On the Design of Fault-Tolerant Logical Topologies in Wavelength-RoutedNetworks supporting Packet Datagrams

(Article begins on next page)

03 May 2024

On the Design of Fault-Tolerant Logical Topologies in Wavelength-Routed Packet Networks

Antonio Nucci, *Member, IEEE*, Brunilde Sansò, *Member, IEEE*, Theodor Gabriel Crainic,
Emilio Leonardi, *Member, IEEE*, and Marco Ajmone Marsan, *Fellow, IEEE*

Abstract—In this paper, we present a new methodology for the design of fault-tolerant logical topologies in wavelength-routed optical networks supporting Internet protocol (IP) datagram flows. Our design approach generalizes the "design protection" concepts, and relies on the dynamic capabilities of IP to reroute datagrams when faults occur, thus achieving protection and restoration, and leading to high-performance cost-effective fault-tolerant logical topologies. In this paper, for the first time we consider resilience properties during the logical topology optimization process, thus extending the optimization of the network resilience also to the space of logical topologies. Numerical results clearly show that our approach outperforms previous ones, being able to obtain very effective survivable logical topologies with limited computational complexity.

Index Terms—Fault-tolerance, Internet protocol (IP), logical topology design (LTD), protection, resilience, restoration, survivability, wavelength-division multiplexing (WDM).

I. INTRODUCTION

OPTICAL networks exploiting wavelength-division multiplexing (WDM) and wavelength routing (WR) are promising architectures for the implementation of high-capacity Internet protocol (IP) infrastructures. Indeed, such networks permit the exploitation of the huge fiber capacity, with no need for complex processing functionalities in the optical domain. In WR IP networks, nodes comprise an optical section, and an electronic section; the former is an optical cross-connect (OXC), while the latter is a high-capacity IP router. Nodes are connected by optical fibers over which a WDM scheme is implemented. At each node, incoming WDM channels can either be transparently connected to outgoing channels through the OXC, possibly after wavelength conversion (without processing of in-transit information), or converted to the electronic domain, so that packets can be passed to the IP router, processed, and possibly retransmitted after IP routing. This setup allows the definition within the optical domain of

semipermanent optical pipes called "lightpaths" or "logical links" that may extend over several physical links. Thus, lightpaths can be seen as chains of physical channels through which packets are moved from one router to another toward their destinations. OXCs transparently connect the incoming WDM channels corresponding to in-transit lightpaths, and convert to the electronic domain the incoming WDM channels corresponding to terminating lightpaths. The set of lightpaths and routers defines a logical topology, overlayed to the physical topology made of optical fibers and OXCs.

In order to best exploit the capacity of a WDM infrastructure, a crucial task is the identification of the best feasible logical topology for the transport of a given traffic pattern. In recent years, the logical topology design (LTD) problem in WDM networks was extensively studied, considering a number of different setups. It was shown that finding the optimal logical topology is an NP-hard problem and thus computationally intractable for large size networks [1], [2]. Therefore, several heuristic approaches have been proposed in the literature (see, for instance, [3]–[5]).

One of the most critical aspects that operators of IP over WDM networks must face on a daily basis, is related to *reliability* and *availability*. Today, failures are more common than one might expect; in the backbone of a large international Tier-1 carrier, failures happen every day [6]. Most of them are due to IP equipment failures like router hardware/software failures, or protocol misconfigurations, but roughly 12% of all failures are related to the optical layer [7]. Although WDM failures are more rare, they bring about a disturbing instability in the higher layers. With technologies such as WDM, a single fiber failure can bring down a large number of logical links. Sometimes the logical topology even becomes disconnected, and some nodes become isolated from the rest of the network. Even if the logical connectivity is not affected, drastic changes in the traffic flowing at the IP layer are visible, because of the rerouting of many traffic flows on different IP paths. In spite of these frequent failures, the carrier must guarantee to each customer a specific service level agreement (SLA).

In the past, carriers used to implement a multilayer recovery scheme. Each layer is equipped with its own protection/restoration schemes, and reacts to its own layer equipment failures. Synchronous optical network (SONET) is used to offer protection and fast restoration of service at the WDM layer. Protection paths must be precomputed, and wavelengths must be reserved in advance, at the time of connection setup. Physical failures

are completely transparent to the IP layer, and the restoration is provided in less than 50 ms. The dynamic capabilities of Interior Gateway Routing Protocol (IGP), such as OSPF or IS-IS, is used to react to IP link failures. When a failure happens at the IP layer, the IGP detects the failure and automatically recomputes alternate routes around the failed link. Today, most ISPs are deciding to gradually remove SONET due to the high cost of optical equipment and the huge amount of redundant capacity, needed to reroute traffic in case of WDM failures, but never used in the normal operation state [8]. SONET framing is being kept only for failure detection purposes, and SONET protection is allowed only in highly dense areas with high failure probability.

Recently, a new fault-tolerant methodology known as design protection has been gaining a lot of attention in the optical community [9]–[11]. In this case, restoration is obtained by exploiting only the dynamic capabilities of IP routing. When a physical link fails in the optical network, the IP routing algorithm is able to update its tables, and restore disrupted paths, if the set of nondisrupted lightpaths still forms a connected topology. In order to achieve a good degree of fault resilience, it is fundamental *to map* (i.e., to route) each lightpath onto the physical topology in such a way that, given any single physical link failure, the set of nondisrupted lightpaths still forms a connected network. Thus, an optimization of the physical mapping of lightpaths is desirable [10], [11].

In this paper, we generalize the *design protection* approach of [9] and [10] and partially of [11]. We provide, in fact, a powerful framework for the design of logical topologies with a good degree of fault resilience. Our approach differs from the approach proposed in [9]–[11] by considering the resilience properties of the topology directly during the LTD optimization process, thus extending the optimization of the network resilience properties also to the space of logical topologies. By implementing such an idea, we drastically reduce *the level of physical protection* required to achieve a desired level of physical fault-tolerance. This implies less wasted bandwidth and less reserved wavelengths in the normal operation state, but slower recovery after failure due to the execution of signaling and management procedures. Thus, *design protection* should not be necessarily considered as an alternative to physical protection, rather as a complementary technique that may be successfully employed in order to reduce the level of physical protection required to achieve a desired level of fault-tolerance.

The rest of the paper is organized as follows. In Section II, we introduce the problem and formalize the approach aimed at finding logical topologies with good degree of fault resilience. We first propose an integer linear programming formulation by relaxing the shortest path routing requirement. Then we consider the more realistic case where shortest path routing is assumed, which leads to an integer nonlinear formulation. In Section III, we propose a Tabu search methodology to find good solutions while limiting computational effort. Details of the method are provided in Sections IV and V, while Section VI is devoted to the analysis of its theoretical complexity. Section VII contains the results of several sets of experiences, and Section VIII concludes the paper.

II. PROBLEM STATEMENT

The fault-tolerant logical topology design problem (FLTDP) under a given traffic pattern can be stated as follows:

GIVEN:

i) an existing physical topology (which must be at least 2-connected), comprising nodes equipped with a limited integer number of tunable transmitters and receivers, connected by optical fibers that support a limited number of wavelengths;

ii) a traffic matrix whose elements represents the traffic volumes exchanged by sources and destinations;[1]

iii) a multihop IP routing strategy for packets;

iv) a set of single physical link failures.

FIND:

a logical topology (i.e., a set of lightpaths through which packets are routed from source to destination) and a "mapping," (i.e., a set of physical routes for each IP lightpath), such that an appropriate objective function depending on all network states (i.e., no failure and all single link failures) is optimized.

A. Problem Formulation

In this section, we report two variants of the FLTD problem formulation. For both, we consider that wavelength converters are available at each node. In the first case, the paths taken by the IP packets are not restricted to be the shortest. This leads to an integer linear programming (ILP) formulation. When the shortest path requirement is added, however, the problem keeps its integrality nature, but becomes nonlinear, which greatly increases the complexity of its resolution. This variant is presented at the end of the section. Unfortunately, since all Tier-1 ISPs use routing protocols based on shortest paths, the more realistic formulation would be the nonlinear one. However, we believe that the ILP model represents a powerful tool to find a theoretical lower bound to test the accuracy of the heuristic approaches proposed for the solution of the nonlinear formulation.

1) Notation: We adopt the notational typology for multilayered networks presented in [12]. The supra-index indicates the layer, starting by the lowest layer, zero, that represents the physical network. Let $G^0 = (V, E^0)$ be the unidirectional graph representing the physical topology. It is composed by the set of OXC nodes V interconnected by optical fibers represented by set E^0. Let $|V| = N$ be the cardinality of set V and $|E^0| = M$ that of set E^0. Let R_i and T_i be the numbers of receivers and transmitters at physical node $i \in V$. Let S_k be the network state, where S_0 represents the no-failure state, while S_v for $v \geq 1$ is the state of failure of optical fiber $v \in E^0$. Let $\mathcal{S}$ be the set of all operational states, whose cardinality is $|S| = M + 1$. Let $\mathcal{E}$ be the set of all possible lightpaths in any logical topology. Let $G^1(S_0) = (V, E^1(S_0))$ be the directed graph representing the logical topology in the no failure state. It is composed of IP routers V interconnected by lightpaths $E^1(S_0) \subseteq \mathcal{E}$. Note that

in order to simplify the notation, we assume that there is a router associated with each OXC, and, by abuse of notation, we equate the set of routers with the set of OXC. However, our formulation can be easily extended to the more general case.

Let $G^1(S_v) = (V, E^1(S_v))$ denote the logical topology in the network state S_v, obtained from $G^1(S_0)$ by dropping all the lightpaths $u \in E^1(S_0)$ crossing the optical fiber $v \in E^0$. Let $\Lambda = (\lambda_{sd})$ indicate the peak-time traffic matrix where each entry λ_{sd}, in arbitrary units, represents the peak-time traffic flow between source s and destination d.

2) Decision Variables: Three types of binary variables are introduced into the formulation: X_u, Y_{uv}, $t_u^{sd}(S_v)$, that correspond, respectively, to logical topology, mapping, and routing.

The logical topology variables $X_u \in \{0,1\}$ describe the lightpaths included in the logical topology $G^1(S_0)$

$$X_u = \begin{cases} 1, & \text{if lightpath } u \in \mathcal{E} \\ & \text{belongs to the logical topology } G^1(S_0) \\ 0, & \text{otherwise.} \end{cases}$$

Then we can state that logical topology $G^1(S_0) = (V, E^1(S_0))$ comprises the lightpaths $E^1(S_0) = \{u : X_u = 1, u \in \mathcal{E}\}$.

The mapping variables $Y_{uv} \in \{0,1\}$ contain the routing information of lightpaths belonging to the logical topology $G^1(S_0)$ over the physical topology G^0

$$Y_{uv} = \begin{cases} 1, & \text{if lightpath } u \in \mathcal{E} \text{ crosses} \\ & \text{the optical fiber } v \in E^0 \\ 0, & \text{otherwise.} \end{cases}$$

The variables $t_u^{sd}(S_v)$ contain the information related to routing of packets on the logical topology $G^1(S_v)$

$$t_u^{sd}(S_v) = \begin{cases} \lambda_{sd}, & \text{if traffic } s \to d \text{ crosses} \\ & \text{lightpath } u \in \mathcal{E} \text{ in state } S_v \\ 0, & \text{otherwise.} \end{cases}$$

We notice that traffic splitting is not allowed in our model (i.e., all the traffic originated in s and destined in d is forced to follow the same route). The model can be easily extended to consider traffic splitting by relaxing the variables $t_u^{sd}(S_v)$ to the continuous.

3) Constraints: Let $\Gamma^+(i)$ be the set of lightpaths outgoing from node $i \in V$ and $\Gamma^-(i)$ be the set of lightpaths incoming to node $i \in V$. Let $\Theta^+(i)$ be the set of physical links outgoing from node $i \in V$ and $\Theta^-(i)$ be the set of physical links incoming to node $i \in V$. Finally, let $O(u)$ be the origin node and $D(u)$ the destination node of lightpath $u \in \mathcal{E}$. We can then write the model constraints.

• Connectivity:

$$\sum_{u \in \Gamma^+(i)} X_u \leq T_i \quad \forall i \in V \tag{1}$$

$$\sum_{u \in \Gamma^-(i)} X_u \leq R_i \quad \forall i \in V \tag{2}$$

where inequalities (1) indicate that the number of lightpaths outgoing from each node cannot be larger than the number of transmitters in the node, for each logical topology in the no failure state ($G^1(S_0)$); inequalities (2) indicate that the number of lightpaths incoming to each

node cannot be larger than the number of receivers in the node, for each logical topology in the no failure state ($G^1(S_0)$).

• Routing:

$$\sum_{u \in \Gamma^+(i)} t_u^{sd}(S_v) - \sum_{u \in \Gamma^-(i)} t_u^{sd}(S_v)$$
$$= \begin{cases} \lambda_{sd}, & \text{if } s = i \\ -\lambda_{sd}, & \text{if } d = i \\ 0, & \text{otherwise} \end{cases}$$
$$\forall s, d, i \in V, \forall S_v \in \mathcal{S} \tag{3}$$

$$t_u^{sd}(S_v) \leq X_u \lambda_{sd}$$
$$\forall s, d \in V, \forall S_v \in \mathcal{S}, \forall u \in \mathcal{E} \tag{4}$$

where (3) represent the routing continuity constraints for packet routes on the logical topology $G^1(S_v)$. They state that for each network operational state, an available (working) path on the logical topology must exist for each source-destination pair; equations (4), instead, state that traffic can be routed only on lightpaths belonging to the logical topology.

• Mapping:

$$Y_{uv} \leq X_u \quad \forall u \in \mathcal{E}, v \in E^0 \tag{5}$$

$$\sum_{v \in \Theta^+(i)} Y_{uv} - \sum_{v \in \Theta^-(i)} Y_{uv} = \begin{cases} 1, & \text{if } O(u) = i \\ -1, & \text{if } D(u) = i \\ 0, & \text{otherwise} \end{cases}$$
$$\forall u \in \mathcal{E}, \forall i \in V \tag{6}$$

$$\sum_{s,d \in V} t_u^{sd}(S_v) \leq \left(\sum_{s,d \in V} \lambda_{sd} \right) (1 - Y_{uv})$$
$$\forall u \in \mathcal{E}, \forall v \in E^0, \forall S_v \in \frac{\mathcal{S}}{\{S_0\}} \tag{7}$$

where inequalities (5) ensure that only the lightpaths in the considered logical topology are mapped; equations (6) represent routing continuity constraints for lightpaths on the physical topology G^0; and inequalities (7) impose that all the lightpaths that cross the physical link v are not available in state S_v.

• Limit on the number of wavelengths:

Let W_v be the number of wavelengths supported on each fiber. The set of lightpaths $v \in E^1(S_0)$ must satisfy the following constraint:

$$\sum_{u \in \mathcal{E}} Y_{uv} \leq W_v \quad \forall v \in E^0 \tag{8}$$

which indicates that the number of lightpaths that cross each optical fiber has to be smaller than the wavelength number.

4) Objective Function: The objective function must be carefully selected, in order to obtain the best tradeoff between network performance in normal conditions and fault-resilience properties (see [15] and [16]). Since the network performance depends on the network failure states, the objective function must combine the network performance levels under different network failure states. We selected as objective of the optimization process the minimization of the network congestion

level, defined as the maximum amount of traffic flowing on any lightpath, belonging to the logical topology under any failure state

$$\min H$$

with

$$H \geq \left\lceil \sum_{s,d \in V} t_u^{sd}(S_v) \right\rceil \quad \forall S_v \in \mathcal{S}, \forall u \in \mathcal{E}. \quad (9)$$

5) Observations: Note, first, that in the above formulation the routing of packets on the logical topology is unspecified; thus, the minimization of the network congestion level is jointly performed on all admissible logical topologies and routings.

Also note that under the assumption that at least one topology exists, the above ILP model provides a logical topology that can tolerate any single physical link failure. Indeed, the resulting logical topology is connected, under any single link failure (3). If no topology exists, the ILP model produces an infeasible solution warning message.

Lightpath capacity constraints are ignored in the above formulation, for the sake of the model simplicity; we notice, however, that the minimization of the network congestion level corresponds to a minimization of the lightpath capacity needed to guarantee an efficient transport of the offered traffic. Thus, a minimization of the network congestion level leads to the minimization of the capacity needed to guarantee good performance.

6) Extension to Shortest-Path Routing: In order to restrict the optimization to act only on the set of the admissible logical topologies with shortest path routing, we need to introduce some extra variables and constraints.

Let us introduce an extra set of variables $\tau_u^{sd}(S_v) \in \{-1, 0, 1\}$, that represent a possible alternate routing with respect to the routing specified by $t_u^{sd}(S_v)$ on the logical topology

$$\tau_u^{sd}(S_v) = \begin{cases} 1, & \text{if traffic } s \to d \text{ is re-routed on} \\ & \text{lightpath } u \in \mathcal{E} \text{ in state } S_v \\ -1, & \text{if traffic } s \to d \text{ is no longer routed on} \\ & \text{lightpath } u \in \mathcal{E} \text{ in state } S_v \\ 0, & \text{otherwise} \end{cases}$$

$\tau_u^{sd}(S_v)$ must satisfy the following constraints:

$$\sum_{u \in \Gamma^+(i)} \tau_u^{sd}(S_v) - \sum_{u \in \Gamma^-(i)} \tau_u^{sd}(S_v) = 0$$
$$\forall s, d, i \in V, \forall S_v \in \mathcal{S} \quad (10)$$

$$\tau_u^{sd}(S_v) \leq X_u - Y_{uv} - \frac{t_u^{sd}(S_v)}{\lambda_{sd}}$$
$$\forall s, d \in V, \forall u \in \mathcal{E}, \forall v \in E^0, \forall S_v \in \frac{\mathcal{S}}{\{S_0\}} \quad (11)$$

$$\tau_u^{sd}(S_0) \leq X_u - \frac{t_u^{sd}(S_0)}{\lambda_{sd}} \quad \forall s, d \in V, \forall u \in \mathcal{E} \quad (12)$$

$$-\tau_u^{sd}(S_v)\lambda_{sd} \leq t_u^{sd}(S_v)$$
$$\forall s, d \in V, \forall u \in \mathcal{E}, \forall S_v \in \mathcal{S} \quad (13)$$

where (10) represent the routing continuity constraints for the rerouting on logical topology $G^1(S_v)$. Equations (11) and (12),

instead, state that traffic can be rerouted only on a path consisting of working lightpaths belonging to the logical topology; equations (13), finally, state that, after rerouting, routes defined by $t_u^{sd}(S_v)$ may be no longer valid.

Finally, we define G as

$$G = -\sum_{\substack{S_v \in \mathcal{S} \\ u \in \mathcal{E} \\ s,d \in V}} \tau_u^{sd}(S_v).$$

Note that G represents the difference between the total path length before rerouting and after rerouting. It is possible to find a set of $\tau_u^{sd}(S_v)$ such that G assumes positive values whenever the set of $t_u^{sd}(S_v)$ does not describe a shortest path routing. In conclusion, if and only if the set $t_u^{sd}(S_v)$ defines a shortest path routing, we find $\max_{\tau_u^{sd}} G = 0$; thus, selecting

$$\min_{t_u^{sd}} \left[\epsilon H + \max_{\tau_u^{sd}} G \right]$$

as objective function, where H is defined in (9), and $\epsilon < 1/\sum_{sd} \lambda_{sd}$, we obtain the result of restricting the optimization to the set of logical topologies implementing a shortest path routing. Indeed, we observe that, by construction, $0 < \epsilon H < 1$, while $\max G$ can assume only nonnegative integer values. Thus, $\epsilon H + \max_{\tau_u^{sd}} G > 1$, whenever variables $t_u^{sd}(S_v)$ do not define a shortest path routing; on the other hand, $\epsilon H + \max_{\tau_u^{sd}} G < 1$ if variables $t_u^{sd}(S_v)$ define a shortest path routing. As a consequence, we can state that the optimal solution of the previous problem is the logical topology which minimizes the network congestion level under a shortest path routing. Note, however, that the resulting objective function in this specific case is nonlinear. Thus, the formulation falls in the class of integer nonlinear programming problems, and no general methodologies and tools are available for an optimal solution of this formulation.

III. SOLUTION STRATEGY

The FLTD problem is NP-hard, since it is a generalization of the traditional LTD problem that was proved to be NP-hard. Even for moderate size networks, an optimal solution of the FLTD problem appears to be quite problematic due to the large number of variables and constraints involved in the formulation. Thus, the development of heuristic solution methodologies is required. The heuristic approach to the FLTD problem proposed in [9] and [13] consists in decomposing the whole problem into two independent subproblems: the LTD problem, in which the logical topology optimization is performed on the basis of the congestion level in the full operational state (S_0), thus ignoring the resilience property of the solution; and the fault-tolerant mapping (FM) problem, according to which the mapping of the logical topology onto the physical topology is aimed at the achievement of good resilience properties. While the LTD problem has been widely investigated in the literature, and many algorithms have been proposed ([2]–[5]), the FM problem has been considered only recently. In [9] this problem was found NP-complete and a heuristic approach based on the application of the Tabu search optimization algorithm has been proposed, while in [13] an ILP formulation of the problem is
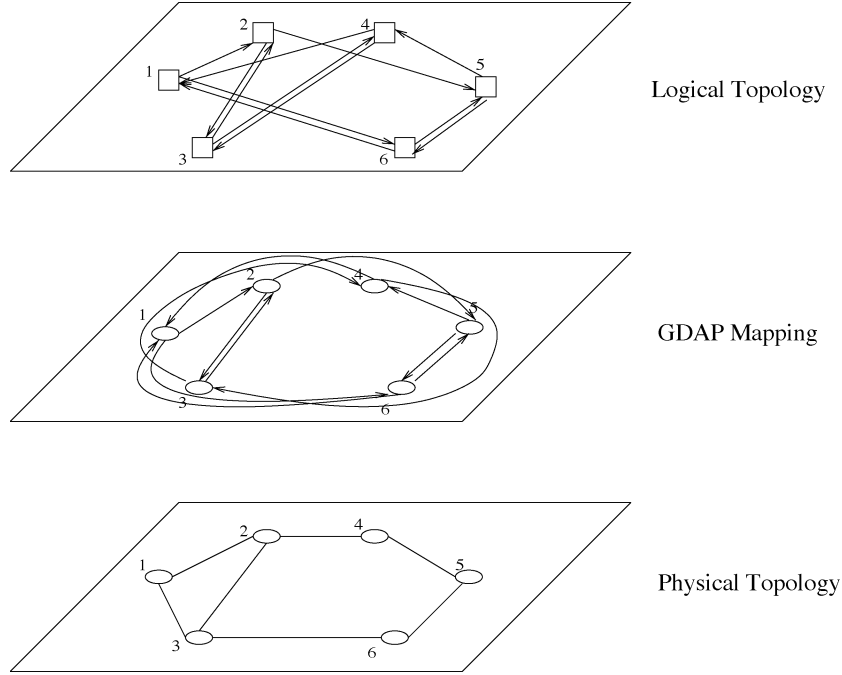
Fig. 1. Example of GDAP mapping with $W_v = 3 \; \forall v \in E^0$ and $T_i = R_i = 2 \; \forall i \in V$.

provided and solved for instances of moderate size (e.g., physical topology with 14 nodes and 21 links and logical topologies with 14 routers and node-degree equal to 3, 4, and 5) applying the CPLEX [14] optimization tool.

In this paper, we adopt a different strategy for the solution of the FLTD problem. We apply Tabu search for the optimization of the logical topology, considering both the case of no failure, and all possible cases of a single physical link failure in the network. For each considered logical topology, lightpaths are routed over the physical topology, and the number of wavelengths to be used on each fiber is computed. Both lightpath routing and wavelength assignment are obtained through a new heuristic algorithm, called GDAP, and described in Section IV.

Finally, the traffic routing on the lightpaths forming the logical topology is taken to be shortest path.

IV. MAPPING BETWEEN PHYSICAL AND LOGICAL TOPOLOGY: GDAP

The definition of algorithms that optimally map the lightpaths on the physical topology is an important subproblem of FLTDP. This problem is related to inequalities (5) and (6) in Section II-A3. The mapping problem can be stated as follows: *given a logical topology, find a routing for each lightpath of the logical topology over the physical topology, such that the negative effects of a single optical link failure are minimized.*

Since the mapping problem is only a part of FLTDP, the utilization of a computationally expensive algorithm to solve the mapping could have a disruptive impact on the CPU time necessary for the solution of the entire problem. Thus, for the solution of the mapping problem, we present a simple greedy algorithm, the greedy disjoint alternate path (GDAP), whose computational complexity is small. A brief description of the GDAP algorithm follows.

GDAP Algorithm: Let $OR(i)$ and $IR(i)$ be the sets of already routed lightpaths, respectively outgoing from and incoming to node i, and let $ON(i)$ and $IN(i)$ be the sets of outgoing and incoming lightpaths not yet routed. Let V_{ij} be the lightpath belonging to the logical topology, with endpoints i and j. Let $\mathcal{O}$ denote a set of nodes. Initialize $\mathcal{O}$ to the set of all nodes in the network.

Step 0: Route all lightpaths V_{ij} whose endpoints are adjacent in the physical topology. Insert V_{ij} in $(OR(i), IR(j))$ and remove V_{ij} from $(ON(i), IN(j))$.

Step 1: If $\mathcal{O} = \emptyset$, STOP, otherwise randomly select a node $i \in \mathcal{O}$ and remove i from $\mathcal{O}$.

Step 2: If $ON(i) = \emptyset$, GOTO Step 3, otherwise randomly select each $V_{ik} \in ON(i)$ and find the shortest path for V_{ik} which is physically disjoint from the routes on which the $V_{ij} \in OR(i)$ and the $V_{jk} \in IR(k)$ have already been routed. If no such physical path exists, V_{ik} is routed on the shortest path. If also the shortest path is not available, due to the lack of free wavelengths, lightpath V_{ij} is not mapped.

Step 3: If $IN(i) = \emptyset$ GOTO Step1, randomly select each $V_{ki} \in IN(i)$ and try to find a route for V_{ki} which is physically disjoint from the routes on which the $V_{ji} \in IR(i)$ and the $V_{kj} \in OR(k)$ have already been routed. If a physically disjoint route for V_{ki} has not been found, V_{ki} is routed on the shortest path. If also the shortest path is not available, due to the lack of free wavelengths, lightpath V_{ji} is not mapped.

An example of mapping produced by GDAP is shown in Fig. 1. The lightpaths that are mapped first over the physical topology are those whose end points are two adjacent physical nodes (see, for example, the lightpaths $1 \rightarrow 2$, $2 \rightarrow 3$, $3 \rightarrow 2$, etc.). Then, starting from node 1, the steps 2 and 3 of GDAP
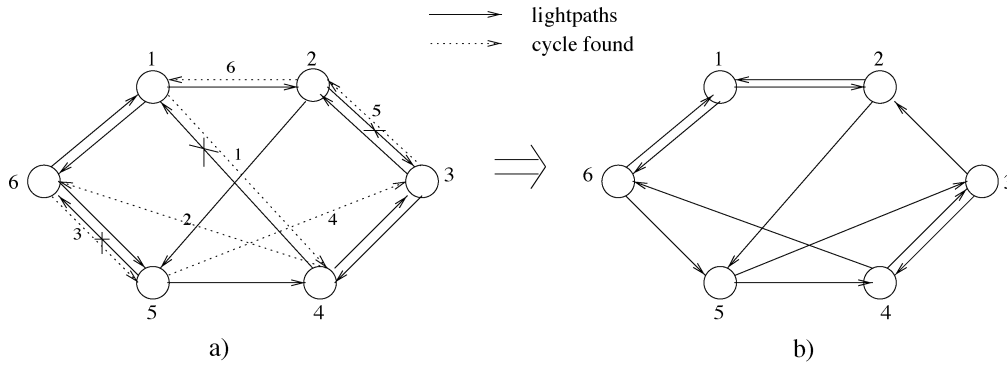
Fig. 2. Perturbation using a cycle $C(1,6)$. a) Old solution; b) new solution.

are iteratively and sequentially applied to all nodes of the network. Focusing on node 1, GDAP maps the outgoing remaining lightpath $1 \rightarrow 6$ over the optical fibers $(1,3)$ and $(3,6)$. Note that all the possible routes for lightpath $1 \rightarrow 6$ must comprise fiber $(1,3)$, since lightpath $1 \rightarrow 2$ is already routed on fiber $(1,2)$. Concerning the incoming lightpaths of node 1, GDAP maps lightpath $4 \rightarrow 1$ over the optical fibers $(4,2)$ and $(2,1)$. It is interesting to look at the mapping for lightpath $6 \rightarrow 1$. It must cross optical fiber $(6,3)$, since $(6,5)$ has been already used by lightpath $6 \rightarrow 5$. Then the only possible physical path for lightpath $6 \rightarrow 1$ is represented by the sequence of optical fibers $(6,3)$ and $(3,1)$. Note that if no disjoint path for $6 \rightarrow 1$ were possible, the algorithm would have selected one shortest path.

It is worth noting that GDAP computes both the number of lightpaths crossing each fiber, and the number of wavelengths used per fiber; if the maximum number of wavelengths over a fiber is limited, some lightpaths may be impossible to map over the physical topology.

V. TABU SEARCH FOR FLTDP: TABUFLTDP

A. General Description of Tabu Search

The heuristic we propose for the solution of FLTDP relies on the application of the Tabu search (TS) methodology. The TS algorithm can be seen as an evolution of the classical local optimum solution search algorithm called steepest descent (SD); however, thanks to the TS mechanism that allows worsening solutions to be also accepted, contrary to SD, TS is less likely to be subject to local minima entrapments. TS is based on a partial exploration of the space of admissible solutions, finalized to the discovery of a good solution. The exploration starts from an initial solution that is generally obtained with a greedy algorithm, and when a stop criterion is satisfied, the algorithm returns the best visited solution. For each admissible solution, a class of neighbor solutions is defined. A neighbor solution is defined as a solution that can be obtained from the current solution by applying an appropriate transformation, called a *move*. The set of all admissible moves uniquely defines the *neighborhood* of each solution.

At each iteration of the TS algorithm, all solutions in the neighborhood of the current one are evaluated, and the best is selected as the new current solution. Note that, in order to efficiently explore the solution space, the definition of neighborhood may change during the solution space exploration; in this way, it is possible to achieve an *intensification* or a *diversification* of the search in different solution regions.

A special rule, the *Tabu list*, is introduced in order to prevent the algorithm from deterministically cycling among already visited solutions. The Tabu list stores the last accepted moves; while a move is stored in the Tabu list, it cannot be used to generate a new move. The choice of the Tabu list size is very important in the optimization procedure: too small a size could cause the cyclic repetition of the same solutions, while too large a size can severely limit the number of applicable moves, thus preventing a good exploration of the solution space.

B. Fundamental Aspects of TabuFLTDP

In order to put in place a Tabu procedure, we must define the following elements:

- the choice of an initial solution;
- the definition of the moves and the neighborhood;
- the evaluation of the visited solutions;
- the stopping criterion.

1) Initial Solution: As initial solution, we selected the result of the D-MLTDA heuristic [3], which is briefly described in the Appendix. This heuristic initially considers a fully connected logical topology. Traffic is routed according to a shortest-path routing protocol, and a set of least-loaded lightpaths is sequentially removed from the logical topology until the degree constraints are satisfied.

2) The Moves and the Neighborhood: Let T represent a given feasible topology and $\mathcal{N}(T)$ be the neighborhood of such a topology when the Tabu moves are applied. A new solution $T' \in \mathcal{N}(T)$ is found by searching for cycles of a given length and erasing the right number of lightpaths to keep the degree constraint feasible. In a more detailed manner, let us denote by l the fixed length of the cycle, l being an even number such that $l \leq 8$. Let us assume that $n_1, n_2, \ldots, n_l$ are the nodes to be visited in the cycle, starting at node n_1. From a given node n_i, the next node to be visited, n_{i+1}, is found as follows.

- If i is an odd number, choose an incoming lightpath and travel in the opposite direction. The resulting node is n_{i+1}.
- If i is even, choose any node that has not yet been visited in the cycle as n_{i+1}.

Once the cycle has been defined, the new degree of each node is assessed. The superfluous lightpaths are removed in those nodes presenting a degree larger than their original value. An

example of the procedure is given in Fig. 2 for a cycle of length 6. We found that the visited nodes in the cycle are $n_1 = 1$, $n_2 = 4$, $n_3 = 6$, $n_4 = 5$, $n_5 = 3$, and $n_6 = 2$. Lightpaths 2-3, 4-1, and 5-6 are removed from the topology and replaced by lightpaths 5-3 and 4-6 to get to the new topology.

This procedure guarantees that degree constraints are not violated, thus generating a valid move. Note that with this perturbation, it is very easy and fast to implement a *diversification* and/or *intensification* criterion by exploring a region of the solution space with small cycles, and move to another region of the solution space with large cycles.

Then, the neighborhood of the current solution T (i.e., $\mathcal{N}(T)$) is generated by considering sequentially all network nodes and applying all possible cycles of length l.

3) Solution Evaluation: Each solution in the neighborhood is evaluated by routing the traffic into the topology for all network states (i.e., no failure and all single link failure states), and computing the network congestion level in each state. The solution with the minimum congestion level is selected as the new current solution and the lightpaths selected during its generation are stored in the Tabu list.

4) Stopping Criterion: The search procedure is stopped when a given number of iterations is reached. The number of iterations should be chosen relative to the size of the network and to achieve a good tradeoff between computational time and quality (distance from the optimal solution) of the solutions reached.

C. TabuFLTDP Pseudocode

In this section, we present the pseudocode of our Tabu procedure. First, let us define some useful notation.

- $F_{\text{eval}}(T)$ is the evaluation function to compute the merit coefficient of logical topology T. It returns M, the network congestion level. Note that the evaluation of M requires the execution of the routing algorithm on the logical topology and of the GDAP mapping algorithm.
- *BuildInitialSolution* is used to build an initial logical topology applying D-MLTDA.
- *BuildCycle*(T, l) is used to build the neighbor solution of the current solution T, using cycles of length l. When the *diversification* criterion has to be used, the cycle is longer than in normal TabuFLTDP. We denote by q the length of the normal cycle and by p the length of the cycle used for the *diversification* criterion, where $p \gg q$.
- *BuildNeighborhood* (T) is a procedure to build the neighborhood of the current solution T, by applying iteratively the *BuildCycle*(T, l) procedure.
- *BestNeighSol* (T) is a procedure that evaluates each solution in the neighborhood of T, and returns the best solution. The evaluation is based on $F_{\text{eval}}(T)$.
- *TabuList* is a fixed size Tabu list to store the latest moves.
- $\mathcal{N}(T)$ is the neighborhood of logical topology T, built applying procedure *BuildNeighborhood* (T) and using only cycles not belonging to *TabuList*.
- T, T^*, and T^{**} represent, respectively, the current logical topology, the best logical topology in $\mathcal{N}(T)$, and the best solution found by FLTDP.

```
T:=BuildInitialSolution
IterBest:=1
T**:=T
M**:=M
counterDiv:=1
FOR  counter=1 TO IterationsNumber
     IF  counterDiv=LimitDiv THEN
         T:=BuildCycle(T,p)
         counterDiv:=1
     ENDIF
     N(T):=BuildNeighborhood(T)
     T*:=BestNeighSol  and  update TabuList
     IF  M* <= M** THEN
         T**:=T*
         M**:=M*
         IterBest:=counter
         counterDiv:=1
     ELSEIF
         counterDiv++
     ENDIF
     T:=T*
ENDFOR
```

Fig. 3.   TabuFLTDP pseudocode.

- M, M^*, and M^{**} represent, respectively, the merit associated with the logical topologies T, T^*, and T^{**}.
- *IterationsNumber* is the number of iterations.
- *LimitDiv* is the number of consecutive iterations without improvements, expressed by the variable *counterDiv*, after which the *diversification criterion* is applied. When this happens, only *one cycle* of length p is generated; the cycle is such to change several lightpaths at the same time and then visit a different area of the solution space. After the new solution is generated, the procedure works as before, using cycles of length q.
- *IterBest* is the iteration at which T^{**} is found.

The pseudocode for TabuFLTDP is given in Fig. 3.

VI. COMPLEXITY

We now discuss the complexity of the proposed heuristics. Let $\Delta = T_i = R_i \, \forall i \in V$ denote the identical in/out degrees for each node in the logical topology. We further suppose $\Delta \leq N$. For each analyzed logical topology, we route i) its lightpaths over the physical topology with the GDAP algorithm and ii) the traffic over the logical topology.

The GDAP algorithm has complexity $O(N\Delta(M + \Delta N \log(N\Delta)))$, since at most $O(N\Delta)$ iterations are executed, while at each iteration at most $O(M + \Delta N \log(N\Delta))$ operations are required; $O(M)$ operations, indeed, are necessary to update the cost of the links of the physical topology and $O(N \log(\Delta N))$ operations are necessary to run the Dijkstra algorithm. Since $M \leq N^2$, the GDAP complexity is upper bounded by $O(N^3\Delta + \Delta^2 N^2 \log(N\Delta))$.
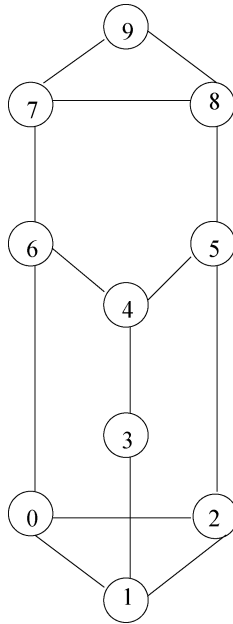
Fig. 4. Physical topology—10 nodes, 14 links.

Fig. 5. Physical topology—10 nodes, 13 links.

To evaluate solutions, it is necessary to route the traffic. The routing algorithm requires $O(N^2 \log(N\Delta))$ operations (Dijkstra algorithm).

The D-MLTDA heuristic to evaluate the initial solution has complexity $O(N^4)$, since at most $O(N)$ iterations are needed to complete the algorithm, and at each iteration it is necessary to route traffic and execute a 1-minimum weight matching (1-m WM) algorithm whose complexity is $O(N^3)$.

Let us now focus on the complexity of the TS algorithm. At each iteration, the evaluation of all solutions in the neighborhood is necessary; this requires $O(\Delta^2 N^2 (N^3 \Delta + (\Delta^2 N^2 + MN^2) \log(N\Delta)))$ operations, since $O(\Delta^2 N^2)$ neighbors are evaluated (assuming perturbations are generated using cycles of length 4), and the evaluation of each solution requires the execution of the GDAP algorithm and the execution of the routing algorithms for each failure state (i.e., $M+1$ times). If the number of iterations is I, the resulting complexity is $O(I\Delta^2 N^2 (N^3 \Delta + (\Delta^2 N^2 + MN^2) \log(N\Delta)))$. Thus, the computational complexity of TS is upper-bounded by $O(I\Delta^2 N^6 (\log(N\Delta)))$.

VII. NUMERICAL RESULTS

In this section, we present numerical results obtained with the proposed approach (called joint optimization), and compare them against those obtained by performing a conventional optimization of the logical topology, and then optimally mapping the lightpaths on the physical topology according to the algorithm proposed in [13], and extended in order to deal with a unidirectional logical topology (this approach is called disjoint optimization).

Optimal results are reported for the medium-sized (ten-node) topologies plotted in Figs. 4 and 5, since we were unable to run the optimal mapping for larger networks. Larger instances were heuristically obtained using our proposed approach. The network of Fig. 4 was obtained by removing some nodes and
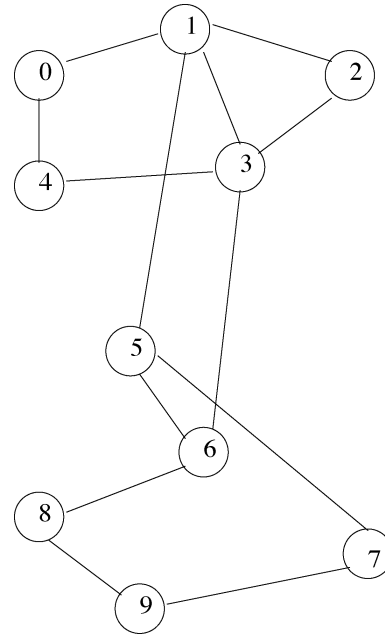
links from the NSF-net topology, while the network of Fig. 5 has the structure of a possible Italian backbone IP network.

We consider randomly generated traffic patterns. The bandwidth required for every source-destination traffic is randomly extracted from an exponential distribution with mean $\mu = 1$.

While TabuFLTDP does not consider lightpath capacities, in the presentation of numerical results in this section we will consider that each lightpath has a fixed capacity, and can thus carry a limited amount of traffic.

As we already noted, considering capacities is not essential in the optimization, where topologies are ranked according to their maximum flow on lightpaths, thus implicitly minimizing the lightpath capacity required to carry a given traffic pattern. On the other hand, including lightpath capacities in the optimization process would lead to a significant increase of complexity. Considering (variable) lightpath capacities in the presentation of results is instead important, since it allows us to estimate the actual amount of traffic that cannot be carried over the logical topologies produced by the different optimization algorithms.

The Tabu parameters used in our experiments, after an initial calibration, were set as follows.

- *TabuList*: A Tabu list of fixed size equal to 7 is used.
- *Cycles size*: During the exploration of the solution space, cycles of length 4 are used. In some cases, however, different perturbation rules are used to implement the *diversification criterion*. In particular, to ease the exit from local minima regions, after 50 iterations without improvement, a cycle of size 6 is used.
- *Stopping Criterion*: The procedure is stopped after a fixed number of iterations. The number of iterations is set to 300, since this value seems to provide a good tradeoff between the conflicting requirements of limiting the CPU time and obtaining good results.

TABLE I
COMPARISON BETWEEN FLTDP AND THE DISJOINT OPTIMIZATION OF LTDP AND OPT-MP WITH $\Delta = 2$ FOR NETWORK 1

Parameters	Disjoint Opt (LTDP+Opt-MP)							
	LC=20	LC=25	LC=30	LC=35	LC=40	LC=45	LC=50	LC=55
C_{S_0} Mean	13.02	13.02	13.02	13.02	13.02	13.02	13.02	13.02
C_S Mean	20.00	25.00	30.00	35.00	38.98	40.48	41.04	41.35
TL Mean [%]	27.19	20.30	13.58	7.48	2.75	1.01	0.36	0.00
C_{S_0} wrst	14.72	14.72	14.72	14.72	14.72	14.72	14.72	14.72
C_S wrst	20.00	25.00	30.00	35.00	40.00	45.00	50.00	52.81
TL wrst [%]	39.39	32.25	26.45	20.65	14.85	9.05	3.25	0.00
Parameters	Joint Opt (FLTDP)							
	LC=20	LC=25	LC=30	LC=35	LC=40	LC=45	LC=50	LC=55
C_{S_0} Mean	15.96	15.96	15.96	15.96	15.96	15.96	15.96	15.96
C_S Mean	20.00	23.44	24.43	24.54	24.54	24.54	24.54	24.54
TL Mean [%]	6.58	1.36	0.13	0.00	0.00	0.00	0.00	0.00
C_{S_0} wrst	19.84	19.84	19.84	19.84	19.84	19.84	19.84	19.84
C_S wrst	20.00	25.00	30.00	30.97	30.97	30.97	30.97	30.97
TL wrst [%]	18.76	7.74	1.14	0.00	0.00	0.00	0.00	0.00

TABLE II
COMPARISON BETWEEN FLTDP AND THE DISJOINT OPTIMIZATION OF LTDP AND OPT-MP WITH $\Delta = 3$ FOR NETWORK 1

Parameters	Disjoint Opt (LTDP+Opt MP)				
	LC=15	LC=20	LC=25	LC=30	LC=35
C_{S_0} Mean	7.51	7.51	7.51	7.51	7.51
C_S Mean	15.00	20.00	24.46	26.85	27.7
TL Mean [%]	15.98	9.26	3.76	0.98	0.00
C_{S_0} wrst	9.18	9.18	9.18	9.18	9.18
C_S wrst	15.00	20.00	25.00	30.00	33.55
TL wrst [%]	22.16	15.72	10.56	4.75	0.00
Parameters	Joint Opt (FLTDP)				
	LC=15	LC=20	LC=25	LC=30	LC=35
C_{S_0} Mean	9.11	9.11	9.11	9.11	9.11
C_S Mean	12.51	13.11	13.11	13.11	13.11
TL Mean [%]	0.15	0.00	0.00	0.00	0.00
C_{S_0} wrst	11.29	11.29	11.29	11.29	11.29
C_S wrst	15.00	15.66	15.66	15.66	15.66
TL wrst [%]	0.76	0.00	0.00	0.00	0.00

TABLE IV
COMPARISON BETWEEN FLTDP AND THE DISJOINT OPTIMIZATION OF LTDP AND OPT-MP WITH $\Delta = 5$ FOR NETWORK 1

Parameters	Disjoint Opt (LTDP+Opt MP)				
	LC=10	LC=15	LC=20	LC=25	LC=30
C_{S_0} Mean	6.07	6.07	6.07	6.07	6.07
C_S Mean	10	13.88	15.76	16.43	16.76
TL Mean [%]	8.98	3.61	1.14	0.38	0.00
C_{S_0} wrst	8.12	8.12	8.12	8.12	8.12
C_S wrst	10.00	15.00	20.00	25.00	27.99
TL wrst [%]	20.87	15.07	9.27	3.47	0.00
Parameters	Joint Opt (FLTDP)				
	LC=10	LC=15	LC=20	LC=25	LC=30
C_{S_0} Mean	6.27	6.27	6.27	6.27	6.27
C_S Mean	7.62	7.62	7.62	7.62	7.62
TL Mean [%]	0.00	0.00	0.00	0.00	0.00
C_{S_0} wrst	7.48	7.48	7.48	7.48	7.48
C_S wrst	9.74	9.74	9.74	9.74	9.74
TL wrst [%]	0.00	0.00	0.00	0.00	0.00

TABLE III
COMPARISON BETWEEN FLTDP AND THE DISJOINT OPTIMIZATION OF LTDP AND OPT-MP WITH $\Delta = 4$ FOR NETWORK 1

Parameters	Disjoint Opt (LTDP+Opt MP)						
	LC=10	LC=15	LC=20	LC=25	LC=30	LC=35	LC=40
C_{S_0} Mean	6.32	6.32	6.32	6.32	6.32	6.32	6.32
C_S Mean	10.00	14.42	17.36	19.96	21.73	22.82	23.31
TL Mean [%]	17.42	10.36	6.90	3.85	1.83	0.57	0.00
C_{S_0} wrst	7.91	7.91	7.91	7.91	7.91	7.91	7.91
C_S wrst	10.00	15.00	20.00	25.00	30.00	35.00	39.41
TL wrst [%]	34.11	28.31	22.51	14.41	9.41	5.11	0.00
Parameters	Joint Opt (FLTDP)						
	LC=10	LC=15	LC=20	LC=25	LC=30	LC=35	LC=40
C_{S_0} Mean	7.05	7.05	7.05	7.05	7.05	7.05	7.05
C_S Mean	8.90	9.01	9.01	9.01	9.01	9.01	9.01
TL Mean [%]	0.22	0.00	0.00	0.00	0.00	0.00	0.00
C_{S_0} wrst	8.63	8.63	8.63	8.63	8.63	8.63	8.63
C_S wrst	10.00	10.98	10.98	10.98	10.98	10.98	10.98
TL wrst [%]	2.04	0.00	0.00	0.00	0.00	0.00	0.00

In Tables I–IV, we report results obtained with the disjoint and the joint optimization techniques, for different logical network configurations on the physical topology plotted in Fig. 4. In each column, the results for a particular lightpath capacity value LC is portrayed.

Three important network performance indexes are reported: the network congestion level for the no failure state (C_{S_0}), the network congestion level (C_S), i.e., the maximum network congestion level over all the states S_v, and the maximum amount of traffic (TL) that is lost in the network, due to a single link failure. The latest is expressed as a percentage of the total offered traffic. Traffic losses are encountered whenever the flow on a lightpath exceeds the lightpath capacity. For the three measures, we report the mean and the worst values obtained over ten randomly generated traffic instances.

We report results for different values of the nodal in/out degree in the logical topology (Δ), and maximum number of wavelengths on a fiber $(\mathcal{W})$.

In Tables I–IV, respectively, the results for four different values of nodal in/out degree $(\Delta = 2, 3, 4, 5)$ are portrayed.

It can be observed that the joint optimization approach in these cases outperforms disjoint optimization, especially for what concerns the maximum traffic lost because of failures. For example, in Table I, we see that with disjoint optimization the maximum lost traffic is still nonnull when the link capacity LC is 50, while, under joint-optimization, almost null losses are observed when the lightpath capacity is 30.

The difference between the two optimization procedures increases when the logical topology degree increases. Table IV shows that with joint-optimization no losses are observed for

TABLE V
COMPARISON BETWEEN FLTDP AND THE DISJOINT OPTIMIZATION OF
LTDP AND OPT-MP WITH $\Delta = 5$ FOR NETWORK 2

Parameters	Disjoint Opt (LTDP+Opt MP)						
	LC=20	LC=25	LC=30	LC=35	LC=40	LC=45	LC=50
C_{S_0} Mean	12.71	12.71	12.71	12.71	12.71	12.71	12.71
C_S Mean	20	25	30	35	38.95	41.54	42.28
TL Mean [%]	24.77	18.68	12.94	7.52	3.44	0.75	0.00
C_{S_0} wrst	14.23	14.23	14.23	14.23	14.23	14.23	14.23
C_S wrst	20.00	25.00	30.00	35.00	40.00	45.00	47.94
TL wrst [%]	35.90	27.03	19.57	13.36	8.20	3.04	0.00
Parameters	Joint Opt (FLTDP)						
	LC=20	LC=25	LC=30	LC=35	LC=40	LC=45	LC=50
C_{S_0} Mean	14.95	14.95	14.95	14.95	14.95	14.95	14.95
C_S Mean	16.08	23.51	24.76	25.81	25.81	25.81	25.81
TL Mean [%]	7.27	2.92	1.07	0.00	0.00	0.00	0.00
C_{S_0} wrst	18.07	18.07	18.07	18.07	18.07	18.07	18.07
C_S wrst	20.00	25.00	30.00	34.17	34.17	34.17	34.17
TL wrst [%]	17.30	11.68	4.31	0.00	0.00	0.00	0.00

TABLE VI
NUMBER OF WAVELENGTHS REQUIRED TO MAP THE LOGICAL TOPOLOGY

Mean W-Number	Disjoint Opt (LTDP+Opt MP)		
	$\Delta = 2$	$\Delta = 3$	$\Delta = 4$
Network 1: 10 nodes 14 links	3.2	5.0	6.1
NSF-net: 14 nodes 21 links	3.9	5.1	6.7
ARPA-net: 21 nodes 26 links	6.4	9.2	12.6
U.S.A. Long Distance: 28 nodes 45 links	7.2	11.4	14.5
Mean W-Number	Joint Opt (FLTDP)		
	$\Delta = 2$	$\Delta = 3$	$\Delta = 4$
Network 1: 10 nodes 14 links	2.5	3.8	5.1
NSF-net: 14 nodes 21 links	2.3	4.4	6.3
ARPA-net: 21 nodes 26 links	5.4	8.4	11.7
U.S.A. Long Distance: 28 nodes 45 links	7.0	9.2	12.1

configurations in which the lightpath capacity is 10, while under disjoint-optimization, losses are still registered when $LC = 25$.

Differences become even greater when the physical topology plotted in Fig. 5 is considered. In this case, trying to build a logical topology with nodal degree equal to 2, out of 20 traffic instances, ten times the disjoint optimization algorithm fails, since no mapping exists such that the logical topology remains connected under any single link failure scenario. This means that some source-destination pairs cannot communicate under some failure patterns, whichever capacity is assigned to lightpaths. In those cases, of course, the solution provided by disjoint optimization algorithms leads to unacceptable performance in terms of failure resilience. Table V reports results restricted to the ten cases in which the disjoint optimization does not fail.

Table VI instead reports a comparison between joint optimization and disjoint optimization in terms of the average required number of wavelengths. Results refer to four topologies with different numbers of nodes and links: the network 1, shown in Fig. 4 (10 nodes, 14 links), the NSF-net topology (14 nodes, 21 links), the ARPA-net topology (21 nodes, 26 links), and the USA Long Distance topology (28 nodes, 45 links). We observe that while for the 10-node topology the logical topology resulting from disjoint-optimization was obtained by applying the optimal mapping algorithm proposed in [13], for larger networks the results were obtained by performing the heuristic GDAP mapping algorithm over the outcome of the logical topology optimization procedure, because the algorithm
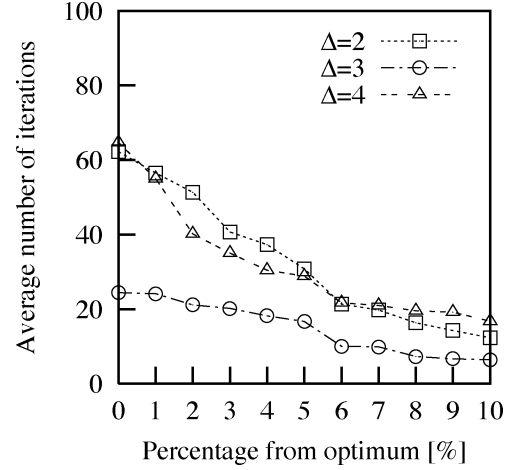

Fig. 6. Average number of iterations versus the percentage distance from the optimal solution.

of [13] is too complex for networks of this size. Results show that also in terms of required number of wavelengths, the application of the joint-optimization algorithm appears to be advantageous, yielding an average saving of about 20%.

Table VII reports the CPU time needed to run an iteration of the joint-optimization Tabu search algorithm. All results were obtained over an 800-MHz Pentium III PC running Linux 6.2.

Table VII also shows the iteration number at which the optimal solution was found; the average value over ten instances (M O-it) and the worst case (wrst O-it) value are reported. In all cases, 300 iterations were run before stopping the algorithm. We notice that only in one instance more than 100 iterations (114) were necessary to find the optimum value.

Finally, Fig. 6 reports the average number of iterations required to find a solution that differs by a given percentage from the optimum. It is worth noting that a solution that is few percentages worse than the best can be obtained in a significantly smaller number of iterations than for the best solution.

VIII. CONCLUSION AND FURTHER WORK

In this paper, we proposed a new methodology for the design of fault-tolerant logical topologies in wavelength-routed WDM networks supporting IP datagram flows.

Our approach to protection and restoration generalizes the concepts first proposed in [9]–[11], and relies on the exploitation of the intrinsic dynamic capabilities of IP routing, thus leading to cost-effective fault-tolerant logical topologies.

Our approach differs from those proposed in [9]–[11], since it considers the resilience properties of the topology during the logical topology optimization process, thus extending the optimization of the network resilience performance also on the space of logical topologies.

Several avenues are open for further work. For instance, the Tabu procedure could be improved by allowing moves that change the node degree and the mapping procedure could be refined also using a Tabu search such as in [9] and [10]. However, even without such improvements, we have found that the proposed joint optimization approach largely outperforms

TABLE VII
CPU TIMES FOR ONE ITERATION OF THE TABU SEARCH ALGORITHM

CPU TIME	$\Delta = 2$			$\Delta = 3$			$\Delta = 4$		
	1-It[sec]	M O-It	wrst O-It	1-It[sec]	M O-It	wrst O-It	1-It[sec]	M O-It	wrst O-It
Network 1	0.78	19.88	85	1.38	31.55	91	1.49	65.67	96
NSF-net	6.69	62.22	98	7.12	24.44	41	9.66	64.67	94
ARPA-net	60.46	36.22	67	47.48	43.22	91	46.98	40.78	90
U.S.A. Long Distance	246.78	36.1	60	177.20	68	114	160.43	63.5	81

the previous ones, and is able to obtain very good logical topologies with fault-tolerance properties at a limited cost.

APPENDIX

The algorithm used by TabuFLTDP to generate an initial solution is known as D-MLTDA. This heuristic initially considers a fully-connected logical topology, and sequentially removes a set of least-loaded lightpaths from the logical topology, until the degree constraints are satisfied. To describe this algorithm, we use a bipartite graph associated with the current logical topology according to the following rules: two vertices i_n and o_n in the bipartite graph correspond to each node n in the (logical) topology; in the bipartite graph, an edge exists between i_s and o_d, whose weight is initialized to the traffic flow value between nodes s and d; a Boolean variable is associated with each edge, which can assume the values Removable or Unremovable. The D-MLTDA algorithm can be described as follows.

Step 0) Select the fully-connected logical topology and mark all lightpaths as Removable.

Step 1) If all the in/out-degree constraints are satisfied GOTO Step 2, else STOP.

Step 2) Select o', the node in O closest to d.

Step 3) Solve the routing problem on the current topology and compute traffic flows on lightpaths.

Step 4) Assign to each edge of the bipartite graph a weight equal to the flow traversing the associated lightpath.

Step 5) Find a set of edges that can be removed from the graph by solving a 1-minimum weight matching[2] (1-m WM) on the bipartite graph. Only the edges that are marked as Removable can be chosen in the matching.

Step 6) Remove all edges in the 1-m WM, together with the corresponding lightpath in the logical topology, only if the resulting logical topology remains connected and GOTO Step 1. If the removal of a matched lightpath would disconnect the logical topology, mark the lightpath as Unremovable.

REFERENCES

[1] D. Bienstock and O. Gueluek, "Computational experience with a difficult mixed-integer multicommodity flow problem," *Math. Program.*, vol. 68, pp. 213–235, 1995.

[2] B. Yener and T. Boult, "A study of upper and lower bounds on minimum congestion routing in lightwave networks," in *Proc. Infocom'94*, Toronto, ON, Canada, June 1994.

[3] E. Leonardi, M. Mellia, and M. Ajmone Marsan, "Algorithms for the topology design in WDM all-optical networks," *Opt. Networks Mag.*, vol. 1, pp. 35–46, Jan. 2000.

[4] R. Ramaswami and K. N. Sivarajan, "Design of logical topologies for wavelength-routed optical networks," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 840–851, June 1996.

[5] B. Mukherjee, D. Banerjee, S. Ramamurthy, and A. Mukherjee, "Some principles for designing a wide-area WDM optical network," *IEEE/ACM Trans. Networking*, vol. 4, pp. 684–695, Oct. 1996.

[6] G. Iannaccone, C. Chuah, S. Bhattacharyya, and C. Diot, "Analysis of link failures in IP backbone," in *ACM Sigcomm Internet Measurement Workshop*, Nov. 2002.

[7] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C. Chuah, and C. Diot, "Characterization of failures in an IP backbone," in *Proc. Infocom '04*, Hong Kong, Mar. 2004.

[8] F. Giroire, A. Nucci, N. Taft, and C. Diot, "Increasing the robustness of IP backbones in the absence of optical level protection," in *Infocom '03*, San Francisco, CA, Mar. 2003.

[9] J. Armitage, O. Crochat, and J. Y. Le Boudec, "Design of a survivable WDM photonic network," in *Infocom '97*, Apr. 1997, pp. 244–252.

[10] O. Crochat and J. Y. Le Boudec, "Design protection for WDM optical networks," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1158–1165, Sept. 1998.

[11] O. Crochat, J. Y. Le Boudec, and O. Gerstel, "Protection interoperability for WDM optical networks," *IEEE/ACM Trans. Networking*, vol. 8, pp. 384–395, June 2000.

[12] M. Plante and B. Sansò, "A typology for multi-technology multi-service network synthesis methods," *Telecommun. Syst.*, vol. 19, no. 1, pp. 39–73, 2002.

[13] E. Modiano, "Survivable routing of logical topologies in WDM networks," in *Proc. Infocom*, Anchorage, AL, Apr. 2001.

[14] *http://www.ilog.com/products/cplex* [Online]

[15] B. Sansó, F. Soumis, and M. Gendreau, "On the evaluation of telecommunications network reliability using routing models," *IEEE Trans. Commun.*, vol. 39, pp. 1494–1501, Oct. 1991.

[16] B. Sansó, M. Gendreau, and F. Soumis, "An algorithm for network dimensioning under reliability considerations," *Ann. Oper. Res.*, vol. 36, pp. 263–274, 1992.

Antonio Nucci (M'99) received the Dr.Ing. degree in electronics engineering and the Ph.D. degree in telecommunications engineering from the Politecnico di Torino, Turin, Italy, in 1998 and 2002, respectively.

In 1999, he spent four months at the CRT of the Université de Montréal, Montreal, QC, Canada, where he worked with Prof. Theodor Gabriel Crainic and Prof. Brunilde Sansò. He has been a member of the IP research group at the Sprint Advanced Technology Laboratories, Burlingame, CA, since September 2001. His research interests are in traffic characterization, performance evaluation, traffic engineering, and network design.

---

[2]The 1-minimum weight matching algorithm provides the maximum size matching with minimum weight.

Brunilde Sansò (M'91) was born in Rome, Italy, in 1960. She received the E.E. degree from the Universidad Simon Bolivar, Caracas, Venezuela, in 1981, the M.S. degree in reliability and the Ph.D. degree in operations research from École Polytechnique de Montréal, Montreal, QC, Canada, in 1985 and 1988, respectively.

After Postdoctoral studies at the CRT, University of Montreal, and a Research Fellowship at the GERAD, she joined the faculty of École Polytechnique de Montréal in 1992, where she has been a Full Professor since 1997. She is currently with the Department of Electrical Engineering, where she is the Director of the LORLAB, a research laboratory devoted to the performance, reliability, design and optimization of operational planning of broadband networks. She is Co-Editor of the book *Telecommunications Network Planning* (Norwell, MA: Kluwer, 1998) and the forthcoming book *Performance and Planning Methods for the Next Generation Internet* (Norwell, MA: Kluwer).

Dr. Sansò is a recipient or co-recipient of several awards and honors, among them, the 2003 DRCN Best Paper Award, the Second Prize in the 2003 CORS Practice Competition, the 1995 IEEE/ASME JRC Best Paper Award, the 1992 NSERC Women Faculty Award, and the 1992 FCAR Young Researcher Award. She is an Associate Editor of *Telecommunication Systems* and has been a referee and technical committee member for major journals and scientific conferences, reviewer for government agencies, and industry consultant. She was the Program Co-Chair of the Fifth INFORMS Telecommunications Conference.

Emilio Leonardi (S'94–M'99) was born in Cosenza, Italy, in 1967. He received the Dr.Ing. degree in electronics engineering and the Ph.D. degree in telecommunications engineering from Politecnico di Torino, Turin, Italy, in 1991 and 1995, respectively.

He is currently an Assistant Professor at the Dipartimento di Elettronica of Politecnico di Torino. In 1995, he visited the Computer Science Department of the University of California, Los Angeles (UCLA). In summer 1999, he joined the High Speed Networks Research Group at Bell Laboratories/Lucent Technologies, Holmdel, NJ; in summer 2001, the Electrical Engineering Department of the Stanford University and finally in summer 2003, the IP Group at Sprint, Advanced Technologies Laboratories, Burlingame CA. He has participated in several national and international research projects dealing with telecommunication networks. He has co-authored over 100 papers published in international journals and presented in leading international conferences, all of them in the area of telecommunication networks. His research interests are in the field of: performance evaluation of communication networks, all-optical networks, queueing theory, packet switching.

Dr. Leonardi received the IEEE TCGN best paper award for a paper presented at the IEEE Globecom 2002, "High Speed Networks Symposium." He participated in the program committee of several conferences including: IEEE Infocom, IEEE Globecom and IEEE ICC. He was Guest Editor of two special issues of IEEE JOURNAL OF SELECTED AREAS OF COMMUNICATIONS that focused on high speed switches and routers.

Theodor Gabriel Crainic received the Ph.D. degree in operations research from the Université de Montréal, Montreal, QC, Canada, in 1982.

He is Professor of Operations Research at Ecole des sciences de la gestion of the Université du Québec à Montréal, Montréal, QC, Canada, and Director of the Intelligent Transportation Systems Laboratory of the Centre for Research on Transportation of the Université de Montréal. He is adjunct Professor at the Department of Computer Science and Operations Research of the Université de Montréal, the Molde University in Norway, and the Computer Science Department of the University of Manitoba, Winnipeg, MB, Canada. His research interests are in operations research models, network and combinatorial optimization methods, metaheuristics, and parallel computation applied to transportation and telecommunications network planning, management and operations, electronic market design, and electronic commerce-driven logistics and transportation systems. He authored or coauthored some 100 scientific articles and coauthored STAN, a method and interactive-graphic software for strategic planning of multimodal multicommodity transportation systems used in 30 organizations in 16 countries. He co-founded the TRiennial Symposium on Transportation Analysis (TRISTAN) series of international meetings and served as Associate Editor for Operations Research. He is North American Editor of the *Journal of Mathematical Modeling and Algorithms* and serves on the editorial boards of several other operations research and transportation journals.

Marco Ajmone Marsan (S'76–M'78–SM'86–F'99) received the Dr.Ing. degree in electronic engineering from Politecnico di Torino, Turin, Italy, in 1974 and the M.S. degree from the University of California, Los Angeles (UCLA), in 1978. He was awarded an "Honoris Causa" degree in telecommunications networks from the Budapest University of Technology and Economics, Budapest, Hungary, in 2002.

He is a Full Professor in the Electronics Department of Politecnico di Torino. He has been the Director of the Institute for Electronics, Information and Telecommunications Engineering of the National Research Council since September 2002. He was with the Electronics Department of Politecnico di Torino from November 1975 to October 1987, first as a researcher and then as an Associate Professor. He was a Full Professor at the University of Milan's Computer Science Department from November 1987 to October 1990. During the summers of 1980 and 1981, he was with the Research in Distributed Processing Group, Computer Science Department, UCLA. During the summer of 1998, he was an Erskine Fellow at the University of Canterbury's Computer Science Department, New Zealand. He has coauthored over 300 journal and conference papers in Communications and Computer Science, as well as the two books *Performance Models of Multiprocessor Systems* (Cambridge, MA: MIT Press) and *Modeling with Generalized Stochastic Petri Nets* (New York: Wiley). He has been the principal investigator in national and international research projects dealing with telecommunication networks. His current interests are in the performance evaluation of communication networks and their protocols.

Dr. Ajmone Marsan received the Best Paper Award at the Third International Conference on Distributed Computing Systems, Miami, FL, in 1982. He is a corresponding member of the Academy of Sciences of Torino. He participates in a number of editorial boards of international journals, including the IEEE/ACM TRANSACTIONS ON NETWORKING, and the *Computer Networks Journal* by Elsevier.