by sRWA and $\mathrm{TS_{ch}}$ using different values of $K_{\max}$ for classes S_w (weak time correlation) and S_s (strong time correlation). The table also shows the gain of $\mathrm{TS_{ch}}$ with respect to sRWA. The gain increases with $K_{\max}$ (regardless the class) because the size of the solution space increases with $K_{\max}$; as a consequence, $\mathrm{TS_{ch}}$ has a larger solution space to explore. Finally, the gain is higher for class S_w than for class S_s because the greater potential of channel reuse in S_w is better exploited by $\mathrm{TS_{ch}}$ than by sRWA.

Fig. 5 shows the ratio of the number of WDM channels computed by $\mathrm{TS_{ch}}$ ($K_{\max} = 2, 3, 4$) to the number of WDM channels computed by sRWA for the 100 sets Δ of each class. Clearly, the smaller the ratio value, the greater the gain.

We then evaluated the gain in terms of wavelengths (wavelength assignment metric) obtained with the combination of algorithms $\mathrm{TS_{cg}}/\mathrm{GGC}$ with respect to sRWA. Note that we use $\mathrm{TS_{cg}}$ instead of $\mathrm{TS_{ch}}$ because the former aims at minimizing the congestion, which sets a lower bound on the number of required wavelengths.

Table V shows, for each class, the average number of wavelengths (wavelength assignment metric) computed by sRWA, and by GGC on the routing solutions obtained with $\mathrm{TS_{cg}}$ using different values of $K_{\max}$. The average number of wavelengths is smaller in the weak time correlation class S_w (regardless the algorithm). Indeed, the greater time disjointness of SLDs in S_w allows for a better "time reuse" of wavelengths. Note, however, that the gain in this class is smaller than in S_s. A point worth noting is that in all cases we required on average less than 16.08 wavelengths to satisfy $M = 500$ SLDs, that is, the number of required wavelengths was significantly smaller than the number of demands even with the wavelength continuity constraint. This is a result of the time and spatial reuse of wavelengths.

The combined execution time of $\mathrm{TS_{cg}}$ and GGC was of 6480 s in average (200 problem instances). The execution time of sRWA was of 20 s on average. The gain, both in WDM channels and wavelengths makes the execution time increase worthwhile. Furthermore, the algorithms are not expected to be run in a real-time environment (as would be the case for sRWA), but as batch processes at time intervals of several hours (remember that the SLDs correspond to planned demands).

VI. Conclusion

In this paper, we presented algorithms for the routing and wavelength assignment of SLDs in a wavelength-switching mesh network without wavelength conversion functionality. The routing problem and WA problem were separately formulated as spatio-temporal combinatorial optimization problems. A B&B algorithm and a TS algorithm were proposed to compute exact and approximate solutions, respectively, to the resulting optimization problems. For the WA problem, we used a greedy graph vertex coloring algorithm previously proposed in the literature.

We assessed the quality of the approximate solutions computed by the TS algorithm by comparing them to exact solutions computed with the B&B algorithm on problem instances of small size. We were able to obtain approximate solutions very close to the optimal ones by modifying the TS parameters. We also evaluated the gain in terms of WDM channels (routing metric) and wavelengths (wavelength assignment metric) obtained with the combination of the TS and GGC algorithms when compared with a sequential routing and wavelength assignment algorithm. The time correlation among SLDs in a set Δ has a significant incidence on the average gain in WDM channels. Remarkably, the number of required wavelengths was significantly smaller than the number of demands because of the time and space wavelength reuse.

The inherent determinism of the SLD traffic model in general, and the time and space disjointness of demands in particular, can be interestingly exploited in contexts different to the one considered in this paper. For example, the combined time and space disjointness of demands can be used in the minimization of the spare capacity required to protect SLDs [17]. A different application of the SLD traffic model can be the design of sequential algorithms. For example, the RWA algorithms presented in this paper can be used in combination with simulations in order to design efficient sequential RWA algorithms for dynamic-random traffic demands. For this, we employ simulations to evaluate the sequential RWA algorithm (as is typically done). During the simulations, we store a trace of the demands, dynamically generated according to a stochastic model. We then interpret the obtained trace as a set of SLDs and use it as input to the SLD RWA algorithms. In this manner, we can compare the properties of the routes computed by the sequential algorithm against the routes obtained with the SLD RWA algorithms in order to identify particular patterns. If successfully abstracted, these patterns can be used to improve the sequential algorithm. For this approach to work, we must be able to solve with the SLD RWA algorithms problem instances with a very large number of demands (in the order of tens of millions).

References

[1] *Architecture of Optical Transport Network*, ITU-T Recommendation G.872, Nov. 2001.

[2] *Management of the Transport Network*, ITU-T Recommendation G.852, Mar. 1999.

[3] *Requirements for the Automatic Switched Transport Network (ASTN)*, ITU-T Recommendation G.807, July 2001.

[4] Generalized Multi-Protocol Label Switching (GMPLS) Architecture, E. Mannie, Ed., IETF ccamp group, Mar. 2002. Work in progress.

[5] Advanced Networking for Research and Education [Online]. Available: http://abilene.internet2.edu/

[6] H. Zang, J. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *Opt. Networks Mag.*, vol. 1, no. 1, pp. 47–60, 2000.

[7] D. Eppstein, "Finding the k shortest paths," *SIAM J. Computing*, vol. 28, no. 2, pp. 652–673, 1998.

[8] J. Kuri, N. Puech, M. Gagnaire, and E. Dotaro, "Routing and wavelength assignment of scheduled lightpath demands in a WDM optical transport network," in *Proc. ICOCN 2002*, Singapore, Nov. 2002, pp. 270–273.

[9] J. Clausen, "Branch and Bound Algorithms—Principles and Examples," Dept. Comput. Sci., Univ. Copenhagen, [Online]. Available: http://www.imm.dtu.dk/~jha/.

[10] J. Clausen and M. Perregaard, "On the best search strategy in parallel branch-and-bound—Best-first-search vs. lazy depth-first-search," Annals of Operations Research, No. 90, pp. 1–17, 1999.

[11] F. Glover and M. Laguna, *Tabu Search*. Boston, MA: Kluwer-Academic, 1997.

[12] J. Kuri, N. Puech, M. Gagnaire, and E. Dotaro, "Routing foreseeable lightpath demands using a tabu search meta-heuristic," in *Proc. GLOBECOM 2002*, Taipei, Taiwan, Nov. 2002, pp. 2803–2807.

[13] D. Kirovski and M. Potkonjak, "Efficient coloring of a large spectrum of graphs," in *Proc. 35th Conf. Design Automation*, San Francisco, CA, June 1998, pp. 427–432.

[14] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: An approach to high bandwith optical WAN's," *IEEE Trans. Commun.*, vol. 40, pp. 1171–1182, July 1992.

[15] A. Marzetta, "ZRAM: A Library of Parallel Search Algorithms and Its Use in Enumeration and Combinatorial Optimization," Ph.D. dissertation, Comput. Sci. Dept., Swiss Fed. Inst. Technol., Zurich, Switzerland, 1998.

[16] S. Kirkpatrick and E. Stoll, "A very fast shift-register sequence random number generator," *J. Comput. Phys.*, vol. 40, 1981.

[17] J. Kuri, N. Puech, and M. Gagnaire, "Diverse routing of scheduled lightpath demands in an optical transport network," 4th Workshop on Design of Reliable Computer Networks, Oct. 2003, submitted for publication.

Josué Kuri (S'99) received the B.S. degree (honors) in computer science from Instituto Tecnologico y de Estudios Superiores de Monterrey (ITESM), Monterrey, Mexico, in 1996, the M.Sc. degree in telecommunications from École Superiéure d'Electricité, Rennes, France, in 1998, and the Ph.D. degree in telecommunications from École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 2003.

From 1996 to 1998, he worked at InfoSel SA, Monterrey, Mexico, in the development of a real-time financial information system. His research areas of interest are optimization problems in transport networks, design of telecommunications equipment, and standardization of telecommunications systems.

Nicolas Puech received the engineer degree in telecommunications from École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1987. He received the M.S. degrees in computer science and mathematics from the University Paris 6, France, in 1987 and 1989, respectively, and the Ph.D. degree in computer science from the University Paris 11, France, in 1992.

From 1998 to 2002, he was an Associate Professor at the University Paris 5, France. He joined ENST in 2002 as an Associate Professor and is currently with the Access and Mobility in Networks Group, Computer Science and Networks Department. His research interests include network planning and modeling for WDM photonic networks, as well as numerical optimization and mathematical modeling.

Maurice Gagnaire (M'88) graduated from the Institut National des Télécommunications, Evry, France. He received the Diplôme d'Études Approfondies from the University Paris 6, France, the Ph.D. degree from École Nationale Supérieure des Télécommunications (ENST), Paris, France and the Habilitation from the University of Versailles, France.

He is a Professor in the Computer Science and Networks Department, ENST, a subsidiary of the Groupement des Écoles des Télécommunications (GET), Paris, France. He is coauthor of two collective books in English on *ATM networks* (Norwell, MA: Kluwer, 2000) and on *Optical Networks* (Reading, MA: Addison-Wesley, 2003), respectively. He is the author of a book in English on *Broadband Access Systems* (Norwood, MA: Artech House, 2003). He has authored or coauthored about 80 technical papers in IEEE or IFIP international journals or conferences. His research activities are focused on traffic management in broadband local loops and on all-optical WDM networks planning. He has been appointed as an expert by the French Ministry of Industry, the Flemish Government of Belgium and the National Science Foundation of the United States.

Emmanuel Dotaro (M'98) received the M.Sc. degree from the University of Versailles, France, in 1996. He is working toward the Ph.D. degree in computer science from Institut National des Télécommunications, Evry, France, while teaching at the University of Versailles.

He joined Alcatel Research Labs, Marcoussis, France, in 1999, where he is now coordinating activities on packet/transport integration in metropolitan networks. His current research interests include performance evaluation and network dimensioning, data, control (GMPLS) and management architectures, and solutions enabling integrated approaches.

Richard Douville received the M.Sc. degree in electrical engineering from the University of Versailles, France, in 2000. He is currently working toward the Ph.D. degree at the Transport Network System and Architecture Unit, Alcatel Research and Innovation Center, Marcoussis, France.

His main interests are IP over optical internetworking issues, including network and system architectures, traffic engineering, and performance evaluation.