Multi-Metrics Reconfiguration in Core WDM Networks

Gurvan Huiban
MASCOTTE Team
I3S-CNRS/INRIA/Université de Nice Sophia Antipolis
Nice - France
gurvan.huiban@sophia.inria.fr

Pallab Datta
Computer and Computational Sciences Division
Los Alamos National Laboratory
Los Alamos, NM - USA
pallab@lanl.gov

***Abstract*— We consider the reconfiguration problem in multi-fiber WDM optical networks. In a network with evolving traffic, the virtual topology may not remain optimal, leading to degradation of network performance. However, adapting the virtual topology to the changing traffic may lead to service disruption. This optimization problem hence captures the trade-off between network performance and number of reconfigurations applied to the virtual topology. This trade-off is considered via a multi-metrics approach.**

The above problem is solved through a Mixed Integer Linear Programming (MILP) formulation with a multivariate objective function. However the problem is NP-hard and such an approach is unable to solve large problem instances in a reasonable time. Therefore we propose a simulated annealing (SA) based heuristic approach for solving problems of higher complexity.

We compare the performance and the computation time of solving the MILP model and the heuristic approach considering different test instances. We can find near optimal solutions for instances of medium complexity using the MILP model. The SA scheme can be used as a heuristic to arrive at near optimal solutions when the run-time of the MILP becomes practically infeasible. It also appears that the trade-off's involved in the reconfiguration problem cannot be left aside, as a little flexibility with respect to one metric allows to drastically improve the quality of the solution with respect to other metrics.

***Index Terms*— Reconfiguration, WDM networks, Optimization, MILP, Simulated annealing.**

I. INTRODUCTION

The *optical* technology, and more specifically the *Wavelength Division Multiplexing* (WDM) technology is the key component of large-scale and long-distance data transmission. Used in core networks, this technology still evolves and offers capacity to cope with the ever increasing traffic demand [1].

Installing a large-scale telecommunication network is expensive. For instance the cost of a North-American network covering 15 cities was estimated to 200 millions dollars [2]. An important part of the expense comes from the infrastructure operations: digging and installing cables. The thinness of an optical fiber allows a single cable to contain tens of fibers. Consequently companies generally install many optical fibers at the same time, even if it is not required, resulting in multifiber networks.

The bandwidth offered by optical technology is very high, and one limitation comes from the electronic part of the transmission scheme. The devices that are able to deal with tens of Gigabits per seconds are very expensive. The optical technology makes possible the definition of an optical layer over a physical layer, called *virtual topology* or *logical topology*, where no electronic processing is performed. It is defined through the configuration of the optical cross-connects (OXC) installed in each network node and can be modified [3].

Since the logical topology represents the effective communication graph, the virtual topology has a direct impact on the data traffic routing efficiency. From this point of view, each modification in the traffic pattern should trigger a redefinition of the virtual topology. However, changing the virtual topology means modifying the configuration of the OXC's, which may lead to total or partial network disruption [4].

The virtual topology is constituted of *lightpaths*. A lightpath is a path on the physical topology, and it corresponds to a link in the virtual topology. A lightpath is established using the same wavelength from the source to the demand, thus obeying the *wavelength continuity constraint*. If the network is equipped with wavelength converters, such restriction can be avoided. However the technology for optical wavelength conversion is not yet matured enough for most commercial applications. Optical-electronic-optical converters are still very expensive and generate delay in the overall transmission time.

In the present work we consider core optical networks. Such networks carry highly aggregated traffic coming from the various subnetworks connected through the core network. It provides a certain stability in the traffic and the evolutions, if any, occur slowly and smoothly, and thus can be discretized. Other consequences of the traffic aggregation, is that each node sends data to each other node and receive data from every other nodes. Moreover, the traffic evolution is quite predictable, since it follows large-scale trends. Such assumptions are reasonable in a core network.

To summarize, we consider multifiber WDM networks. The virtual topology is constituted of lightpaths obeying the wavelength continuity constraints. The traffic follows a all-to-all pattern. The traffic evolutions are discretized.

We deal with the *network reconfiguration problem*. It is an extension of the static *Routing and Wavelength Assignment* (RWA) problem, which is proven to be a NP-hard [5]. Hence the reconfiguration problem is also NP-hard. Remember that the static RWA problem can be stated as, given a network and a traffic matrix, we need to determine the logical topology to be imposed on the physical topology, hence routing the lightpaths

This work has been partially funded by a grant from CNPQ, by a grant from French PACA province, by a Lavoisier grant from the French "Ministère des Affaires Étrangères", by ACI-SI PRESTO and by European project IST FET AEOLUS. This manuscript is also available as Los Alamos Technical Report LA-UR-07-3547.

over the physical topology and assigning a wavelength to each lightpath [6]. At the same time the RWA problem is solved, it is common to route the packet traffic over the logical topology obtained.

In the reconfiguration problem, we consider a physical topology and a succession of traffic matrices. For each traffic matrix, we solve the associated RWA problem and we route the packet traffic, obtaining a succession of virtual topologies and packet routings. The main difference between a succession of RWA problems and the reconfiguration problem is that the latter takes into consideration the fact it will be required to switch from a virtual topology and the associated routing to the next one. As mentioned earlier, such operation may generate network disruption, which is not desired. When two virtual topologies are similar, it is easy to switch from one to the other. The reconfiguration problem involves a trade-off between the virtual topology quality and the network disruption that may occur each time it is re-defined.

The network reconfiguration problem is well known in literature. However, there are not too many satisfying methods to solve it. Some works restrict themselves to very-specific cases. In [7], the authors develop reconfiguration algorithm for ring networks. The proposed algorithm is based on branch-exchange techniques. In [4] a Markovian process is used to study the trade-offs involved in reconfiguration in single-hop broadcast WDM networks. The work in [8], [9], [10] considers the reconfiguration problem in the case of a unique traffic evolution, and not as a succession of traffic evolutions, and use as input an existing virtual topology and routing.

Different Mixed Integer Linear Programming (MILP) approaches has been developed in literature which addresses short-term, mid-term and long-term network reconfiguration issues with respect to evolving traffic in WDM optical networks [11], [12], [13].

The problem has been also addressed in [14], [15] under the context of "dynamic traffic grooming". In both works, the authors modify the initial network graph. The modifications consist of splitting nodes to represent different part of the optical devices (electronic processing, purely optical router, and so on). That allows to use quite simple algorithms based on the shortest path [14] solving the problem with elegant mathematical models [15]. However, these works focus on the grooming aspect and does not consider the adaptation of the virtual topology to meet the ever increasing traffic demands across multiple periods of network evolution.

Our approach takes into account the trade-off between the network configuration quality and the network disruption, which is generally absent in the articles found in the literature. Since this trade-off is the essence of the reconfiguration problem, we believe it cannot be left aside. The practical objective of this work is to make the best virtual topology reconfigurations in relation to predicted traffic evolution.

We present a *Mixed Integer Linear Programming* (MILP) formulation to solve this problem, similar to the approaches developed in [16]. However, the MILP approach is unable to solve large network instances within reasonable time limits. Meta-heuristics, such as the *simulated annealing* (SA), are generally able to find good solutions to optimization problems for an affordable computational cost [17]. As we compare the results obtained with a MILP approach and a SA approach, we can use the information provided by the solver to make a pertinent evaluation of the SA performance.

This paper is organized as follows: We describe formally the problem solved in section II. In section III, the MILP model is developed for solving the virtual topology reconfiguration problem. We describe the simulated annealing heuristic algorithm in section IV. Some experimental results are given and analyzed in section V. We finally conclude the paper in section VI.

II. Problem description and notations

We consider a WDM network $\mathcal{P}$, constituted of a set of $\mathcal{N}$ nodes and a set of $\mathcal{L}$ links. The maximum number of fibers between nodes n_1 to node n_2 is given by $\mathcal{F}_{(n_1,n_2)}$. Each fiber carries a maximum number of $\mathcal{W}$ wavelengths. Each wavelength has a maximum capacity of $\mathcal{C}$ Mbps. We assume that $\mathcal{W}$ and $\mathcal{C}$ are the same throughout the entire network. Many technological parameters (range of frequency used, type of optical fiber, and so on) are involved, and we believe that few telecommunication providers would build heterogeneous networks.

Each node routes the lightpaths without any restrictions. However, two distinct lightpaths cannot use the same wavelength in an optical fiber. Lightpaths are set up between the end points of the demands. In our network model, we allow multihop packet routing, i.e traffic from A to B can first use a lightpath from A to C, and then another lightpath from C to B. We call a *time period* the period of time between two traffic evolutions. In other words, the overall time window is divided into $\mathcal{T}$ periods $t_1, \ldots, t_{\mathcal{T}}$, and data changes occur each time a time period ends and another begins. The traffic remains constant during a whole time period. We note $\mathcal{D}_{s,d}(t)$, expressed in Mbps, as the demand for the source-demand pair $(s, d) \in \mathcal{N}^2$ during time period t.

The traffic is known *a priori*. Our objective is to find a virtual topology that is adapted to the traffic for each time period. We solve the problem keeping in mind the trade-off's involved in the reconfiguration problem mentioned above. Since the traffic matrices are known in advance, we focus on minimizing the amount of network resources but simultaneously intend to reduce the number of reconfigurations needed in the virtual topology. The resources can be measured by the number of optical links or the number of lightpaths used. The former metric is usual and represents directly the load of the network. The latter one represents the number of lightpaths required to implement the defined virtual topology. For each used lightpath, a transmitter and a receiver is required. This has a direct influence on the cost of the network nodes.

A solution S corresponds to a set of virtual topologies; one for each time period. Let us call P_S the sum of the number of used optical links and L_S the sum of the number of used lightpaths for each time period. We also take into account the number of reconfigurations ΔP_S incurred by a solution.

The objective function used to reflect this trade-off between minimal network resource usage and minimal network reconfiguration is given by (1) if the network resources are measured as number of optical links, and by (2) if the resources are measured in terms of number of lightpaths.

$$F_S = \beta_P P_S + \beta_{\Delta P} \Delta P_S \quad (1)$$
$$F_S = \beta_L L_S + \beta_{\Delta P} \Delta P_S \quad (2)$$

where β_P, β_L and $\beta_{\Delta P}$ are parameters that allows to create a multi-variate objective function. Depending on the value of each parameter, more weight is given to one or another aspect.

III. MILP MODEL

We present a MILP model for the reconfiguration problem. With such a model, the reconfiguration problem is seen as a succession of flow problems - one flow problem for each time period - coupled by reconfiguration constraints.

We tried to come up with the most concise model possible. To do so, we aggregated all commodities from a given node. This led us to a source formulation of the reconfiguration problem. Such source formulation, already used in a virtual topology design problem in [18], significantly reduces the memory occupancy overhead while solving the problem.

We define the following variables:

- $p^i_{(m,n),w}(t) \in \mathbb{N}$ is the number of optical links of wavelength w used by lightpaths having node i as source on physical link $(m,n) \in \mathcal{L}$ during time period t.
- $l^{(i,j)}_w(t)$ is the number of lightpaths from node i to node j using wavelength w during time period t.
- $l^{(i,j)}(t)$ is the number of lightpaths from node i to node j during time period t.
- $f^s_{(i,j)}(t)$ is the flow from source s using lightpath (i,j) during time period t.
- $\delta p^i_{(m,n),w}(t)$ is the number of changes for the number of optical links of wavelength w used by lightpaths having node i as a source on physical link $(m,n) \in \mathcal{L}$, between time period $t-1$ and t.

The variables $p^i_{(m,n),w}(t)$ have to be integer, but the other ones will have integer values at the end of the optimization process. The number of integer variables is $O\left(|\mathcal{N}|^3 \mathcal{W}\mathcal{T}\right)$, and the number of continuous variables is $O\left(\left(|\mathcal{N}^2|\mathcal{W} + |\mathcal{N}|^3\right)\mathcal{T}\right)$. Even for small networks and considering only a few time periods, the program generates thousands of integer variables and constraints, thus making it infeasible to solve large problem instances.

A. Virtual topology constraints

The constraints associated with the virtual topology design problem are the following:

$$\sum_{(i,n)\in\mathcal{L}} \sum_{w=1}^{\mathcal{W}} p^i_{(i,n),w}(t) = \sum_{j\in\mathcal{N}} l^{(i,j)}(t), \quad \begin{array}{l}\forall i \in \mathcal{N}\\ 1 \leqslant t \leqslant \mathcal{T}\end{array} \quad (3)$$

$$\sum_{(m,n)\in\mathcal{L}} p^i_{(m,n),w}(t) - \sum_{(n,p)\in\mathcal{L}} p^i_{(n,p),w}(t) = l^{(i,n)}_w(t), \quad \begin{array}{l}\forall i,n \in \mathcal{N}^2\\ i \neq n\\ 1 \leqslant w \leqslant \mathcal{W}\\ 1 \leqslant t \leqslant \mathcal{T}\end{array} \quad (4)$$

$$\sum_{w=1}^{\mathcal{W}} l^{(i,j)}_w(t) = l^{(i,j)}(t), \quad \begin{array}{l}\forall i,j \in \mathcal{N}^2\\ i \neq j\\ 1 \leqslant t \leqslant \mathcal{T}\end{array} \quad (5)$$

$$\sum_{i\in\mathcal{N}, i\neq n} p^i_{(m,n),w}(t) \leqslant \mathcal{F}_{(m,n)}, \quad \begin{array}{l}\forall (m,n) \in \mathcal{L}\\ 1 \leqslant w \leqslant \mathcal{W}\\ 1 \leqslant t \leqslant \mathcal{T}\end{array} \quad (6)$$

Constraints (3) corresponds to the flow conservation for each source node i: It corresponds to the sum of all lightpaths having node i as source. Constraints (4) corresponds to the flow conservation in demand nodes n, for each wavelength: The difference between the number of lightpaths entering node n and the number of lightpaths leaving node n corresponds to the number of lightpaths having n as endpoint. Constraints (5) corresponds to the number of lightpath conservation. Constraints (6) corresponds to the capacity limitation: It is not possible to have more lightpaths using a given wavelength than there are fibers installed on a given link.

B. Routing constraints

$$\sum_{j\in\mathcal{N}, j\neq s} f^s_{(s,j)}(t) = \sum_{d\in\mathcal{N}, d\neq s} \mathcal{D}_{s,d}(t), \ \forall s \in \mathcal{N}, 1 \leqslant t \leqslant \mathcal{T} \quad (7)$$

$$\sum_{i\in\mathcal{N}, i\neq s} f^s_{(i,k)}(t) - \sum_{j\in\mathcal{N}, j\neq s} f^s_{(k,j)}(t) = \mathcal{D}_{s,k}(t), \quad \begin{array}{l}\forall (s,k) \in \mathcal{N}^2\\ k \neq s\\ 1 \leqslant t \leqslant \mathcal{T}\end{array} \quad (8)$$

$$\sum_{s\in\mathcal{N}, s\neq j} f^s_{(i,j)}(t) \leqslant \mathcal{C} \sum_{w=1}^{\mathcal{W}} l^{(i,j)}_w(t), \quad \begin{array}{l}\forall (i,j) \in \mathcal{N}^2\\ 1 \leqslant t \leqslant \mathcal{T}\end{array} \quad (9)$$

Constraints (7) corresponds to the flow conservation at source node s: The sum of the flow leaving node s corresponds to the sum of the demands having s as source. Constraints (8) corresponds to flow conservation at demand nodes k: The flow entering node k minus the flow leaving node k corresponds to the demands to k. Finally, (9) is the lightpath capacity limitation.

C. Reconfiguration constraints

$$p^i_{(m,n),w}(t) - p^i_{(m,n),w}(t-1) \leqslant \delta p^i_{(m,n),w}(t), \quad \begin{array}{l}\forall i \in \mathcal{N}\\ (m,n) \in \mathcal{L}\\ i \neq n\\ 1 \leqslant w \leqslant \mathcal{W}\\ 2 \leqslant t \leqslant \mathcal{T}\end{array} \quad (10)$$

$$p^i_{(m,n),w}(t-1) - p^i_{(m,n),w}(t) \leqslant \delta p^i_{(m,n),w}(t), \quad \begin{array}{l}\forall i \in \mathcal{N}\\ (m,n) \in \mathcal{L}\\ i \neq n\\ 1 \leqslant w \leqslant \mathcal{W}\\ 2 \leqslant t \leqslant \mathcal{T}\end{array} \quad (11)$$

We consider that each variation of the allocation variables (the $p^i_{(m,n),w}(t)$ variable) from one time period to another is a change of the virtual topology. Hence, it has to be taken into account. This is done by (10) and (11).

D. Objective functions

$$P(t) = \sum_{i\in\mathcal{N}} \sum_{(m,n)\in\mathcal{L}} \sum_{w=1}^{\mathcal{W}} p^i_{(m,n),w}(t),\ 1 \leqslant t \leqslant \mathcal{T} \quad (12)$$

$$L(t) = \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}} l^{(i,j)}(t),\ 1 \leqslant t \leqslant \mathcal{T} \quad (13)$$

$$\Delta P(t) = \sum_{i\in\mathcal{N}} \sum_{(m,n)\in\mathcal{L}} \sum_{w=1}^{\mathcal{W}} \delta p^i_{(m,n),w}(t),\ 2 \leqslant t \leqslant \mathcal{T} \quad (14)$$

Equation (12) computes the overall number of used optical links. Equation (13) computes the overall number of defined lightpaths. The overall number of changes is given by (14).

Consequently, the objective function of our optimization model is the following:

$$\min F = \beta_P \sum_{t=1}^{\mathcal{T}} P(t) + \beta_L \sum_{t=1}^{\mathcal{T}} L(t) + \beta_{\Delta P} \sum_{t=2}^{\mathcal{T}} \Delta P(t) \quad (15)$$

IV. Simulated annealing algorithm

Simulated annealing is a Monte Carlo metaheuristic for minimizing multivariate functions [19]. It develops an analogy between optimization and statistical mechanics, which is the central discipline of condensed matter physics. When a system temperature decreases, the behavior of atoms is a major concern in statistical mechanics. Whether the matter will solidify as a crystal or as a glass not only depends on the temperature, but also on the way the temperature is decreased. Decreasing the temperature too quickly will lead to a crystal with many defects or a glass with no crystalline order and only local optimal structure.

Finding the best low-temperature state of a matter is similar to search for a local optimal solution of an optimization problem. A temperature for the system is defined. The algorithm progresses by lowering gradually this temperature until the system freezes. At each temperature, a large number of different solutions for the problem is computed, allowing the system to reach a steady state. This process is called *thermalization*.

The system is initialized with a particular configuration. Each new solution is constructed by imposing a displacement. If the energy of this new state is lower than the previous one, this new solution is kept. If not, this new solution is accepted with a given probability. The acceptance probability decreases with the temperature of the system, allowing to explore large portions of the solution space at the beginning of the process. As the temperature decreases, the probability of accepting a bad solution decreases, leading to a local search converging towards the nearest local optima. The probability of acceptance is generally given by $\rho = \exp^{-\delta/KT}$, where K is the Boltzmann's constant, T the temperature and δ the solution variation or the variation between two successive solution states. With the execution of the algorithm, the temperature decreases, leading to a more stable system.

There are different possible annealing schemes to update the temperature T. We may use an annealing scheme where the temperature varies as $T_n = \alpha \times T_{n-1}$, where T_n is the temperature at the n^{th} temperature update, and α is an arbitrary constant between 0 and 1. The parameter α decides how slowly T decreases. Typical values of α lie between 0.9 and 0.95. The parameter α and the value of T_0, the initial value, plays a critical role for the performance of the SA. An annealing scheme where the temperature update is made as $T_n = T_0/(1 + \alpha \times T_{n-1})$ can also be defined. We choose to use this update scheme. The typical values of α can be of the order of 0.01 to 0.1 to have a graceful degradation of the temperature. We call *transition* the fact that the temperature decreases, and *sub-transition* each time a problem is solved without any modification of the temperature.

We associate to each link $e = (n_1, n_2)$ of the network a weight ω_e, creating the link weight vector Ω. Depending on the weights, different routes will be found by the shortest path algorithm. The weights of the edges are mutated by a factor γ between each transition of the SA algorithm.

The SA algorithm is given by Algorithm 1. Our algorithm transforms the set of traffic matrices into an ordered list of requests, and then assign resources to each request. The `Solve` algorithm used to generate solution is given by Algorithm 2.

Algorithm 1 Simulated annealing for the Reconfiguration problem

Initialize an empty ordered list of requests $\mathcal{R}$
{Transformation of the demands into a set of requests}
for $\forall i, j, t$ **do**
  Add to $\mathcal{R}$ $\lfloor \frac{\mathcal{D}_{i,j}(t)}{\mathcal{C}} \rfloor$ requests of size $\mathcal{C}$ and one request with the remaining traffic (lower than $\mathcal{C}$)
end for
Initialize the link weight vector Ω to 1
Initialize temperature T_0
Compute the initial solution: S^* = `Solve`($\mathcal{P}, \Omega, \mathcal{R}$)
for Y transitions **do**
  for X sub-transitions **do**
    Evaluate the hop number h_r of each request $r \in \mathcal{R}$
    Reorder the requests $r \in \mathcal{R}$ by decreasing h_r
    S = `Solve`($\mathcal{P}, \Omega, \mathcal{R}$)
    if Compute $F_S < F_{S^*}$ **then**
      $S^* = S$ (update the best solution found)
    else
      $S^* = S$ with a probability of $e^{-\frac{\delta}{KT_n}}$
    end if
  end for
  Update the link weights with $\omega_l = \omega_l\,(1-\gamma)\,, \forall l \in \mathcal{L}$
  Scale down temperature: $T_{n+1} = \frac{T_0}{1+\alpha T_n}$
end for

The complexity of the `Solve` algorithm is $O\left(|\mathcal{N}|^4 \mathcal{W}\mathcal{T}\right)$. Thus the overall complexity of the SA algorithm is $O\left(|\mathcal{N}|^4 \mathcal{W}\mathcal{T}XY\right)$.

Algorithm 2 `Solve(`$\mathcal{P}$`,`Ω`,`$\mathcal{R}$`)` algorithm

Require: A network $\mathcal{P}$, a link weight vector Ω and an ordered list of requests $\mathcal{R}$
  for all request $r \in \mathcal{R}$ **do**
    Let s_r, d_r, t_r and v_r be respectively the source node, the demand node, the time period and the size of r
    if $v_r = \mathcal{C}$ **then**
      Find the shortest path from s_r to d_r considering the wavelengths available during time period t_r. The cost of a link corresponds to its weight.
      Make wavelengths allocation avoiding wavelength changes
      Update available wavelengths for the time period t_r
    else
      if Exist paths p_r from s_r to d_r at time period t_r using only available capacity within the lightpaths able to transport a request of size v_r **then**
        Use the shortest of the possible p_r
      else
        Find the shortest path from s_r to d_r considering the wavelengths available during time period t_r. The cost of a link corresponds to its weight.
        Make wavelengths allocation avoiding wavelength changes
      end if
    end if
    Update the available capacity in used links
    Update the used link weights with remaining capacity: $\omega_l = \omega_l * v_i$
  end for
Ensure: A virtual topology for each time period

V. Experimental results

A. Experimental parameters

We performed experimental simulations on a hypothetical small network SN2, represented on figure 1, on the COST239 network [20], represented on figure 2, on the NSFNET network, represented on figure 3, and on N20, N30 and N40 topologies, having respectively 20 nodes and 68 edges, 30 nodes and 160 edges, 40 nodes and 240 edges.

We consider the following parameters: For all existing links (n_1, n_2), $\mathcal{F}_{(n_1,n_2)} = 5$, and we solve each instance considering $\mathcal{W} = 8$, $\mathcal{C} = 40$Mbps and $\mathcal{W} = 16$, $\mathcal{C} = 20$Mbps.

The traffic is generated the following way: we first generate an initial traffic matrix. The initial demand from a node n_1 to a node n_2 is randomly chosen between 20 and 60Mbps. We then compute the evolution of the demand for each time period, based on the value of the demand at previous time period. This evolution is between -10 (that is, the traffic can decrease) and +10 Mbps (increasing traffic). For instance, it is possible to have the following evolution of traffic from node A to node B, over five time period: 57Mbps, 67Mbps, 75Mbps, 68Mbps 77Mbps.

The MILP model is solved using the commercial software Cplex[1] version 9, on a desktop PC with one gigabyte of RAM. We limit the computation time of our tests to ten hours in the vast majority of our experiments. When the solver hit the time-limit, its search process is interrupted and it returns the best solution found. This solution may not be the optimal and the solver provides a solution gap giving the maximum relative difference between the solution returned and the theoretical optimal solution.

For the SA experiments, the total number of sub-transitions at a given temperature is chosen between 10-15 and the transitions across different temperatures is considered to be between 30-40 based on the size of the demand sets. These numbers are chosen empirically. They are moderate but allows the SA algorithm to explore a solution space large enough to find good solutions. Further increasing of these numbers didn't yield any better results.

The parameters related to the Simulated Annealing algorithm were empirically chosen. The K Constant was chosen

[1]Copyright ©Ilog 1997-2004. Cplex is a registered trademark of Ilog.

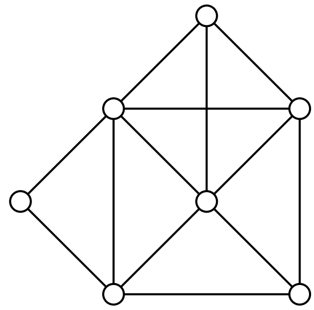

Fig. 1. Small network 2 (SN2)

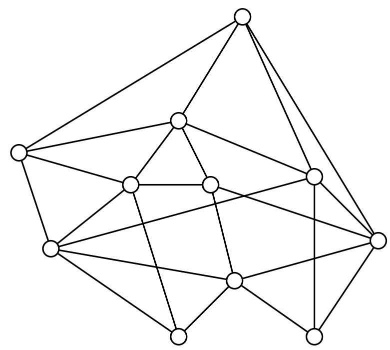

Fig. 2. COST239 network

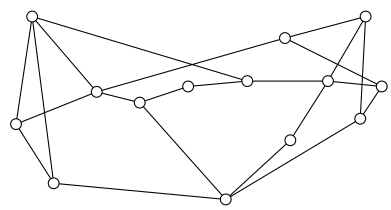

Fig. 3. NSFNET network

such that, $0 \leq \exp^{-\delta/(K \times t_i)} \leq 1$ where t_i is the temperature at the i^{th} iteration. The temperature mutation parameter α is taken to be 0.005 so that that the temperature does not drop abruptly. Higher values of α leads to a fast convergence for the SA procedure. We mutated the values of α so that the SA procedure explores the maximal possible solution states, and shows no further improvements. The edge weight mutation parameter γ was chosen to be between 0.5 and 1.0.

B. Performance Analysis

The most representative results we obtain with the MILP approach and the SA are reported in tables I to VIII. The $^+$symbol means that the solver hits the time-limit after having found at least a possible solution. In this case, it returns the best solution found. The ^{0}symbol means that the solver hit the time limit without finding any solution. We do not report the results obtained with the MILP for N30 an N40, since the solver aborts the process before its end due to lack of memory.

TABLE I

SN2 NETWORK, $\mathcal{W} = 8$, SIMULATION TIME

SN2 network, $\mathcal{W} = 8$			
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)	Gap(%)
MILP			
$(1,0,0)$	$(312, 292, 452)$	36241^+	4.79
$(0,1,0)$	$(409, 239, 613)$	36173^+	3.59
$(0,0,1)$	$(1190, 875, 0)$	15.75	0
$(1,0,1)$	$(318, 304, 5)$	36135^+	5.05
$(0,1,1)$	$(5137, 250, 8)$	36060^+	8.1
Simulated annealing			
$(1,0,0)$	$(307, 297, 454)$	234	
$(0,1,0)$	$(392, 246, 606)$	298	
$(0,0,1)$	$(1228, 896, 1)$	18	
$(1,0,1)$	$(324, 307, 5)$	302	
$(0,1,1)$	$(5188, 253, 8)$	337	

TABLE II

SN2 NETWORK, $\mathcal{W} = 16$, SIMULATION TIME

SN2 network, $\mathcal{W} = 16$			
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)	Gap(%)
MILP			
$(1,0,0)$	$(603, 542, 885)$	36333^+	1.63
$(0,1,0)$	$(691, 447, 1040)$	36238^+	1.66
$(0,0,1)$	$(1900, 1440, 0)$	54.44	0
$(1,0,1)$	$(611, 544, 11)$	36110^+	1.58
$(0,1,1)$	$(953, 459, 5)$	36045^+	2.07
Simulated annealing			
$(1,0,0)$	$(599, 556, 891)$	304	
$(0,1,0)$	$(682, 469, 1043)$	331	
$(0,0,1)$	$(1945, 1523, 4)$	41	
$(1,0,1)$	$(616, 584, 13)$	358	
$(0,1,1)$	$(874, 533, 6)$	390	

The computation time depends on the instance size - which is expected - but also on the metric used. Mixing the resource and reconfiguration objectives turns the problem much more difficult to solve than considering a single objective. This can be observed for both the MILP and for the SA approach.

We had difficulties to solve the MILP model within a bounded time limit even for small network instances. For some

TABLE III

COST239 NETWORK, $\mathcal{W} = 8$, SIMULATION TIME

COST239 network, $\mathcal{W} = 8$			
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)	Gap(%)
MILP			
$(1,0,0)$	$(944, 780, 1308)$	36309^+	1.90
$(0,1,0)$	$(1179, 591, 1698)$	36086^+	3.24
$(0,0,1)$	$(4220, 2335, 0)$	767.8	0
$(1,0,1)$	$(-,-,-)$	36171^0	-
$(0,1,1)$	$(-,-,-)$	36193^0	-
Simulated annealing			
$(1,0,0)$	$(933, 784, 1331)$	642	
$(0,1,0)$	$(1119, 585, 1702)$	620	
$(0,0,1)$	$(4304, 2389, 1)$	221	
$(1,0,1)$	$(1136, 884, 1304)$	600	
$(0,1,1)$	$(1236, 778, 8)$	443	

TABLE IV

COST239 NETWORK, $\mathcal{W} = 16$, SIMULATION TIME

COST239 network, $\mathcal{W} = 16$			
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)	Gap(%)
MILP			
$(1,0,0)$	$(1871, 1406, 2744)$	36299^+	1.00
$(0,1,0)$	$(2252, 1110, 3364)$	36093^+	1.14
$(0,0,1)$	$(5215, 3305, 0)$	2243	0
$(1,0,1)$	$(-,-,-)$	36034^0	-
$(0,1,1)$	$(-,-,-)$	36047^0	-
Simulated annealing			
$(1,0,0)$	$(2102, 1410, 2473)$	800	
$(0,1,0)$	$(1304, 1194, 3001)$	813	
$(0,0,1)$	$(5367, 3381, 8)$	340	
$(1,0,1)$	$(1069, 863, 11)$	832	
$(0,1,1)$	$(1322, 699, 12)$	849	

TABLE V

NSFNET NETWORK, $\mathcal{W} = 8$, SIMULATION TIME

NSFNET network, $\mathcal{W} = 8$			
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)	Gap(%)
MILP			
$(1,0,0)$	$(1814, 1310, 2379)$	36191^+	1.85
$(0,1,0)$	$(2559, 965, 3669)$	36054^+	2.87
$(0,0,1)$	$(8100, 3575, 0)$	8100	0
$(1,0,1)$	$(1883, 1249, 12)$	36098^+	4.82
$(0,1,1)$	$(-,-,-)$	36191^0	-
Simulated annealing			
$(1,0,0)$	$(1791, 1343, 2345)$	1023	
$(0,1,0)$	$(2512, 968, 3666)$	1088	
$(0,0,1)$	$(8168, 3601, 1)$	1496	
$(1,0,1)$	$(1798, 1210, 11)$	1045	
$(0,1,1)$	$(1801, 1306, 2279)$	1292	

TABLE VI

NSFNET NETWORK, $\mathcal{W} = 16$, SIMULATION TIME

NSFNET network, $\mathcal{W} = 16$			
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)	Gap(%)
MILP			
$(1,0,0)$	$(3588, 2365, 4967)$	36262^+	0.75
$(0,1,0)$	$(4038, 1810, 5912)$	36060^+	0.69
$(0,0,1)$	$(-,-,-)$	36065^0	-
$(1,0,1)$	$(-,-,-)$	36040^0	-
$(0,1,1)$	$(-,-,-)$	36050^0	-
Simulated annealing			
$(1,0,0)$	$(3601, 2310, 5107)$	2044	
$(0,1,0)$	$(4047, 1806, 5934)$	2181	
$(0,0,1)$	$(9106, 4408, 3)$	2102	
$(1,0,1)$	$(1914, 1223, 11)$	2017	
$(0,1,1)$	$(1943, 1405, 2406)$	2118	

TABLE VII
N20 NETWORK, $\mathcal{W} = 8$, SIMULATION TIME

N20 network, $\mathcal{W} = 8$			
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)	Gap(%)
MILP			
$(1, 0, 0)$	$(2655, 1719, 2622)$	86799^+	1.34
$(0, 1, 0)$	$(-, -, -)$	36058^0	-
$(0, 0, 1)$	$(7782, 3339, 0)$	12236	0
$(1, 0, 1)$	$(-, -, -)$	86467^0	-
$(0, 1, 1)$	$(-, -, -)$	86464^0	-
Simulated annealing			
$(1, 0, 0)$	$(2682, 1668, 2534)$	3145	
$(0, 1, 0)$	$(3189, 1677, 2872)$	3089	
$(0, 0, 1)$	$(7810, 3409, 2)$	3278	
$(1, 0, 1)$	$(2694, 1745, 2676)$	3127	
$(0, 1, 1)$	$(2781, 1783, 2760)$	3112	

TABLE VIII
N20 NETWORK, $\mathcal{W} = 16$, SIMULATION TIME

N20 network, $\mathcal{W} = 16$			
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)	Gap(%)
MILP			
$(1, 0, 0)$	$(5266, 3139, 5613)$	86690^+	0.53
$(0, 1, 0)$	$(-, -, -)$	36036^0	-
$(0, 0, 1)$	$(-, -, -)$	36100^0	-
$(1, 0, 1)$	$(-, -, -)$	86444^0	-
$(0, 1, 1)$	$(-, -, -)$	86478^0	-
Simulated annealing			
$(1, 0, 0)$	$(5307, 3166, 5575)$	3578	
$(0, 1, 0)$	$(5489, 3105, 5891)$	3610	
$(0, 0, 1)$	$(15043, 6465, 4)$	3888	
$(1, 0, 1)$	$(5038, 3104, 5428)$	3619	
$(0, 1, 1)$	$(5120, 3165, 5603)$	3624	

TABLE IX
N30 NETWORK, $\mathcal{W} = 8$, SIMULATION TIME

N30 network, $\mathcal{W} = 8$		
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)
$(1, 0, 0)$	$(3019, 2018, 2834)$	3452
$(0, 1, 0)$	$(3488, 2011, 3109)$	3312
$(0, 0, 1)$	$(8311, 3672, 6)$	3441
$(1, 0, 1)$	$(3096, 2113, 2901)$	3376
$(0, 1, 1)$	$(3200, 2103, 2944)$	3312

TABLE X
N30 NETWORK, $\mathcal{W} = 16$, SIMULATION TIME

N30 network, $\mathcal{W} = 16$		
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)
$(1, 0, 0)$	$(5987, 3976, 5811)$	3609
$(0, 1, 0)$	$(6759, 4000, 6079)$	3549
$(0, 0, 1)$	$(15835, 7188, 11)$	3456
$(1, 0, 1)$	$(6192, 4014, 5689)$	3599
$(0, 1, 1)$	$(6371, 4218, 5763)$	3617

TABLE XI
N40 NETWORK, $\mathcal{W} = 8$, SIMULATION TIME

N40 network, $\mathcal{W} = 8$		
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)
$(1, 0, 0)$	$(3141, 2231, 2984)$	3588
$(0, 1, 0)$	$(3790, 2467, 3451)$	3455
$(0, 0, 1)$	$(8671, 4005, 9)$	3491
$(1, 0, 1)$	$(3134, 2345, 3024)$	3502
$(0, 1, 1)$	$(3410, 2409, 3310)$	3498

TABLE XII
N40 NETWORK, $\mathcal{W} = 16$, SIMULATION TIME

N40 network, $\mathcal{W} = 8$		
$(\beta_P, \beta_L, \beta_{\Delta P})$	$(P, L, \Delta P)$	Comp. time (s)
$(1, 0, 0)$	$(6018, 4151, 5567)$	3718
$(0, 1, 0)$	$(7211, 4879, 6671)$	3655
$(0, 0, 1)$	$(16093, 8173, 15)$	3722
$(1, 0, 1)$	$(6094, 4487, 5831)$	3708
$(0, 1, 1)$	$(6652, 4674, 6598)$	3698

of the larger instances, the solver is unable to find even one feasible solution. Solving the problem with the SA always returns a solution, even for larger problem instances.

When the solver returns a non-optimal solution, the obtained solution gap is quite low (between 0 and 5%). The SA algorithm returns solutions which are very close to the solution found by the solver for a computation time significantly lower. Sometimes the solutions found by the heuristic are even better than the solutions obtained by the solver after a ten hours computation, due to the non-optimality of these solutions.

If we optimize only one metric, the results obtained with the other metrics are generally very bad. On the other hand, if we consider the performance-reconfiguration trade-off, that is using at the same time two metrics, we obtain good solutions in relation with both metrics, even if neither is optimal. For instance, with the SN2 network ($\mathcal{W} = 16$), minimizing objective function P gives a solution using 603 optical links, but triggering 885 reconfigurations. On the other hand, the solution obtained minimizing ΔP uses 1900 optical links and triggers 0 reconfigurations. Mixing the objectives with an equal weight, we obtain a solution using 611 optical links and triggering only 11 reconfigurations. A little flexibility with respect to one metric allows to drastically improve the quality of the solution with respect to other metrics.

We also compare the solutions provided by both the MILP and the SA approach, given a specified amount of time (see table XIII for some examples). To do so we measure the amount of time the SA required and run the solver with this amount of time as time-limit. When the solver succeeds in finding a solution, which is only the case for small instances, it finds good quality solutions that compete with the ones obtained by the SA algorithm. It confirms the fact that the solver succeeds in finding a good solution quite easily. The difficult task is, eventually to find the optimal solution, and to prove its optimality. When we attempt to solve relatively large problem instances, the solver is unable to find any solution in such a short time.

VI. CONCLUSION

In this paper we present two approaches for minimizing the virtual topology reconfiguration cost and optimizing the network quality in a network with evolving traffic across multiple time periods. Our multi-metrics approach allows to deal with the trade-off between the reconfiguration cost and the solution quality by considering at the same time two objectives. We present a MILP model and an heuristic based

TABLE XIII

COMPARISON: GIVEN COMPUTATION TIME REQUIRED BY THE SIMULATED ANNEALING

instance	MILP	simulated annealing
sn2_8, (1,0,0)	316	307
sn2_8, (0,1,0)	241	246
sn2_8, (0,0,1)	0	1
sn2_8, (1,0,1)	-	329
sn2_8, (0,1,1)	-	261
COST239_16 (1,0,0)	1896	2102
COST239_16 (0,1,0)	1115	1305
COST239_16 (0,0,1)	-	8
COST239_16 (1,0,1)	-	1080
COST239_16 (0,1,1)	-	711
NSFNET_8 (1,0,0)	-	1791
NSFNET_8 (0,1,0)	-	968
NSFNET_8 (0,0,1)	-	1
NSFNET_8 (1,0,1)	-	1809
NSFNET_8 (0,1,1)	-	3585

on the simulated annealing scheme. Both methods are used to solve the reconfiguration problem with different instances.

Using a MILP model allows to find near optimal solutions for small and medium instances. The difficult task is to prove the optimality of the solution found. However, this approach is computationally expensive for large instances. The SA algorithm is able to find good solutions with a run-time significantly lower than the solver run time. Hence the SA scheme can be used as a heuristic to arrive at near optimal solutions when the run-time of the MILP becomes practically infeasible.

The trade-off between the reconfiguration cost and the solution quality is the essence of the reconfiguration problem and cannot be left aside. This appears experimentally: Giving a little flexibility with respect to one metric drastically improves the quality of the solution with respect to other metrics. On the contrary, optimizing a single metric does not lead to balanced solutions, that is, the performance with respect to the optimized metric is good while the performance with respect to other metrics are not.

This work considers three different metrics to evaluate the quality of a solution. However we do not make a comprehensive multiobjective study of the problem. Carrying such analysis could derive useful information such as relationship between the metrics.

REFERENCES

[1] F. Télécom, Ed., *Les communications optiques du futur*, June 2002, mémento technique France Télécom 19. http://www.rd.francetelecom.fr/fr/conseil/mento19/index.html.

[2] F. Télécom, "Document de référence 2001," France Télécom, Tech. Rep., 2002.

[3] R. Ramaswami and K. Sivarajan, *Optical networks. A practical perspective*. Academic press / Morgan Kaufmann Publishers, 1998.

[4] I. Baldine and G. Rouskas, "Traffic adaptive WDM networks: A study of reconfiguration issues," *Journal of Lightwave Technology*, vol. 19, no. 4, pp. 433–455, April 2001.

[5] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: An approach to high-bandwidth optical WAN's," *lEEE Transactions on Communications*, vol. 40, no. 7, pp. 1171–1182, July 1992.

[6] G. Rouskas and H. Perros, "A tutorial on optical networks," *Lecture Notes in Computer Science*, vol. 2497, pp. 155–193, 2002, networking 2002 tutorials.

[7] A. Narula-Tam and E. Modiano, "Dynamic load balancing in WDM packet networks with and without wavelength constraints," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, pp. 1972–1979, 2000.

[8] D. Banerjee and B. Mukherjee, "Wavelength-routed optical networks: Linear formulation, resource budgeting tradeoffs, and a reconfiguration study," *IEEE/ACM Transactions on Networking*, vol. 8, no. 5, pp. 598–607, 2000.

[9] M. Kato and Y. Oie, "Reconfiguration algorithms based on meta-heuristics for multihop WDM lightwave networks," in *IEEE International Conference on Communications (ICC)*, June 2000, pp. 1638–1644.

[10] A. Gençata and B. Mukherjee, "Virtual-topology adaptation for WDM mesh networks under dynamic traffic," in *INFOCOM*, vol. 1. IEEE, June 2002, pp. 48–56.

[11] M. Sridharan, M. Salapaka, and A. Somani, "A practical approach to operating survivable WDM networks," *IEEE journal on selected areas in communications*, 2002.

[12] A. Gençata and B. Mukherjee, "Virtual-topology adaptation for WDM mesh networks under dynamic traffic," *IEEE/ACM Transactions on networking*, vol. 11, no. 2, pp. 236–247, April 2003.

[13] R. Dutta and G. Rouskas, "A survey of virtual topology design algorithms for wavelength routed optical networks," NCSU CSC, Tech. Rep. TR-99-06, June 1999.

[14] H. Zhu, H. Zang, K. Zhu, and B. Mukherjee, "Dynamic traffic grooming in WDM mesh networks using a novel graph model," in *Globecom 2002*, 2002, pp. 2681–2685.

[15] Z. P. Jr., P.P.R.Teixeira, and G. Mateus, "Traffic grooming and reconfiguration for incremental traffic in WDM optical networks," in *International Network Optimization Conference (INOC)*. Informs, October 2003, pp. 454–459.

[16] G. Huiban and G. R. Mateus, "A MILP model for the reconfiguration problem in multi-fiber WDM networks," in *SBRC Simpósio Brasileiro de Redes de Computadores*, May 2005.

[17] P. Datta, M. Sridharan, and A. Somani, "A simulated annealing approach for topology planning and evolution of mesh-restorable optical networks," in *8th IFIP Working conference on optical networks design and modeling (ONDM)*, February 2003.

[18] M. Tornatore, G. Maier, and A. Pattavina, "WDM network optimization by ILP based on source formulation," in *INFOCOM*, vol. 3. IEEE, June 2002, pp. 1813–1821, dOI 10.1109/INFCOM.2002.1019435.

[19] S. Kirkpatrick, C. G. Jr., and M. Vecchi, "Optimization by simulated annealing," *Science*, 1983.

[20] P. Batchelor, B. Daino, P. Heinzmann, C. Weinert, J. Späth, B. V. Caenegem, D. Hjelme, R. Inkret, H. Jäger, M. Joindot, A. Kuchar, E. L. Coquil, P. Leuthold, G. de Marchis, F. Materaand, B. Mikac, H.-P. Nolting, F. Tillerot, and N. Wauters, "Ultra high capacity optical transmission networks : Final report of action cost 239," Faculty of Electrical Engineering and Computing, HR, Zagreb, Tech. Rep. ISBN 953-184-013-X, 1999.