Heuristic Algorithms for Joint Configuration of the Optical and Electrical Layer in Multi-Hop Wavelength Routing Networks

Tibor Cinkler[1] Dániel Marx[1] Claus Popp Larsen[2] Dániel Fogaras[1]

Solutions presented in this paper are alone or in certain combination covered by EU patents 19719170 and 19810348

[1] cinkler@ttt-atm.ttt.bme.hu
High-Speed Networks Laboratory, Department of Telecommunications and Telematics, Technical University of Budapest, Budapest, Hungary
[2] Claus.Popp.Larsen@etx.ericsson.se
Ericsson Telecom AB, Stockholm, Sweden

Abstract— **An efficient and general graph-theoretic model (the Wavelength-Graph (WG)) has been proposed which enables solving the static Routing and Wavelength Assignment (RWA) problems in Multihop Wavelength Routing (WR) Wavelength Division Multiplexing (WDM) Networks simultaneously, and - as a unique feature - it optimises the optical layer jointly with the electrical one. Based on the proposed WG model the problem has been formulated as an Integer Linear Program (ILP), solved by stochastic algorithms improved by simple heuristics.**

The topology of the physical layer, the type of each node (e.g., OADM, OXC or EXC), the number of available wavelengths per link and the capacity of each wavelength-channel are assumed given with the aggregated traffic demand of each node-pair. The output of the optimisation is the system of wavelengthpaths, lightpaths and semi-lightpaths.

The objective of the optimisation is to reduce resource usage at upper (electrical) layers, subject to constrained amount of capacity of each wavelength and limited number of wavelengths. Although the problem to be solved is NP-hard, all methods proposed give result in very short time.

I. Introduction

WDM has been introduced to increase the transmission capacity of existing optical links. Instead of using one several transmitter and receiver pairs were used over the same fibre at different wavelengths forming independent channels and overbridging the speed limitations of electronics. It has been soon recognised that the switching decision can be made according to the incoming wavelength without any processing of the data stream. In WDM based All-Optical Networks (AON, where the whole network including the user-to-network interface (UNI) is optical) a wavelength is assigned to a connection in such a way that each connection (wavelength) is handled (switched) in the optical domain without any electrical conversion during the transmission [1]. This will be referred to as a *wavelength-path*. Wavelength (WL) reuse is allowed in parts of the network where that WL was not used. This WDM AON would require a huge number of different WLs. The technology sets the limit to around 40 different WLs per fibre in the 1550 nm window with 100 GHz (about 0.8 nm) spacing in the flat operating gain band (1530-1560 nm) of the present Erbium-Doped Fibre Amplifiers (EDFA) according to the recently completed ITU-T Recommendation G.692. For this reason WL conversions are needed. The most expensive way is to make optical WL conversion which ensures transparency of the network. Then, the end-to-end channel will be referred to as a *lightpath*. Simpler and cheaper method is to do first opto-electrical conversion, electrical space-switching and then electro-optical conversion. The end-to-end connection will then use a *semi-lightpath*. This is the idea for realising so called opaque networks, where systems using different sets of WLs are to be interconnected. The optimisation method proposed in this paper can be applied to all, wavelength-paths, lightpaths and semi-lightpaths.

It has been shown in [2] that the required number of different wavelengths per fibre for networks with and without WL conversion capability is about the same. In general, for networks of practical size, the number of available wavelengths is lower by a few orders of magnitude than the number of connections to be established. The only solution here is to join some of the connections (sometimes referred to as traffic grooming), which can fit into the capacity of wavelength-links. This can be done at electrical layer only since re-multiplexing is required. For this reason, taking not only the optical, but both, optical and electrical layer into account when configuring the system is demanded.

Many excellent papers deal with design, configuration and optimisation of WDM Networks. See, e.g., $[2-8]$. The widely accepted approach is to decompose the problem to the following sub-problems in given order. First, determine the virtual topology (route the light-paths); second, assign a wavelength to each light-path (WA); and third, route the traffic over the light-paths.

In [3] a heuristic (greedy) algorithm is proposed for WA followed by routing over established light-paths. In [4] is defined a bound for carried traffic in all-optical networks and shown that the proposed heuristic Routing and Wavelength Assignment (RWA) algorithm gives results very close to this bound. A performance study has been carried out. In [5] mathematical formulation of the design problem is given along with heuristics for solving the sub-problems relaxing some of the constraints one-by-one. In [2] there is also given a heuristic algorithm, and is showed that networks with and without wavelength converters require about the same number of wavelengths. A hybrid solution is also proposed where wavelengths are electrically regenerated in some specific nodes. In [6] multi-commodity flow model with randomised rounding is applied followed by graph colouring algorithms. In [7] an algorithm is given for lightpath routing by transforming the network to a special structure called wavelength graph (Although we will use the same term the structure covered by it differs significantly). In this graph costs are assigned to edges and shortest path algorithms are run. The optimality of the algorithm is also proved. In [8] design principles of Optical Networks are explored. The design is formulated as an optimisation problem with two objectives to be solved by heuristics. The (weighted) average delay trough the network is minimised, while the total carryable traffic over the network is maximised. In [9] the extended layered graph is used which is a bit similar to our model, but not flexible enough. In [10] the optical path routing strategies for WDM networks are investigated. The performance of networks with and without wavelength converters is evaluated. In [11] is described a method for planning WDM layer for carrying ATM traffic over it with the survivability constraint using Tabu Search. In [12] a reconfigurable OADM is demonstrated. In [13] hitless reconfiguration of WDM networks is investigated. [14] gives a performance evaluation and comparison of WDM networks with and without wavelength interchange capability. [15] proposes an algorithm for simultaneous Routing and Wavelength Assignment on a Path Graph. Channel capacities are not taken into account. In [16] an exact linear programming formulation is presented and the "closest" virtual topology is chosen for reconfiguration. [17] investigates the performance of partial reconfiguration on a SDM/WDM architecture. [18] proposes a model and algorithm for finding the globally optimal WL assignment with high probability using generally applicable heuristics for global optimisation. In [19] Mixed Integer Linear Programming (MILP) formulation of the static RWA problem is given for the case without WL changers as a minimax problem. The MILP formulation is then relaxed to LP and solved, followed by rounding.

Our subject is to configure the light-path system optimally without separating the network-layers. This improves the quality of results, but on the other hand the complexity of the problem grows.

As the optimisation result we decrease the traffic to be processed and carried in the electrical domain over-bridging the speed limits of electronics. Since a considerable part of the load of electrical, e.g., ATM switches is undertaken by the optical switches much larger networks with higher loads can be realised by the current technology offering better granularity and using optimally any limited number of WLs.

In Section II we present the model of the network with different node-types. In Section III we formulate the problem first informally, and then formally as an Integer Linear Program (ILP). In Section IV solution alternatives are proposed and in Section V heuristic algorithms are investigated those. Section VI presents the numerical results comparing different methods while Section VII draws conclusions.

II. The Wavelength Graph (WG)

The task was to provide a general model for configuration of WDM networks with different types of nodes and arbitrary topologies. Although the most popular topology is ring or interconnected rings, the model must be able to handle any specific or mesh topology. The nodes can also be quite different: Optical Add-and-Drop Multiplexers (OADM), Optical Cross-Connects (OXC) with full or limited (optical or opto-electrical) WL conversion or even an Electro-Optical Cross Connect (EXC). The protection strategies can also be quite different. All these aspects are taken into account in the proposed model. First the link model is described followed by models of different nodes. In this section we assume that all traffic demands are bidirectional and symmetrical. In this case the network can be modeled by an undirected graph. The model can be simply generalised for un-symmetrical demands, by using directed graphs. In later case the model is more complex and for this reason the algorithms will run slower.

A. Model of Links

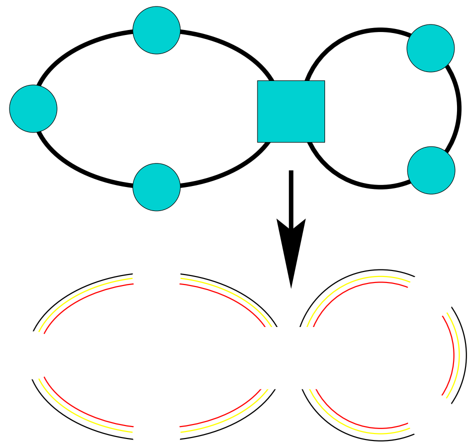

Fig. 1. Modeling edges.

A network consists of nodes, and links connecting the nodes. This can be modeled by a graph: a node is a vertex and a link is an edge. Having multiple WLs we will represent a WL of a link as an edge in the graph of wavelengths according to Figure 1 for the network proposed in [20]. To prioritise filling up WLs one-by-one we can assign slightly different weights to different channels of one link. For example, edges representing WL1, WL2 and WL3 will have weights 101, 102 and 103 respectively.

B. Model of Nodes

A node is modeled by a subgraph. The subgraph-nodes are the switch-ports, while the weighted edges represent the costs of transitions, terminations, conversions, etc. There are different types of nodes. Models of nodes differ for these. Here will be shown some examples. In similar manner a model can be derived for any additional node-type. The models proposed here are similar to those described in [21], but those were used for setting up connections one-by-one using shortest path algorithms, while here is the emphasis on global simultaneous configuration requiring special node-models.

B.1 Optical Add-and-Drop Multiplexer: OADM

The OADM Nodes have in general two bi-directional ports (4 fibres). Their function is either to transmit a WL channel or to terminate it and usually they do not allow WL-conversion.

The weights assigned to edges representing termination (e.g., 50) are higher than weights of transition (e.g., 25), because transition is preferred to termination. According to the proposed model (Figure 2) the traffic streams can enter or exit the OADM crossing vertex E or can be even re-multiplexed.

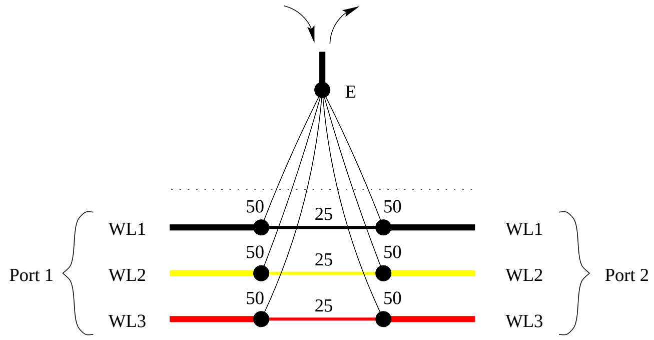

Fig. 2. Model of OADM Nodes.

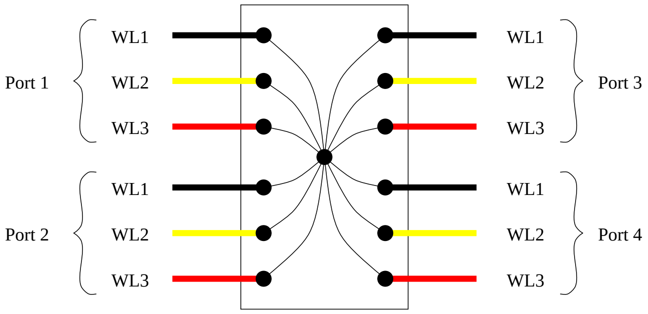

Fig. 3. Model of EXC Nodes.

B.2 Electro-Optical Cross-Connect: EXC

In the model shown in Figure 3 each pair of nodes should be connected by an edge. All edges should have equal weights. Instead of connecting all pairs using nxn edges we use n edges and one node. This simplifies the model. Each incoming channel is converted to electrical domain switched by a space-switch and again converted to the optical domain to arbitrary WL. Each termination, transition or WL change of a light-path has the same cost (e.g., 25). Therefore all edges have the same weight (e.g., 25/2).

B.3 Optical Cross-Connect: OXC

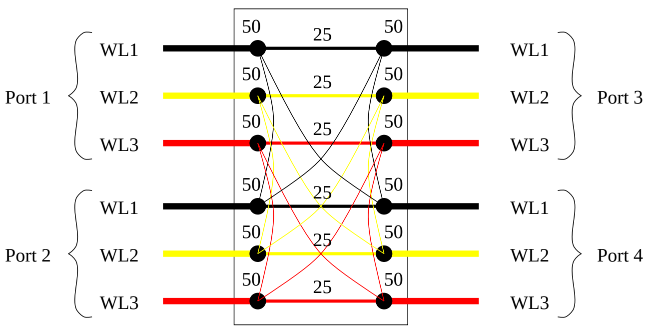

Fig. 4. Model of a Simple Optical Cross-Connect Node (without WL conversion)

An optical Cross-Connect has more than two ports, e.g., four bi-directional ports according to Figure 1. In an OXC a light-path can make transition to any output port which supports that WL, and that WL is not yet used. This OXC type (without WL change capability) will be referred to as *simple* OXC (see Figure 4). In this case one incoming channel can exit at any of the remaining output ports where that WL is supported and not yet used.

In some OXC devices WL translation (change) is also supported. This node will be called OXC. Its model is showed in Figure 5. For this node any incoming channel can exit at one of 11 remaining channels. Now there are 3 possibilities for light-path transition since there are channels of the same wavelength on all ports in our example. For WL change there are 8 possibilities. It has higher cost (e.g., 2x50=100) than the WL transition. WL change is modeled by conducting the traffic stream through node E.

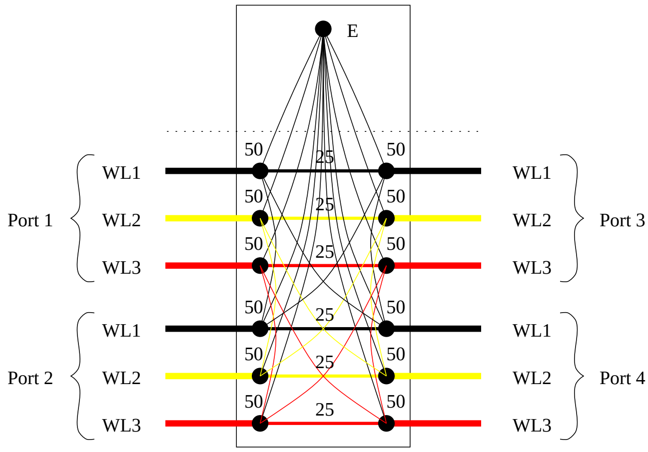

Fig. 5. Model of an Optical Cross-Connect Node (with WL conversion)

In some cases the traffic stream termination is also among the functions of an OXC. In that case the model does not need any change. The only difference will be that there will be some traffic offered to that OXC node which can be modeled by offering traffic to node E and considering it as an end-node. In this case traffic-stream re-multiplexing capability is also required.

B.4 Modeling opto-electro-optical conversions, multiplexing and re-multiplexing

If we want to differentiate the simple wavelength-change from the electrical signal re-multiplexing a more complex model is needed. An example has been shown in Figure 6 for an OADM node for simplicity reasons, which can be extended to any other node-type. As can be seen node E has been substituted by a fully connected graph. In this case assigning costs to internal edges the costs of wavelength-change and signal re-multiplexing can be differentiated.

All-Optical WL conversion is not supported by all OXCs. Therefore the optical signal is terminated and passed to the electrical layer where space switching or space switching with time switching (re-multiplexing) is done and then the resulting electrical signal passed back to the optical layer.

In cross connects three levels of cross-connecting and switching are to be differentiated:

- WL transition - This is done by the optical layer without any processing. This is the preferred and cheapest function. The signal can bypass the electrical layer using lightpaths of the optical layer.

- WL translation - It can be carried out by optical WL shifters, or by opto-electrical conversion, space-switching and electro-optical conversion. It is more expensive than the previous function, but still cheaper than the next one. Here is the switching very simple and no traffic stream processing is needed.
- multiplexing and re-multiplexing - In a larger WDM transport network there are considerably less available WLs per fibre than it would be needed for full interconnection of the end nodes by single-hop lightpaths. For this reason some of the traffic streams have to be multiplexed along a lightpath, i.e., in some cases the lightpath termination is not a traffic stream termination. In these cases time-division re-multiplexing is needed.

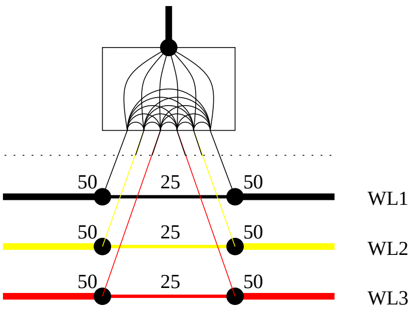

Fig. 6. Modeling opto-electro-optical conversions and re-multiplexing: the complex model

III. Problem Formulation

It is algorithmically very complex to obtain globally optimal solution for the global simultaneous routing and wavelength assignment problem. (The problem can be expressed as well as configuration of the lightpath system of a WDM network.) This problem very likely belongs to the class of NP-hard problems [22], because its sub-problem, the Static Lightpath Establishment (SLE) has been shown to be NP-hard [23]. Therefore it is righteous to use approximations. The proposed model enables solving the Routing and Wavelength Assignment problems *simultaneously* either using approximations or exact formulation.

The task was to find a shortest path in the obtained graph (which is built up of the link and node models) between all pairs of nodes simultaneously. There are alternatives for choosing the objective of the optimisation, e.g.,:

1. Minimise the total number of used WLs per fibre.
2. Decrease the total amount of used resources at the optical layer.
3. Decrease the total amount of used resources and processing at the electrical layer.
4. Decrease the total amount of lost traffic. It is also possible to introduce a scale-up factor for all traffic demands. For example scaling up all traffic demands by 10% the network should still work properly.
5. Minimise the number of WL conversions in total and for each path.

Our objective function will optimise 3. and 5. simultaneously, as will be discussed.

A. ILP formulation

The above described problem can be formulated as an Integer Linear Program using the proposed model. For this purpose the undirected graph model will be used, which is less complex (i.e., needs less variables) than the model with directed graphs. This formulation has slight similarities with Minimal Cost Multicommodity Flow (MCMCF) problem formulation [24].

Objective:

$$\text{minimise} \sum_{w \in W} \sum_{o \in O} c_w b^o x^o_w$$

Subject to constraints:

$$\sum_{\forall o \in O} x^o_w b^o \le B_w \qquad \forall w \in W \tag{1}$$

$$\sum_{\forall j \in A_i} x^o_{ij} = 1 \qquad \forall i \in V_E, \quad \forall o \in O \tag{2}$$

$$\frac{1}{2} \sum_{\forall j \in A_i} x^o_{ij} = z^o_{xi} \qquad \forall i \in V \setminus V_E, \quad \forall o \in O \tag{3}$$

$$x^o_w \le y_w \qquad \forall w \in W, \quad \forall o \in O \tag{4}$$

$$y_w \le \sum_{\forall o \in O} x^o_w \qquad \forall w \in W \tag{5}$$

$$\frac{1}{2} \sum_{\forall j \in A_i} y_{ij} = z_{yi} \qquad \forall i \in V \setminus V_E \tag{6}$$

$$x^o_w \in \{0,1\} \qquad \forall w \in W, \quad \forall o \in O \tag{7}$$

$$y_w \in \{0,1\} \qquad \forall w \in W \tag{8}$$

$$z^o_{xi} \in \{0,1\} \qquad \forall i \in V \setminus V_E, \quad \forall o \in O \tag{9}$$

$$z_{yi} \in \{0,1\} \qquad \forall i \in V \setminus V_E \tag{10}$$

where the following notation has been used:

$o \in O$	OD pair demands (commodities) o of the set of all commodities O
$w \in W$	wavelength-links w of the set of all wavelength-links W
$x^o_w, x^o_{i,j}$	flow indicator of commodity o over wavelength-link w or link $(i-j)$
y_w	light-path indicator
z	indicator for emulating the condition "equal to 0 or 2"
b^o	bandwidth requirement for traffic stream o
c_w	cost for using wavelength-link w or link $i-j$
B_w	bandwidth (capacity) of wavelength-link w or link $i-j$
i, j	node-indices in the model
A_i	set of nodes adjacent to node i
V	set of all nodes (vertices) in the model
$V_E \subset V$	set of all end-nodes (vertices)

The objective is to minimise the number of hops for each traffic demand weighted by the required capacity of that traffic stream and by the cost of using those light-links subject to the following constraints. The first constraint states that the amount of traffic using a light-link may not exceed the capacity of that light-link. The second constraint ensures that traffic-streams are to be terminated at end-nodes. Third the traffic flows must be conserved at each non-end node. Constraints 4 and 5 guarantee that traffic streams may use only available light-paths, and a light-path will be established only if it is needed for carrying a traffic flow. Sixth constraint expresses that a light-path can not branch. The last four constraints mean that all variables can take values 0 or 1.

The optimisation will result in a single-hop configuration (Wavelength-paths) whenever possible or in a multihop configuration with as few WL translations and re-multiplexing as possible, i.e., the largest possible part of the load of the electrical layer will be overtaken by the optical layer.

If the aim was to decrease the number of used WLs in total the objective can be expressed as

$$\text{minimise} \sum_{w \in W} \left(c_w \sum_{o \in O} b^o x_w^o + y_w \right)$$

as well. In this case constraint (5) can be avoided.

B. Comments on the Problem

In paper [2] the authors present different ways of formulating the problem for both single-hop and multihop lightpaths referred to as Wavelength-Path (WP) and Virtual Wavelength-Path (VWP) respectively. Among other methods they also use ILP. The drawback of the VWP formulation using ILP (and also of other methods applied to VWP) is that they implicitly assume pure electrical nodes, where not only electrical space-switching but also time-switching, i.e., re-multiplexing has to be performed. This approach degrades the Wavelength Routing network to a network employing WDM links and therefore it requires electrical (e.g., ATM) switches of large capacities increasing the costs and deteriorating the performance. The advantage of the method proposed in this paper is that it can differentiate various nodes with flexible functionality.

In paper [7] also a graph-model has been used but vertices are arranged in a matrix like grid, and edges representing lightpaths afterwards. Although the method is advantageous because of the applicability of fast shortest path algorithms, it does not take into account different node-types except the electrical switches. Furthermore, the method is used for routing of lightpaths only, not for configuration, i.e., simultaneous routing of all traffic demands.

The bin-packing problem, where we want to pack objects of different size into bins (of equal size) in optimal way is NP-hard. If the wavelength-channel capacities over all links are considered to be "bins" and we want to "pack" them optimally by traffic streams of different demands (different "objects"), we would have the same problem. However, our problem is even more complex, since there is "interaction" between "bins", because loading one bin will induce loading one of the neighbouring bins, and if two neighbouring bins are loaded by the object of same type no other bins from the neighbourhood can be loaded by objects of that particular type. For the above reason this method is also NP-hard.

IV. Solution Alternatives

The presented network model allows many methods for solving the above formulated problem. The alternatives include, but are not limited to:
- ILP Solvers
- Generally applicable heuristics for global optimisation[1] directly applied for 0-1 programming.
- Generally applicable heuristics for global optimisation[1] applied through specially matched models.
- Heuristic Algorithms based on decomposition.

Solving the problem by any available ILP solver (e.g., LP_SOLVE or CPLEX) will be possible for very small networks only. The reason is that the number of both, variables and constraints will grow by increasing the size of the network and this will result in exponential growth of the alternatives to be investigated by ILP software.

For this reason ILP formulation does not solve the problem, but enables running randomised "0-1" program solving methods, e.g., Simulated Annealing, Genetic Algorithm or Tabu Search - all using binary encoding. All constraints are evaluated, and if violated a penalty term is added to the objective. The penalty term for each constraint is a function of the penalty violation. Although different functions were tried out, ranging from linear to those which punish large deviations from the constraint more strictly, the method does not give expected results. Fine tuning of coefficients in the linear combination of the objective and penalty terms was needed. However, it has happened sometimes, that some traffic demands were not satisfied (x, y and z variables were 0), but non of the constraints was violated.

Based on the above formulation even more sophisticated heuristic methods can be used. The idea is to exclude a part of the state-space which is not of interest, instead of using penalty term as a means of obeying constraints. The work on it is still going on.

The decomposition according to the node-pairs appeared to be very promising. The next section will focus on it presenting our methods supported by the obtained results.

V. Heuristic Algorithms

In this section the proposed heuristic methods will be described, which are based on (decomposed to) shortest path searches using Dijkstra's algorithm. These algorithms can be grouped into three classes.
- S-SP: Sequential Shortest Path Algorithm
- P-SP: Parallelised Shortest Path Algorithm
- I-SP: Shortest Path Algorithm for Improving the obtained results. This method is to be run as a second phase after, e.g., P-SP.

A. Sequential Shortest Path Algorithm (S-SP)

Reference [21] suggests a simple, greedy method, where we successively satisfy the demands by finding a shortest path between the two end-nodes. It also predicts that we can get better results if we sort the demands by distance (number of hops), satisfying first the demand with the shortest distance between

[1] (e.g., Simulated Annealing (SA), Genetic Algorithm (GA), Threshold Accepting (TA), Tabu Search (TS), Go with the Winners (GW), etc.)

its end-nodes. Based on this deterministic approach we propose a randomised method here.

The basic S-SP algorithm is as follows:

S-1 Create a permutation of the set of demands.

S-2 Run Steps S-3 and S-4 for all demands (in the order determined in Step S-1).

S-3 Find a shortest path, which does not violate any capacity constraint, between the end nodes of the demand. If no such path is found, then STOP with "No solution found!"

S-4 Accept the found path for that demand, and modify the capacities of the graph accordingly.

If the graph has e edges, n nodes, the number of demands is d, and the Dijkstra's algorithm is implemented with a running time of $O(e \log n)$, then the total running time of this algorithm is $O(de \log n)$. If the permutation is selected by some randomized method in Step S-1, then it is possible to run the algorithm multiple times, and select the result with the smallest cost[2].

There are more possibilities for creating permutations:

- Select one permutation randomly, with equal probability for all the $d!$ permutations.
- Sort demands by non-decreasing distance. If we sort the demands by the number of hops between their end-nodes, we can get better results. Our experiments support this claim. Since there are demands with the same distance, there will be multiple distance-sorted permutations thus the algorithm can give different results when run multiple times.
- Sort demands by bandwidth requirement satisfying first the demand with the largest requirement. It gives much better results than the previous method when there were large differences between the bandwidth requirements of the demands.
- Sort the demands by both, distance and bandwidth requirement.

We have generalized the distance sorting and bandwidth requirement sorting. If $b^o (o \in O)$ denotes the bandwidth requirement and $D^o (o \in O)$ denotes the hop-count of the demand, then we sort the demands using the following function:

$$f(d) = \lambda (D^d / \sum_{o \in O} D^o) - (1-\lambda)(b^d / \sum_{o \in O} b^o),$$

that is, the sum of the normalized distance and (with negative weight) the normalized traffic. The parameter $\lambda (0 \leq \lambda \leq 1)$ can set the relative importance of the two components. Note that $\lambda = 0$ is sorting by bandwidth requirement, while $\lambda = 1$ is distance sorting.

However, it is easy to find networks and demand patterns where no order of the demands can produce an optimal solution (or even find a feasible solution) with this algorithm.

B. Parallelised Shortest Path Algorithm (P-SP)

The previous S-SP method often get stuck when trying to establish a path for each demand one-by-one. Therefore we have proposed to build the paths in parallel, segment-by-segment.

During the course of this algorithm, there is a path assigned to each demand. One end-node of this path must be at one of the end-nodes of the demand. In the beginning, these paths are of zero length (Step P-1), i.e., they consist of one point only, which is one of the end-nodes of the demand. During the algorithm, a demand is selected (Step P-5), and its path is extended by adding a node which is closer to the destination, i.e., to the the other end-node of the demand (Step P-8). If a path reaches its destination, then we accept it as a path satisfying the demand (Step P-9). Since we extend a path only if it does not violate any of the constraints, we will get a feasible solution when all the demands are satisfied. Occasionally we will also shorten a path to avoid getting stuck (Steps P-4 and P-7).

The P-SP algorithm is as follows:

P-1 *(Initialization)* Assign to each demand a zero-length path containing one of its end-nodes.

P-2 Set the weight of all demands to an initial weight.

P-3 Let x be a random number with uniform distribution between 0.0 and 1.0, if $x > p_1$ then go to Step P-5, otherwise continue with Step P-4.

P-4 *(Deleting a random path)* Select a demand randomly, with equal probability for all demands. Assign to this demand the zero-length path assigned in Step P-1, and the weight set in Step P-2. Continue with Step P-3.

P-5 *(Selecting a path)* Select a demand randomly, with probability proportional to its weight.

P-6 Let x be a random number with uniform distribution between 0.0 and 1.0, if $x > p_2$ then go to Step P-5, otherwise continue with Step P-7.

P-7 *(Crank-back)* If the path assigned to the demand has nonzero length, then remove the last node. Decrease the weight of the demand. Go to Step P-3.

P-8 *(Extending the path in the optimal direction)* Find a shortest path between the end-node of the path and the destination node, which obeys the capacity constraints. If such a path is found, then extend the path assigned to the demand with the first edge of the shortest path, and increase the weight of the demand. If no such path is found, then go to Step P-8.

P-9 If the extended path connects the two end-nodes of the demand, the demand is satisfied, set its weight to zero. If all the demands are satisfied, the algorithm stops and a feasible solution has been obtained. Otherwise go to Step P-3.

In the following paragraphs we will clarify some of the steps described above, and give further improvements and alternatives.

B.1 Weights

In Steps P-1 and P-4, an initial weight is assigned to the demands. This should be the function of the bandwidth requirement of the demand, and the number of hops between its end-nodes. A good weight function should "encourage" the demands with heavy traffic and small distance, hoping that we will get a better result if we deal first with these demands.

In Steps P-7 and P-8, we modify the weights of the demands in order to increase the probability of selecting a demand with a longer established path. We do this to avoid a situations where none of the demands is able to allocate a "good" path. With a larger modifying constant, at least some of them will be able to reach the destination quickly.

[2] Note, however, that if the Dijkstra's algorithm selects one path randomly from the set of shortest paths (as described above), then the algorithm might give different results even if the same permutation is selected in Step S-1.

B.2 Probabilities p_1 and p_2

Steps P-4 and P-7 are introduced to avoid getting stuck the algorithm, i.e., to enable for a path to backtrack from a situation where no path is available to the destination node. A lower value of probabilities p_1 and p_2 will make less likely that a stuck demand will be rerouted, while larger values increase the running time of the algorithm. We have found, that $p_1 = 0.02$ and $p_2 = 0.04$ give good results in our experiments.

B.3 The shortest path is not always realisable

In Step P-8 we determine the distance between the originator and the destination node by finding a shortest path. However, it might not be possible to allocate this path for the demand without violating some of the constraints. If there is a node on the path, where splitting of the traffic is not allowed, but there is already an edge incident to it which is not in the path, then there will be three edges incident to this node with traffic flowing on it if we allocate the path for the demand. Thus first edge of the shortest allowed path to reach the end node (if it is reachable at all) may differ from the first edge of the shortest path. In this case the path will be extended by a node which is not that one which would make it closest to the destination. This might degrade the performance of the algorithm. Unfortunately we have not found an efficient way to get around this problem yet.

B.4 Circle detection

It might happen that we add a node to the path in Step P-8 which already exists in the path. In this case, in order to avoid the creation of a circle in the path, we must delete the loop from the path, and add to the path the next node of the shortest path.

B.5 Abbreviating a path with more than one edge

If there is no path found in Step P-8, then the last edge is removed from the path assigned to the demand. A better approach is to find the node of the path which is closest to the destination node, to remove the remaining part of the path and to continue building the path from there. This speeds up the method and improves the results.

B.6 Building the paths from both directions

We can build paths starting from both end-nodes of the demand. This should be done in Step P-8. We will choose one of the two ends with equal probability for extending. The demand is satisfied, if the two paths reach a common node, i.e., the two paths connect the two end-nodes of the demand.

B.7 Shortest Paths

All three methods use Dijkstra's algorithm for finding a shortest path in a graph. If there are multiple equally short shortest paths within a graph, then it depends on the actual implementation which one will be found. The two different approaches we applied have negligible influence on the results:

- Select always the lexicographically first path from the set of shortest paths.
- All shortest paths have equal probabilities of being selected.

C. *Improving the obtained results (I-SP)*

The second-phase Improving Shortest-Path (I-SP) algorithm takes the feasible solution generated by either P-SP or some other randomised method and creates a solution which is not worse, but likely better than the original one. I-SP improves the solution by replacing some paths with shorter ones, without violating the constraints.

The I-SP algorithm is as follows:

I-1 Create a permutation of the demands.
I-2 Select the next demand (select the first demand again after the last one).
I-3 Remove the route of the demand from the solution and make the allocated capacity free.
I-4 Find a shortest path between the end-nodes of the demand which does not violate any of the constraints.
I-5 If this shortest path has lower cost than the previous one for the same demand then keep it. (Otherwise keep the old one).
I-6 Allocate the capacity for the demand.
I-7 If all demands were selected since the last time a route was changed in Step I-5, then the algorithm stops (because we have made a full circle, and found no path which could be improved), otherwise go to Step I-2.

If the cost of the paths can take only nonnegative integer values, then the algorithm will always terminate: Step I-5 can decrease the total cost only a finite number of times, and after the last change, Step I-7 will eventually cause the algorithm to stop.

Step I-4 will always find a path, because the removed route was part of the solution satisfying the constraints. Note that this algorithm will not improve a solution created by S-SP, because every path was a shortest path in a graph where only some of the demands were allocated. The path found in Step I-4 cannot be of lower cost than the path of the demand, which was originally a shortest path.

In Step I-1 we have several possibilities of creating a permutation including those described for S-SP Algorithm.

VI. TESTING THE ALGORITHMS

A. *Criteria for Comparing the Results*

For comparing the results of different algorithms on different test networks an objective function is needed which measures how good the result is. Preferably, this function should indicate the cost of implementing the obtained solution. Below we define several functions useful for comparing the results.

A.1 Cost function

The cost function, as described in Section III.A is a natural candidate for comparing the results. It measures the load of the both, electric and optical layer. The algorithms will minimize this value according to the weights assigned to the edges.

A.2 Load of the electric layer

By setting the weight c_w to 0 for all edges of the optical layer we would obtain the cost of configuring the electrical layer, i.e., the weighted load of the electrical layer. However, in this case heuristic algorithms would not provide useful results.

Let $W_E \subset W$ denote the set of edges connecting the electric end-nodes to the optical layer, where W is the set of all edges, i.e., $W_E = \{w | w \in W, w = (v_1, v_2), (v_1 \in V_E \wedge v_2 \in V \setminus V_E) \vee (v_2 \in V_E \wedge v_1 \in V \setminus V_E)\}$. Then the total load of the electric layer can be expressed as $L_E = \sum_{w \in W_E} \sum_{o \in O} b^o x_w^o$ while the electric cost of the obtained configuration is $C_E = \sum_{w \in W_E} \sum_{o \in O} c_w b^o x_w^o$.

If the opto-electric conversion including electric processing has the same cost in all network nodes and at each wavelength ($\forall w \in W_E : c_w = c_E$), then a lower bound for C_E can be given as $C_E^{min} \geq 2c_E \sum_{o \in O} b^o$, since each demand must be processed at least twice in the electric layer: once at both ends. This value can be subtracted from the total electric cost to obtain the cost of switching and multiplexing: $C'_E = C_E - C_E^{min}$. This nonnegative value indicates the cost of "unnecessary" electric processing, compared to the ideal, all-optical network, where every demand reaches its destination in a single hop, i.e., where $C'_E = 0$. Analogous relations can be derived for the loads L_E, L_E^{min} and L'_E, as well by setting $c_w = c_E = 1$.

At an electric node, two types of electric processing are required. If all the traffic of a light-path terminated at the considered node will continue on a single light-path without adding or dropping any part of it, then simple electrical space switching is enough. If the traffic is going to be groomed (multiplexed, carried jointly in the optical domain and then demultiplexed), then electric space and time switching is required. Since the later operation requires more complex equipment we are interested in the ratio of these two.

We will call edges w_1 and w_2 pairs, if they are at the same electric node, and they carry the same traffic, that is, $w_1 = (v, v_1), w_2 = (v, v_2), v \in V_E, v_1, v_2 \in V \setminus V_E, v_1 \neq v_2$ and $\forall o \in O : x_{w_1}^o = x_{w_2}^o$. It is clear that every edge w can have at most one pair. Let W_p denote the set of paired edges.

The traffic on paired edges can be handled by switching the traffic from one light-path to the other, so

The total traffic which will be processed by simple electric space switches is $L_s = \sum_{w \in W_p} \sum_{o \in O} b^o x_w^o$, while the total load of the time-and-space switching (i.e., re-multiplexing) is $L_m = L'_E - L_s$.

If optical wavelength conversion is available, then the load of the electric layer will be decreased by L_s.

A.3 Average electrical hop-count

Let H^o denote the number of light-paths the path of the demand o consists of. The demand will be processed electrically at $H^o + 1$ network nodes. H^o can be formulated as $H^o = 1/2 \sum_{w \in W_E} x_w^o$. The average number of hops is $\bar{H} = 1/|O| \sum_{o \in O} H^o$. The weighted average number of hops is $\bar{H}_w = (\sum_{o \in O} b^o H^o)/(\sum_{o \in O} b^o)$. The weighted average hop-count is an equally good measure of the solution as the total electric load, because of the following equality:

$$L_e = 2\bar{H}_w \sum_{o \in O} b^o = \bar{H}_w C,$$

where C is a constant independent of the solution.

B. Numerical Results

The algorithms have been implemented in C++ with Java GUI. Testing has been carried out on Sun UltraSparc computers. Since the algorithms are not deterministic each test was run 200 and 500 times for test networks of Sections B.1 and B.2 respectively. In Section B.1 we consider regular networks of different sizes to show the effects of the different methods on the cost function, and on the electric load. Then, in Section B.2 we optimise a network of practical interest investigating the dependence on the number of WLs and WL channel capacities. In all experiments the weights c_w were 35, 36, 37,... for the edges within the optical domain for different WLs, 10 for WL transitions and 100 for terminating a WL path and going to the electrical layer.

B.1 Comparing the Methods

Five versions of the proposed algorithms have been compared for three networks N20, N30 and N42 consisting of 20, 30 and 42 nodes respectively (Fig. 7) and having 100, 200 and 300 units of capacity per all 5 wavelengths within each link, respectively. All traffic matrix entries can take values 2, 4, 6, 8 or 10. Each method was run 200 times. The running times are ranging from 0.04 s for S-SP for N20 to 12 s for the two-phase (P-SP + I-SP) method for N42. The five versions of the algorithm are as follows:

M1 S-SP with random order of demands (reference method).
M2 S-SP with sorting according to linear combination of normalised hop-counts and traffic-demands.
M3 P-SP with equal weights assigned to all wavelengths.
M4 P-SP with slightly different weights assigned to different wavelengths.
M5 As M4, but with randomly chosen shortest path from the set of shortest paths.

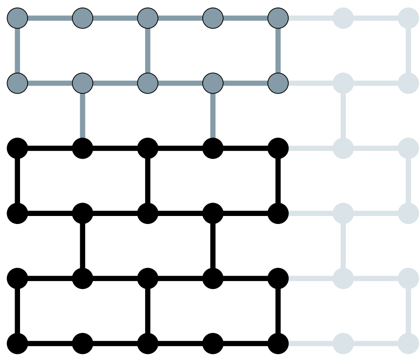

Fig. 7. The N20 test network (black), with extensions to N30 (dark grey) and to N42 (light-grey).

Figures 8 and 9 show the maximal, average and minimal values for network costs and weighted average number of hops for N20, N30 and N42. All nodes of degree 2 were OADM while the others were OXC nodes, all without optical WL conversion capability, but with electrical re-multiplexing.

Based on the obtained numerical results the following conclusions can be drawn:

- The P-SP methods M5 and M4 (and M2 particularly for N30) show better performance than methods M1 and M3 regarding the cost function.
- The S-SP methods, particularly M2 show better performance than the P-SP methods M3, M4 and M5 regarding the electric load.

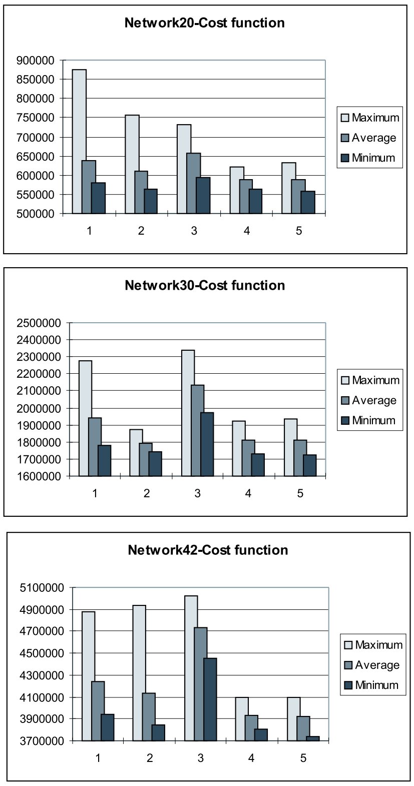

Fig. 8. The costs for N20, N30 and N42

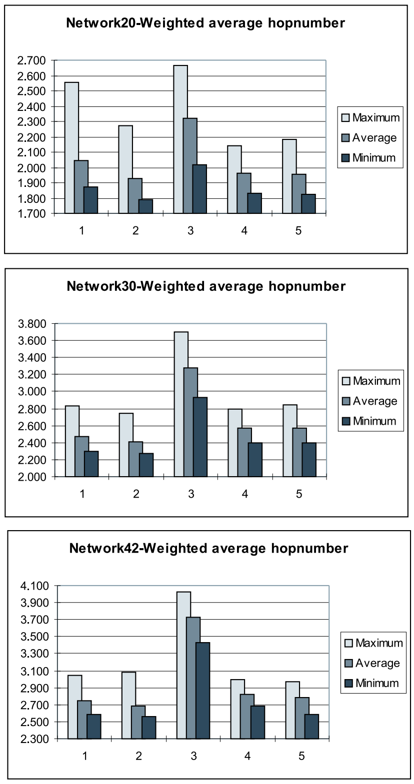

Fig. 9. The weighted average hop-count for N20, N30 and N42

- In most cases the S-SP methods produce greater dispersal of results than the P-SP methods.
- Assigning slightly different weights to different wavelengths improves the performance significantly: Method M3 is the worse one.
- The randomly chosen shortest path slightly improves the performance of P-SP methods: M5 is slightly better than M4, mostly regarding the costs.
- Sorting the demands improves the performance significantly: M2 performs always better than M1.

B.2 Applying the Proposed Methods

Figure 10 depicts a pan-European optical transport network. All nodes are labeled with the names of European capitals followed by a number in brackets ranging from 5 to 10. The traffic demand between two capitals (two states) is the product of these two associated numbers ranging from 25 to 100 (e.g., the demand between Stockholm and Budapest is 36).

The nodes having degree larger than two (having direct link to more than two other nodes) are OXCs while the others are OADMs except that one representing Portugal (labeled as Lissa) because of the trans-Atlantic cable.

For this network, we have found that method M2 has the best performance in all examined cases, i.e., the best value found in 500 runs is lower than for the other methods.

Table I presents the results for method M2 when increasing the number of available WLs per link and when increasing the capacity of WL channels while keeping the number of WLs unchanged.

The demands were first accommodated by 8 WLs, each having capacity of 150 units. Increasing the WL channel capacities, as expected, led to lower number of electrical hops, and lower values of cost functions.

Then, we have increased the number of available WLs to 16. Both, the weighted average hop-count and the configuration cost became lower. For 32 WLs all demands could have been accommodated by a single WL (i.e., in all-optical domain) without any electrical processing or WL conversion.

According to the results presented in Table I we claim that, both parameters reduce both, the cost function and the weighted average hop-count, therefore the load of the electrical layer is also reduced.

Increasing the number of wavelengths has far greater affect

number of WLs	Capacity of WL channels	Configuration Cost	Weighted Avrg. Hop-count
8	150	3426201	1.279
8	175	3387887	1.267
8	200	3351142	1.263
8	300	3346887	1.265
16	125	2953918	1.015
16	150	2948958	1.015
32	100	2920142	1.000

TABLE I

DEPENDENCES ON THE NUMBER OF WLS AND WL CHANNEL CAPACITIES

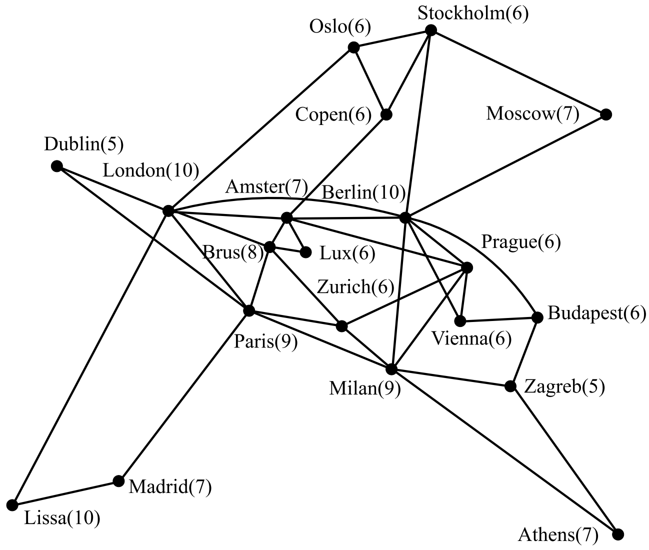

Fig. 10. Example of a pan-European Optical Transport Network

then increasing the capacity, because the increased capacity can only be used by electrical re-multiplexing and routing, which is in our case more expensive than the optical one (i.e., the wavelength routing).

S-SP methods perform better when the number of available WLs is larger as well as the density of the network. This was the case for the pan-European network. Where the node-pairs can be connected along very long paths only (spars topology) and where few WLs are available the shortest path searches get easily stuck. Therefore, for such networks the parallelised P-SP methods will give better results. Their drawback is the longer running time.

Note, that the methods also reduce the number of opitical hops (number of links crossed in the optical domain). Whether the total length, or the number of electrical processings along the paths will be preferrably minimised depends only on the proportion of the weights assigned to WL links and edges modelling electrical processing.

VII. Conclusion

The novelties of this proposal are the general model for WDM networks which enables the ILP formulation of the static RWA problem, the ILP formulation itself and the proposed basic algorithms including numerous heuristics for improving the performance.

The algorithms are very fast even for large networks with many WLs.

The unique feautre of our approach is that the presented models and algorithms are used to optimise jointly the electrical layer and the underlaying wavelength or fiber system.

VIII. Acknowledgements

This work has been done in the research co-operation framework between Ericsson and the High-Speed Networks Laboratory (HSN*Lab*) at the Department of Telecommunications and Telematics, Technical University of Budapest as part of the ACTS METON and ACTS DEMON projects. The authors are grateful to Miklós Boda (Ericsson) and Tamás Henk (HSN*Lab*) for their support.

References

[1] P.E. Green, “Optical Networking Update”, *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 764-779, June 1996

[2] N. Wauters, P. Demeester, “Design of the Optical Path Layer in Multiwavelength Cross-Connected Networks”, *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 881-892, June 1996

[3] Z. Zhang, A.S. Acampora, “A Heuristic Wavelength Assignment Algorithm for Multihop WDM Networks with Wavelength Routing and Wavelength Re-Use” *Transactions on IEEE Networking*, vol. 3, no. 3, pp. 281-288, June 1995

[4] R. Ramaswami, K.N. Sivarajan, “Routing and Wavelength Assignment in All-Optical Networks”, *IEEE Transactions on Networking*, vol. 3, no. 5, pp. 489-500, Oct. 1995

[5] R. Ramaswami, K.N. Sivarajan, “Design of Logical Topologies for Wavelength-Routed Optical Networks”, *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 840-851, June 1996

[6] D. Banerjee, B. Mukherjee, “A Practical Approach for Routing and Wavelength Assignment in Large Wavelength-Routed Optical Networks”, *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 903-908, June 1996

[7] I. Chlamtac, A. Faragó, T. Zhang, “Lightpath (Wavelength) Routing in Large WDM Networks” *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 909-913, June 1996

[8] B. Mukherjee, D. Banerjee, S. Ramamurthy, A. Mukherjee, “Some Principles for Designing a Wide-Area WDM Optical Network”, *IEEE Transactions on Networking*, vol. 4, no. 5, pp. 684-696, Oct. 1996

[9] H. Harai, M. Murata, H. Miyahara, “Performance of All-Optical Networks with Limited-Range Wavelength Conversion”*ICC’97 - IEEE International Conference on Communications*, Montréal, June 1997

[10] M. Garnot, M. Sotom, F. Masetti, “Routing Strategies for Optical Paths in WDM Networks”, *ICC’97 - IEEE International Conference on Communications*, Montréal, June 1997

[11] J. Armitage, O. Crochat, J.-Y. Le Boudec, “Design of a Survivable WDM Photonic Network”, *INFOCOM 97*, Kobe, April 1997

[12] L. Berthelon et.al., “Towards photonic networking: experimental demonstration of a reconfigurable survivable WDM ring network”, *GLOBCOM*, London, November 1996

[13] K. Bala, G. Ellinas, M. Post, “Towarwds Hitless Reconfiguration in WDM Optical Networks for ATM Transport”, *GLOBCOM*, London, November 1996

[14] E. Karasan, E. Ayanoglu, “Effects of Wavelength Routing and Selection Algorithms on Wavelength Conversion Gain in WDM Networks”, *GLOBCOM*, London, November 1996

[15] S. Banerjee, C. Chen, “Design of Wavelength-Routed Optical Networks for Circuit Switched Traffic”, *GLOBCOM*, London, November 1996

[16] D. Banerjee, B. Mukherjee, “Wavelength-Routed Optical Networks: Linear Formulation, Resource Budgeting Tradeoffs and a Reconfiguration Study”, *INFOCOM 97*, Kobe, April 1997

[17] J.-F. P. Labourdette, “Performance Impact of Partial Reconfiguration on Multihop Lightwave Networks”, *IEEE Transactions on Networking*, vol. 5, no. 3, pp. 351-358, June 1997

[18] T. Cinkler, “Heuristic Algorithms for Configuration of the ATM-Layer over Optical Networks” *ICC’97 - IEEE International Conference on Communications*, Montréal, June 1997

[19] R.M. Krishnaswamy, K.N. Sivarajan, “Design of Logical Topologies: a Linear Formulation for Wavelength Routed Optical Networks with no Wavelength Changers”, *IEEE INFOCOM’98*, March 1998

[20] S. Johansson, “Transport Network Involving a Reconfigurable WDM Network Layer - A European Demonstration”, *IEEE Journal on Lightwave Technology*, vol.14, no.6, pp. 1341-1348, June 1996

[21] T. Cinkler, R.S. Castro, S. Johansson, “Configuration and Reconfiguration of WDM Networks”, *NOC’98 - Conference on Networks and Optical Communications*, Manchester, June 1998

[22] M.R. Garey, D.S. Johnson, “Computers and Intractability — A Guide to the Theory of NP-completeness”, W.H. Freeman and Co., New York, 1979

[23] I. Chlamtac, A. Ganz, G. Karmi, “Lightpath Communications: An Approach to High Bandwidth Optical WANs”, *IEEE Transactions on Communications*, vol. 40, no.7, pp. 1171-1182, July 1992

[24] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, “Network Flows, Theory, Algorithms and Applications”, *Prentice-Hall*, 1993