A Two-Stage Simulated Annealing Logical Topology Reconfiguration in IP over WDM Networks

Sugang XU [1], Kaoru SEZAKI [2], and Yoshiaki TANAKA [1,3]

[1] Global Information and Telecommunication Institute, Waseda University,
1-3-10 Nishi-Waseda, Shinjuku-ku, Tokyo, 169-0051 Japan
Tel. +81-3-5286-8389   Fax.+81-3-5286-8389
[2] Institute of Industrial Science, The University of Tokyo,
4-6-1 Komaba, Meguro-ku, Tokyo, 153-8505 Japan
[3] Advanced Research Institute for Science and Engineering, Waseda University,
17 Kikuicho, Shinjuku-ku, Tokyo, 162-0044 Japan
E-mail: xsg@aoni.waseda.jp, sezaki@iis.u-tokyo.ac.jp, ytanaka@waseda.jp
Corresponding Author: Sugang XU

Abstract

WDM optical networks represent the direction to the future high capacity wide-area network applications. By reconfiguring the logical topology, network resources utilization can be optimized corresponding to the traffic pattern changes. From the viewpoint of network operation, the complexity of reconfiguration should be minimized as well. In this paper we consider the logical topology reconfiguration in arbitrary topology IP over WDM networks with balance between network performance and operation complexity. The exact formulation of the logical topology reconfiguration problem is usually given as a Mixed Integer Linear Programming, but it grows intractable with increasing size of network. Here we propose a simulated annealing approach to determine the target topology with a smaller logical topology change and satisfy the performance requirement. A threshold on the congestion performance requirement is used to balance the optimal congestion requirement and operation complexity by tuning this threshold to a feasible value. For an effective solution discovery, a two-stage SA algorithm is developed for multiple objectives optimization.

1   Introduction

Wavelength division multiplexing (WDM) makes the huge optical capacity of fiber to be utilized by transmitting multiple signals on different wavelengths on a single fiber. In IP over WDM networks, each node is equipped with an optical cross-connect (OXC). Each OXC is connected to an edge device, e.g., an IP router. Nodes are connected by fibers to form an arbitrary physical mesh topology. Any two IP routers in this network can be connected together by an all-optical WDM channel, called a lightpath [1]. By using OXCs at intermediate nodes and via appropriate routing and wavelength assignment, a lightpath can create logical neighbors out of nodes that are geographically far apart in the network. Thus, a set of lightpaths embeds a logical topology on the network [2][3].

The motivation of logical topology design is to optimize the network resource utilization improving network performance. The network can dynamically change its logical topology corresponding to the changing traffic conditions. It is called logical topology reconfiguration [4]-[7]. The general approach to the logical topology reconfiguration problem has been a two-phase operation: first phase being a logical topology design for the new traffic conditions, and second phase being a transition period from the old logical topology to the newly designed one, it should achieve the minimal traffic disruption. From the viewpoint of network operation, besides the network performance optimization, the complexity of reconfiguration should be minimized as well.

Normally, the logical topology reconfiguration problem is usually formulated as a Mixed Integer Linear Programming (MILP) [5] [6], but an exact formulation of this problem quickly grows intractable with increasing network size. In fact, this problem and some of its subproblems are known to be *NP-hard* [3][4]. Thus, for large networks it is not practical to attempt to solve this problem exactly. Consequently the feasible heuristic is very important, especially for large networks. There are various studies proposing heuristic methods for arbitrary logical topologies design [2][3]. In addition, the simulated annealing methods (SA) [3][8]-[10] as meta-heuristic method were also proposed to design logical topology in IP

over WDM networks [3][10]. Here a simulated annealing method to resolve the logical topology reconfiguration problem without wavelength conversion is proposed. To keep the connectivity of logical topology during reconfiguration, a multi-layer architecture of logical topology [7] was proposed, it consists of unchanged layer and reconfigurable layer. The unchanged layer logical topology is a connected topology, it is optimized for the long term unchanged traffic and assures at least the traffic will not be broken if lightpath add/delete operations cause unconnected topology for certain time during reconfiguration, whereas by reconfiguring the reconfigurable layer logical topology, network resources utilization will be optimized corresponding to new traffic conditions. Here we focus on the reconfiguration problem of the reconfigurable layer logical topology. Considering the balance between network performance and operation complexity, a threshold on the congestion performance requirement is introduced. For an effective solution discovery, a two-stage SA algorithm is developed for multiple objectives optimization. This paper is organized as follows: Section 2 describes the reconfiguration problem in a MILP model; Section 3 introduces the simulated annealing algorithm; Section 4 shows the numerical results; Section 5 concludes this paper.

2 Problem Statement

In this paper, the network consists of N nodes connected by bidirectional fiber forming an arbitrary physical topology. Each node i is assumed to have Δ_i tunable transmitters and receivers. Now, in most cases there are four main limitations to logical topology design: 1) The number of tunable transmitters and receivers Δ_i at each node i is limited, that is to say the logical degree of the node is limited. We set $\Delta_i = P$ for all nodes i (where $P<N$). 2) The number of wavelengths on each fiber is also limited, so there are W wavelengths that can be used on each fiber. 3) One wavelength cannot be used by different lightpaths on the same fiber. 4) Without the wavelength converter the lightpath has to use the same wavelength along the path.

2.1 Notations

Here are some notations which will be used in the following sections.

1) Physical Topology: A graph $G_p=(V, E_p)$ in which each node in the network is a vertex, where $|V|=N$. And each fiber optic link between two nodes is an arc. Each fiber link is also called a physical link. The graph is usually assumed to be undirected, because each fiber link is assumed to be bi-directional. There is a cost associated with each of the arcs, which is usually the fiber distance or propagation delay over the corresponding fiber,

in this paper for the simplicity it is assigned with unit cost;

2) Logical Topology: A graph $G_l=(V, E_l)$ in which the set of nodes is the same as that of the physical topology, and each lightpath is an arc, it is also called logical link. Usually this graph is assumed to be directed, here we create the bi-directional lightpath between two nodes;

3) Link indicator: If a physical link exists in the physical topology from a node i to another node j, denoted by p_{ij}, which is 1 if a link exists in the physical topology and 0 if not;

4) Lightpath indicator: If a lightpath exists in the logical topology from a node i to another node j, denoted by b_{ij}, which is 1 if such a lightpath exists and 0 if not;

5) Traffic matrix: A matrix, which specifies the traffic between every pair of node in the physical topology. The traffic matrix is an $N \times N$ matrix $\Lambda =[\lambda^{(sd)}]$, where $\lambda^{(sd)}$ is the average traffic from source node s to destination node d;

6) Logical traffic load: When a logical topology is established on a physical topology, the traffic from each source node to destination node must be routed over some lightpaths. The aggregate traffic resulting over a lightpath is the load offered to that logical link. If a lightpath exists from node i to j, the load offered to that lightpath is denoted by λ_{ij}, the component of this load due to traffic from source node s to destination node d is denoted by $\lambda_{ij}^{(sd)}$, the maximum logical link load is called the congestion, and denoted by $\lambda_{max}=max (\lambda_{ij})$;

7) Wavelength indicator: Let $c_{ij}^{(k)}$ be the lightpath wavelength indicator $c_{ij}^{(k)}$ is 1 if a lightpath from node i to node j uses the wavelength k, 0 otherwise. Let $c_{ij}^{(k)}(l,m)$ be the link-lightpath wavelength indicator, to indicate whether the lightpath from node i to node j uses the wavelength k and passes through the physical link from node l to node m.

2.2 Reconfiguration Formulation

The following is a MILP formulation of the logical topology reconfiguration problem. Here we treat the minimum network congestion objective as a constraint. The objective of reconfiguration problem is then to minimize the operation complexity.

Objective:
Minimize the operation during the reconfiguration. It is to minimize the difference between old logical topology and new target logical topology. (b'_{ij} is the lightpath indicator of the old logical topology.)

$$\min \sum_{ij} | b_{ij} - b'_{ij} | \qquad (1)$$

Subject to:
Degree Constraints

$$\sum_j b_{ij} \le P, \forall i \qquad (2)$$

$$\sum_j b_{ji} \le P, \forall i \qquad (3)$$

Bi-directional lightpath Constraints

$$b_{ij} = b_{ji} \qquad \forall(i,j),(i \ne j) \qquad (4)$$

Traffic Constraints

$$\lambda_{ij} \le \lambda_{max\_t\arg et}, \forall(i,j) \qquad (5)$$

$$\lambda_{ij} = \sum_{sd} \lambda_{ij}^{(sd)}, \forall(i,j) \qquad (6)$$

$$\lambda_{ij}^{(sd)} \le b_{ij}\lambda^{(sd)}, \forall(i,j),(s,d) \qquad (7)$$

$$\sum_j \lambda_{ij}^{(sd)} - \sum_j \lambda_{ji}^{(sd)} = \begin{cases} \lambda^{(sd)}, s=i \\ -\lambda^{(sd)}, d=i \\ 0, s\ne i, d\ne i \end{cases} \forall(s,d),i \qquad (8)$$

Wavelength Constraints

$$\sum_{k=0}^{W-1} c_{ij}^{(k)} = b_{ij}, \forall(i,j) \qquad (9)$$

$$c_{ij}^{(k)}(l,m) \le c_{ij}^{(k)}, \forall(i,j),(l,m),k \qquad (10)$$

$$\sum_{ij} c_{ij}^{(k)}(l,m) \le 1, \forall(l,m),k \qquad (11)$$

$$\sum_{k=0}^{W-1}\sum_m c_{ij}^{(k)}(l,m)p_{lm} - \sum_{k=0}^{W-1}\sum_m c_{ij}^{(k)}(m,l)p_{ml}$$
$$= \begin{cases} b_{ij}, m=i \\ -b_{ij}, m=j \\ 0, m\ne i, m\ne j \end{cases} \forall(i,j),m \qquad (12)$$

The degree constraints (2) and (3) constrain the logical topology to a given logical degree P. A bi-directional lightpath constraint is shown as (4). Among the traffic constraints, (5) defines the network congestion, different from previous study [5] we set a threshold λ_{max_target} on congestion. In previous study [5] congestion constraint is a minimum congestion min (λ_{max}) (min (λ_{max}) $\le \lambda_{max\_target}$) yield by first running of logical topology design phase, please see details in [5]. In fact, from the viewpoint of network operations the operation complexity has to be considered, if the congestion can be controlled below certain level, the target is to find a new logical topol-

ogy with smaller operation complexity. Here threshold λ_{max_target} can be tuned to satisfy certain requirement on the congestion. Expression (6) asserts that the total traffic on a lightpath is the sum of the traffic components on that lightpath due to all the different pairs of source and destination nodes. Constraint (7) captures the fact that the component of traffic on a lightpath due to a particular source destination pair can be present only if the lightpath exists in the logical topology, and cannot be more than the total traffic for that source-destination pair. Constraint (8) is an expression of the conservation of traffic flow at lightpath endpoints. Constraint (9) ensures that a lightpath if it exists in the logical topology has a unique wavelength out of the available ones. Constraint (10) enforces the consistency of the lightpath wavelength indicators and the link-lightpath wavelength indicators, and expression (11) enforces that a wavelength can be used at most once in every physical link, avoiding a wavelength clash. Expression (12) asserts the conservation of every wavelength at every physical link endpoint for each lightpath.

The inputs to the formulation are the traffic matrix Λ, the number of wavelengths supported by a fiber W, the desired logical degree P and the details of the physical topology graph. The variables, whose values at optimum are the output of the MILP, relate to the logical topology graph, wavelength assignment in the logical topology, and the traffic routing over the logical topology. The lightpath indicators b_{ij} provide the logical topology graph. The lightpath wavelength and link-lightpath wavelength indicators provide the wavelength assignments to the lightpaths in the logical topology and also the physical links used to implement each lightpath. Lastly, the virtual traffic load variables λ_{ij} and $\lambda_{ij}^{(sd)}$ provide the routing of the traffic between each source and destination on the logical topology.

3 Simulated Annealing Approach

Solving this MILP model quickly grows intractable with increasing size of the networks. In fact, this problem and some of its subproblems are known to be NP-hard [3][4]. Simulated annealing methods (SA) as a meta-heuristic methods were proposed to design logical topology in IP over WDM networks [3][10]. In particular, SA is one of the well-known meta-heuristics. It is based on a partial exploration of the space of admissible solutions, finalized to a good solution. In this paper we propose a simulated annealing approach to find logical topology reconfiguration with smaller amount of lightpath add/delete operations and lower congestion.

3.1 Heuristic Initial Solution

In general, the SA starts from a randomly generated initial solution. For a better solution, here we use a heuristic to design the initial logical topology. In the previous studies [2] several heuristics were proposed, for example, the authors attempt to place logical links between nodes in order of descending traffic which is called HLDA (heuristic logical topology design algorithm). Here we try a modified version of HLDA, which prepares a source-destination nodes pair list $Q=(q_{ij})$, and arranges the list Q in descending order of certain metric, then create the logical links. The sorting metric of the source-destination nodes pair list is shown as follows:

$$q_{ij} = \lambda^{(ij)} \times h^{ij} \qquad (13)$$

where $\lambda^{(ij)}$ and h^{ij} denote the traffic amount and the number of physical hops from source node i to destination node j. Expression (13) considers both traffic load and physical hops number. Intuitively, to design a logical topology that can minimize the network congestion, the lightpath should be created between those node pairs with heavy traffic and long hop distance. The other parts of initial heuristic including wavelength routing and assignment algorithm are given below:

Step1: Select the source destination pair (i_{max}, j_{max}) that satisfies $q_{imax,jmax} = \max_{ij} (q_{ij})$ for any node i and j ($i \neq j$); if all source-destination pairs with non-zero traffic have been tried, then go to step3;

Step2: If node i_{max} and j_{max} are of fewer degree than P (degree limitation at each node) then

Find the lowest available wavelength on the shortest propagation-delay path between i_{max}, j_{max} in physical topology (If there is more than one shortest path, scan them sequentially search the least loaded path);

If wavelength is available then
Create lightpath; $b_{imax,jmax} = 1$;
Else
$b_{imax,jmax} = 0$;
$q_{imax,jmax} = 0$; go to step1;
Else
$b_{imax,jmax} = 0$; $q_{imax,jmax} = 0$;
go to step1;

Step3: If we do not yet have $N \times P$ logical link, place as many remaining logical links as possible at random so that degree constraints are not violated and a wavelength can be found on the shortest path for the logical link. Otherwise end the algorithm.

Regarding the IP traffic routing subproblem, for the simplicity, after logical topology design, Dijkstra shortest path first (SPF) routing is used to route IP traffic on the logical topology. Then the network maximum congestion (load on the most congested logical link) in the logical topology is evaluated.

3.2 Two-Stage SA Algorithm

In [10] we have proposed a SA to design the logical topology. The indirect neighbor discovery (IND) method is efficient. In each iteration, we change the order of node pair list Q shown above by swapping the position of two randomly selected node pairs, then create lightpaths basing on the new node pair list (the procedure is that in section 3.1). A node pair list corresponds to a logical topology solution. Starting from a heuristic initial solution, the node pair list Q is changed step by step, and near optimal solution can be found in a shorter time. Here we use the IND as a basic algorithm to develop the SA that solve logical topology reconfiguration problem. As that of MILP model described in Section 2, we add a congestion threshold $C \leq \lambda_{max\_target}$ in SA. The multiple objectives are: 1) finding a solution with congestion $C \leq \lambda_{max\_target}$; 2) minimizing the distance D between the old logical topology and new logical topology, $D = \Sigma_{ij} |b_{ij} - b'_{ij}|$ (b'_{ij} is the lightpath indicator of the old logical topology, b_{ij} is the lightpath indicator of the new logical topology).

For an effective simulated annealing algorithm, one key point is to find a clue to get closer and closer from the starting point to an optimal solution, rather than a totally random search. The evaluation is a very important clue for optimal solution discovery in SA. Basing on this thinking, instead of treating congestion just as an extra constraint during logical topology distance D optimization, we make the congestion evaluation as a clue, and distance evaluation as another clue. So we separate the logical topology reconfiguration SA algorithm into two stages: congestion optimization stage and distance optimization stage, each with one clue (evaluation) congestion C and distance D respectively. The two-stage SA algorithm is described as follows:

Step1: Create the heuristic initial solution;

Step2: Congestion optimization stage SA-1. Find a solution that the congestion $C \leq \lambda_{max\_target}$. The evaluation metric is the congestion C in this stage. Improvement of C is a clue of exploration. If the solution is found then goto Step3, otherwise continue SA-1.

Step3: Distance optimization stage SA-2. Starting from the solution received in the first stage search the near neighbor solutions with smaller distance D and satisfy $C \leq \lambda_{max\_target}$. The evaluation metric is D. Then the clue is the improvement of D. If the congestion C of current solution is bigger than λ_{max_target}, then goto Step2 to find a new solution with lower congestion, otherwise continue SA-2;

Step4: The system temperature is cooled down $T=\alpha T$ ($0<\alpha<1$) through SA-1 and SA-2. If the system temperature is low enough then terminate the algorithm.

Within each SA stage, the acceptance of current solution is based on certain possibility. In congestion optimization stage SA-1, the acceptance possibility is shown as (14):

$$p(\Delta C) = \begin{cases} e^{(-\Delta C/T)} & (\Delta C > 0) \\ 1 & (\Delta C \leq 0) \end{cases} \quad (14)$$

where $\Delta C = C_c - C_i$, C_c denotes the congestion of current solution and C_i denotes the congestion of the ith solution. In distance optimization stage SA-2, the acceptance possibility is shown as (15):

$$p(\Delta D) = \begin{cases} e^{(-\Delta D/T)} & (\Delta D > 0) \\ 1 & (\Delta D \leq 0) \end{cases} \quad (15)$$

where $\Delta D = D_c - D_i$, D_c denotes the distance of current solution to the old logical topology and D_i denotes the distance of the ith solution to the old one. In SA-2 solution with minimum D will be recorded. If $\Delta D = 0$ and $\Delta C < 0$, current solution will be recorded as well. **Fig. 1** shows the transformation of tow-stage SA algorithm.
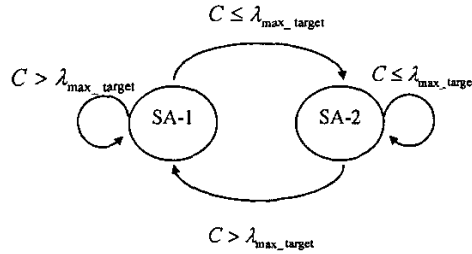

Fig. 1. Two-stage transformation and the conditions.

Regarding the selection of congestion threshold λ_{max_target}, we firstly compute a minimum congestion reference $\min(\lambda_{max})$ yielded by previously solving the congestion optimization logical topology design problem (for example we can use the IND SA or MILP), then we can balance the requirement of the congestion performance and operation complexity by tuning λ_{max_target} to a feasible value. Normally, λ_{max_target} is bigger than $\min(\lambda_{max})$, and a smaller λ_{max_target} will lead to a longer way to find a solution.

4   Numerical Results

In this section, we present three case studies on 6-node and 8-node ring networks and 14-node NSFNET network physical topology. We will give the numerical results of MILP model (by CPLEX 8.0) and our SA algorithm using randomly generated traffic patterns (but not the actual or absolute value of the traffic) with uniform distribution.

Three traffic conditions are considered for each case study. One for the computation of the old logical topology, the other two traffic patterns (Traffic-1 and Traffic-2) are given as the new traffic conditions, based on these•• new conditions, operators reconfigure the logical topology from old one. Here we assume that the traffic pattern is symmetrical, the traffic amount from node i to node j is the same as that from node j to node i ($i \neq j$). The lightpaths are also symmetrical, lightpaths between node i and node j are created along the same route in our SA. For the simplicity, we relax the wavelength amount restriction $W=100$ per fiber for all cases. •• We assume that during reconfiguration if a logical link is added or deleted we count the adding or releasing operation of this bi-direction ligtpath as one operation. If a logical link is not changed during reconfiguration, the wavelength routing and assignment of the lightpath are not changed. The major parameters are initial temperature $T_0=200$ and cooling parameter $\alpha=0.996$, the algorithm will be terminated when system temperature is lower than 10^{-6}.

Table 1 shows the reconfiguration results of 6-node ring network case. The congestion and operation results are shown as XX(YY) format, where XX is the optimal congestion after the reconfiguration and YY is the add/delete operation amount during the reconfiguration from the old logical topology to new one under the new traffic condition Traffic-1 and Traffic-2 respectively. For a small network, the degree constraint P is set to 3, and $\lambda_{max\_target} = \min(\lambda_{max})$. Table 2 shows the reconfiguration results of 8-node ring network case. For a small network, the degree constraint P is set to 3, and $\lambda_{max\_target} = \min(\lambda_{max})$.

Table 1 Congestion vs. operation amount after reconfiguration (6-node ring network)

	Traffic-1	Traffic-2
MILP	1276 (6)	1183 (8)
SA	1626 (3)	1335 (4)
Modified HLDA	1943 (0)	1355 (16)
no reconfiguration	1943 (0)	2003 (0)

Table 2 Congestion vs. operation amount after reconfiguration (8-node ring network)

	Traffic-1	Traffic-2
MILP	1939 (8)	2608 (4)
SA	2136 (7)	2741 (8)
Modified HLDA	2434 (8)	2825 (10)
no reconfiguration	2335 (0)	3559 (0)

In 6-node and 8-node ring network cases, we compare the MILP and SA approach. Also the results of congestion optimization heuristic (here it is a modified HLDA heuristic) and no reconfiguration are given as references. Because our SA uses shortest path first IP routing, it just gets a near optimal solution comparing to MILP approach. If we use heuristic to resolve the reconfiguration, it is difficult to

get an ideal solution comparing to MILP and SA approach.

Table 3 shows the reconfiguration results of NFSNET network case. The degree constraint P is set to 5, and λ_{max_target} is set to 2700, 2800, 2900, respectively. For a bigger size 14-node NSFNET network, the solution from MILP approach is not available in a feasible time, we just show the results yielded from our SA and compare them with that of heuristic and no reconfiguration references. In this case we introduce different congestion threshold, by relaxing the minimum congestion requirement, the distance between old logical topology and new one gets smaller and smaller comparing to the minimum congestion requirement ($\lambda_{max\_target} = \min(\lambda_{max})$). It shows that we can balance the optimal congestion requirement and operation complexity by tuning this threshold to a feasible value. In fact from the viewpoint of network operation, sometimes, the operation complexity should be considered more than the performance optimization if the congestion performance can be controlled below certain level.

Table 3 Congestion vs. operation amount after reconfiguration (NSFNET network)

	λ_{max_target}	Traffic-1	Traffic-2
MILP	-	-	-
SA	$\min\lambda_{max}$	2619 (32)	2165 (30)
SA	2700	2619 (32)	2554 (10)
SA	2800	2763 (21)	2620 (8)
SA	2900	2853 (20)	2620 (8)
Modified HLDA	-	4434 (26)	3788 (20)
no reconfiguration	-	4523 (0)	3893 (0)

5 Conclusions

The logical topology reconfiguration is usually formulated as a Mixed Integer Linear Programming, but this problem quickly grows intractable with increasing size of the network. In fact, this problem and some of its subproblems are known to be *NP-hard*. Thus, for networks of large sizes, it is not practical to attempt to solve this problem exactly. The feasible heuristic is very important especially for large networks. General heuristics have been studied in previous researches. To avoid falling into local optimum, as a meta-heuristic, simulated annealing method was proposed to design logical topology in WDM optical networks. In this paper we proposed a new simulated annealing algorithm to resolve the logical topology reconfiguration problem in IP over WDM networks. From performance comparisons, we have shown that with the new SA algorithm, ideal solution can be found especially for a bigger size network. For a bigger network both the MILP approach and general heuristic are not feasible. Also by introducing the threshold on congestion, we can balance the optimal congestion requirement and op-

eration complexity by tuning this threshold to a feasible value. From the viewpoint of network operation, sometimes, the operation complexity should be considered more than the performance optimization. For an effective solution discovery, a two-stage SA algorithm is developed for multiple objectives optimization.

6 Literature

[1] I.Chlamtac, A.Ganz and G.Karmi, "Lightpath communications: An approach to high bandwidth optical WANs," IEEE Transactions on Communications, vol.40, no.7, pp.1171-1182, July 1992.

[2] R.Ramaswami and 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.

[3] B.Mukherjee, D.Banerjee, S.Ramamurthy and A.Mukherjee, "Some principles for designing a wide-area wdm optical networks," IEEE/ACM Transactions on Networking, vol.4, pp.684-695, October 1996.

[4] B.Mukherjee, "WDM optical communication networks: Progress and challenges," IEEE Journal on Selected Areas in Communications, vol.18, no.10, pp.1810-1824, October 2000.

[5] 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, October 2000.

[6] B.Ramamurthy and A.Ramakrishnan, "Virtual topology reconfiguration of wavelength routed optical WDM networks," IEEE GLOBECOM 2000, pp.1269-1275, November 2000.

[7] S.Xu, K.Sezaki and Y.Tanaka, "A Heuristic Method of Logical Topology Reconfiguration in IP/WDM Optical Networks," 10th International Conference on Telecommunications (ICT), pp.6-11, February 2003.

[8] G.L.Nemhauser, A.H.G.Rinnoy Kan, M.J.Todd, "Optimization - handbooks in operations research and management science," vol.1, North-Holland, 1989.

[9] T.Miyoshi, S.Shimuzu and Y.Tanaka, "Fast topological design with simulated annealing for multicast networks," 7th IEEE Symposium on Computers and Communications (ISCC 2002), Italy, pp.959-966, July 2002.

[10] S.Xu, N. Koyama and Y.Tanaka, "A Logical Topology Design with Simulated Annealing in Wavelength Routed IP Networks", International Network Optimization Conference 2003, France, pp.587-591, October 2003.