Topological Design of Multihop Lightwave Networks*

Cem Ersoy

Computer Engineering Department
Boğaziçi University,Istanbul, Turkey

Shivendra S. Panwar

Electrical Engineering Department
Polytechnic University, New York, USA

Abstract

One of the promising lightwave network architectures is the multihop lightwave networks based on wavelength division multiplexing (WDM). In this architecture, separate channels created by assigning different wavelengths between node pairs define the logical connectivity. This logical topology is independent of the underlying physical topology and can be adaptively changed according to the changing traffic patterns and connectivity requirements. One of the possible objectives for designing this logical topology is minimizing the maximum utilization on any link by choosing the logical connectivity and finding the optimum routing. This paper describes an algorithm based on simulated annealing for solving the joint logical topology design and routing problem. Different algorithms were implemented for the cases with bifurcated and shortest path routing. Computational experiments showed that the simulated annealing solutions are as good as or better than the solutions found in previous studies.

1 Introduction

Telecommunications networks using lightwave technology have become very attractive because of the large bandwith potential of the optical fiber. One of the promising architectures is the multihop lightwave network based on wavelength divison multiplexing (WDM) [1]. In this architecture, all nodes are connected to the same optical medium which may have different physical topologies, such as bus, star or tree as shown in Figure 1. Although all users are connected to the same medium, WDM scheme takes advantage of the large bandwith of the fiber by allowing many concurrent channels. Each node has a small number of transmitters and receivers. One wavelength is dedicated for each channel between two users. Because of the limitation on the number of transmitters and receivers at each user, there is no direct channel between all user pairs. Thus, packets destined to a node may reach their destinations by hopping through a sequence of intermediate nodes. Separate channels created by assigning different wavelengths to receiver transmitter pairs defines the logical connectivity. This logical topology is independent of the underlying physical topology and can be changed by a different wavelength assignment.

*This work was supported by the New York State Science and Technology Foundation's Center for Advanced Technology in Telecommunications, Polytechnic University, Brooklyn, New York and by the NSF under grant NCR-9115864.

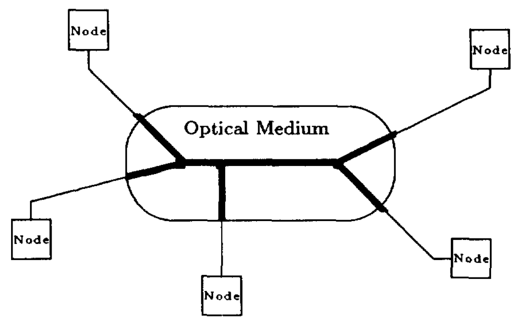

Figure 1: An example physical topology

The original logical topology, ShuffleNet, was proposed by Acampora, Hluchyj and Karol [1] and had several stages connected by a perfect shuffle as shown in Figure 2. Each directional link in this connectivity pattern corresponds to a channel created by a wavelength carrying signals between two user nodes. Acampora *et al.* have shown that ShuffleNet achieves high efficiency for uniform traffic loads [1]. Eisenberg and Mehravari have proved that the performance of the architecture degrades under nonuniform traffic requirements [2]. Different logical topologies and topological design techniques have been proposed for answering weaknesses of the perfect shuffle topology under unbalanced traffic requirements. Sivarajan and Ramaswami have shown that de Brujin graphs as logical topologies can support significantly more nodes than a ShuffleNet [3]. Bannister, Fratta and Gerla have described techniques for the topological design of the Wavelength-Division Optical Network, a generalization of the ShuffleNet concept in which the logical connectivity can be adjusted by changing the wavelength assignment [4].

Recent availability of slowly tunable lasers and receivers makes it easier to change the wavelength assignments. This enables the network manager to adapt new logical topologies according to the changes in the traffic requirements or in the case of component failures. However, because of the slow response time of the lasers and receivers, the reconfiguration is also slow and not suitable for instantaneous changes.

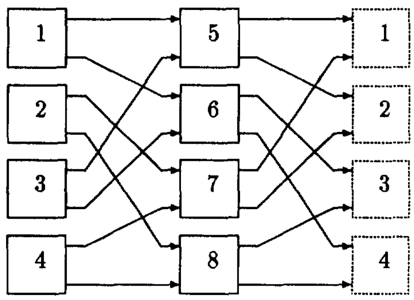

Figure 2: The perfect shuffle connectivity diagram

Labourdette and Acampora introduced a logical topology design problem in which the transmitters and receivers are slowly tunable [5]. The problem consists of finding a logical connectivity diagram (wavelength assignment) and routing the traffic (flow assignment) so that the maximum utilization on any link is minimized. In this problem, the number of transmitters and receivers at each node is limited to a small number T, and the routing is bifurcated. In other words, the traffic between a source-destination pair can be split into different paths. They decoupled the problem into wavelength assignment and flow assignment problems in. They proposed a two phase heuristic algorithm based on linear programming and branch exchange operations.

We applied simulated annealing to the problem introduced in [5]. We implemented two variations of the simulated annealing algorithm with optimum bifurcated and shortest path routing. Section 2 contains a more detailed definition and the formulation of the problem. The description of the simulated annealing algorithm is given in Section 3. Comparisons of the annealing solutions to the lower bounds and to the solutions found in [5] are presented in Section 4. Conclusions are presented in Section 5.

2 Formulation of the Problem

Each directional link on the logical connectivity diagram in Figure 2 corresponds to a channel created by assigning a unique wavelength for each source and destination pair. Given a logical connectivity pattern, the packets will be routed over the links of this logical topology. The problem consists of finding a logical topology and routing the traffic so that the maximum utilization on any link $\rho^M = \max_{(i,j)} \frac{\lambda_{ij}}{C_{ij}}$ is minimized for a given set of traffic requirements (t_{sd}'s), where λ_{ij} and C_{ij} are the flow and capacity of the link from Node i to Node j and λ_{ij}/C_{ij} is taken to be zero if $C_{ij} = 0$. The number of transmitters and receivers at each node (T) is fixed. This min-max performance measure corresponds to the potential increase in throughput for a given traffic pattern because the optimal solution will not change if the traffic requirements are scaled up by the greatest factor without exceeding the link capacities. This measure does not reflect the propagation delay and will not be suitable for wide-area networks. However, it reflects some aspects of the transmission delay, because it has been noted that this measure and M/M/1 type delay measure give similar results for heavy load cases [6]. In the problem described in [5], the routing was bifurcated. We also consider the non-bifurcated routing case where traffic is sent on shortest-path routes. The flow and wavelength assignment problem without any constraints on routing is formulated as follows:

Problem P

$$\text{minimize } \rho^M = \max_{(i,j)} \frac{\lambda_{ij}}{C_{ij}} \tag{1}$$

subject to

$$\sum_j \lambda_{ij}^{sd} - \sum_j \lambda_{ji}^{sd} = \begin{cases} t_{sd}, & \text{if } s = i \\ -t_{sd}, & \text{if } d = i \\ 0, & \text{otherwise} \end{cases} \quad \forall i, s, d \tag{2}$$

$$\lambda_{ij} = \sum_{s,d} \lambda_{ij}^{sd} \quad \forall i, j \tag{3}$$

$$\lambda_{ij} \leq C_{ij} \quad \forall i, j \tag{4}$$

$$C_{ij} = x_{ij} C \quad \forall i, j \tag{5}$$

$$\sum_j x_{ij} = T \quad \forall i \tag{6}$$

$$\sum_i x_{ij} = T \quad \forall j \tag{7}$$

$$C_{ij} \geq 0 \quad \forall i, j \tag{8}$$

$$\lambda_{ij}^{sd}, \lambda_{ij} \geq 0 \quad \forall i, j, s, d \tag{9}$$

$$x_{ij} \text{ is } integer \quad \forall i, j \tag{10}$$

where the logical connectivity is defined by the decision variables x_{ij}'s which are equal to the number of directed connections from Node i to Node j; and the flow assignment is defined by λ_{ij}^{sd}'s which describe the flow on link (i, j) due to t_{sd}.

Constraints (2) and (3) are flow conservation equations. Constraints (4) are the capacity constraints. Constraints in (5) specify link capacities as a function of the number of connections (all channel capacities are assumed to be equal to C). Constraints (6) and (7) specify that the number of transmitters and receivers at each node is equal to T. (8), (9) and (10) are non-negativity and integrality constraints. If needed, routing constraints can be added to the formulation.

Problem P is a mixed integer optimization problem with a min-max type objective function. It is a difficult optimization problem as noted in [5]. Since the search space grows at least as fast as $N!$ (when $T = 1$), it is impractical to determine the optimum logical topology by exhaustive search even for small values of T and N. The heuristic algorithm proposed by Labourdette and Acampora in [5] was based on finding an initial solution using linear programming and performing a local search by applying branch-exchange operations. We apply simulated annealing to the logical topology design problem. We compare the solutions found by the annealing algorithm to the lower bounds and to the solutions of the algorithm described in [5].

3 Simulated Annealing

Simulated annealing is a local neighborhood search technique [7]. Two basic disadvantages of ordinary local search algorithms are that they may get stuck in local minima because they accept only cost improving solutions and the quality of the final result heavily depends on the initial solution. In contrast, simulated annealing algorithms occasionally accept deteriorations in cost in a controlled manner besides accepting improvements in cost. This property enables them to escape from local minima while keeping the favorable features of local search algorithms, i.e., simplicity and general applicability. The simulated annealing algorithm used for solving the multihop lightwave network design problem is as follows:

1. Initialize *Current Topology* = Perfect Shuffle Topology
2. Find initial value of the control parameter, c
3. *New Topology* = Neighbor of *Current Topology*
4. ρ^M = Maximum utilization on any link on *New Topology*
5. $\Delta\rho^M = \rho^M(New\ Topology)$ - $\rho^M(Current\ Topology)$
6. **If** $\Delta\rho^M < 0$ **then** *Current Topology* = *New Topology*;
   else *Current Topology* = *New Topology* with probability $e^{-|\Delta\rho^M|/c}$
7. Decrement control parameter c **if** necessary
8. **Go to 3 if** the stopping criterion is not satisfied
9. **If** $\rho^M(Best\ Topology\ so\ far) < 1$ **then** the solution is feasible;
   else no feasible solution

We use the perfect shuffle topology shown in Figure 2 as the initial logical topology. During the search for an optimum topology, a sequence of neighbor topologies are created. Any two connectivity diagrams which differ by only two links and satisfy the degree constraints (6) and (7) are accepted as neigbor topologies. The link flows are found and the feasibility of each topology is checked by comparing the flows on each link with their capacities.

Given a logical topology, flows on links are dependent on the routing schedule. We have experimented with optimized bifurcated routing and shortest path routing which is not optimized but simpler and more suitable for networks where resequencing of packets is not desirable. After finding the paths for all source-destination pairs, the flows on all links are found. We can find the maximum utilization on any link after dividing the flow on each link to the capacity of that link. In order to check the feasibility of a given logical topology, it is sufficient to check if the capacity constraint (4) is satisfied for the link with the maximum utilization, since all other links have lower utilization. In the case of bifurcated routing, link flows are optimized using a variation of the flow deviation algorithm described in [5], [8]. Since the results of the algorithm were close to the optimum and running times were reasonable, we could embed the flow deviation into the simulated annealing algorithm.

The method of reducing the control parameter c is called the *cooling schedule*. We experimented with different cooling schedules and chose one similar to the one described in [9], because of its simplicity and efficiency. The initial value of the control parameter, c_0, is chosen so that almost all new topologies are accepted at the beginning. The control parameter is decreased linearly after acceptance of N new topologies. Simulated annealing is terminated if the value of the performance criterion does not change after decrementing the control parameter a fixed number of times.

4 Computational Results

4.1 Methodology for the Experiments

We tested the simulated annealing algorithm on the same example problems described in [5]. In these problems, there were $N = 8$ nodes and $T = 2$ transmitters and receivers per node. Each problem had a different type of traffic matrix. These matrices were given in [5] and as follows: **Uniform** in which all traffic requirements are the same; **Quasiuniform** in which the requirements are approximately the same; **Ring** in which the requirements are larger between Nodes 1 and 2, Nodes 2 and 3, Nodes 3 and 4, etc.; **Centralized** in which all the requirements to and from Node 1 are larger than the others; **Disconnected** in which the requirements are larger among Nodes 1,2,3,4 and among Nodes 5,6,7,8.

We found solutions for the cases with shortest path routing and bifurcated routing. We compared these solutions to the lower bounds and to the solutions found in [5] which were obtained by restricting the decision variables (x_{ij}'s) to 0 or 1. In other words, the connectivity diagrams were restricted to have at most one directed connection between any two nodes. In order to have a fair comparison, we restricted our solutions the same way for the 8-node problems

Table 1: Results with Bifurcated Routing

# of Nodes	Traffic Matrix	$\rho^M_{Perfect\ Shuffle}$	Annealing		$\frac{\rho^M_{PS} - \rho^M_{min}}{\rho^M_{PS}}$	$\rho^M_{Lower\ Bound}$	$\frac{\rho^M_{min} - \rho^M_{LB}}{\rho^M_{LB}}$	ρ^M_{LA}	$\frac{\rho^M_{LA} - \rho^M_{LB}}{\rho^M_{LB}}$	$\frac{\rho^M_{min} - \rho^M_{LA}}{\rho^M_{LA}}$
			ρ^M_{min}	ρ^M_{max}						
8	Uniform	0.143	0.143	0.143	0%	0.116	23.0%	0.143	23.3%	0%
8	Quasi	0.153	0.125	0.128	18.3%	0.112	11.6%	0.124	10.7%	0.8%
8	Ring	0.182	0.106	0.109	41.8%	0.084	26.2%	0.105	25.0%	0.9%
8	Centr	0.201	0.197	0.198	2.0%	0.195	1.0%	0.195	0%	1.0%
8	Discon	0.175	0.111	0.115	36.6%	0.088	26.1%	0.119	35.2%	-6.7%
16	Random	-	0.094	0.096	-	0.071	32.3%	-	-	-
32	Random	-	0.060	0.061	-	0.043	39.5%	-	-	-

described above. We also recorded the improvement of the annealing algorithm over the originally proposed perfect shuffle connectivity diagram. We ran the simulated annealing algorithm 10 times with different random seeds on each problem. The difference between the best and the worst solutions in 10 runs gave us the range for the annealing solutions.

In order to see the scaling of the algorithm, we also experimented with 16 and 32 Node problems. Again, T was equal to 2. Initial topologies and traffic matrices for these problems were randomly generated. We checked the quality of the solutions for larger problems using a statistical goodness measure. In this measure, 10,000 random feasible topologies were generated. A histogram corresponding to the ρ^M values of these topologies was created. This was then compared with the simulated annealing solutions.

4.2 Lower Bounds

Two lower bounds on the maximum flow in a logical topology (F_1^{min} and F_2^{min}) have been derived in [5]. In order to find F_1^{min}, first, we have to find the logical topology which carries in one hop the maximum amount of traffic (S^*) over TN links. A lower bound on the total carried traffic is given by

$$T_f = S^* + 2S_2 + 3S_3 + \cdots + kS_k + \cdots + pS_p$$

where S^* is as described above, and S_k is the sum of the T^kN largest traffic requirements not included in S^* or S_j, for $j < k$. By assuming that the total traffic T_f can be evenly spread among the TN links, a lower bound F_1^{min} on the maximum flow on any link can be obtained

$$F_1^{min} = \frac{T_f}{TN}.$$

F_2^{min} is found by evenly spreading the total flow incoming to or outgoing from a node on its links and picking the maximum of this value among all nodes

$$F_2^{min} = \frac{1}{T} \sup \left(\max_i \left\{ \sum_j \lambda_{ij} \right\}, \max_j \left\{ \sum_i \lambda_{ij} \right\} \right).$$

Another lower bound for the non-bifurcated routing case is F_3^{min}. The maximum flow on any link cannot be less than the largest traffic requirement (t_{sd}) since at least one link has to carry this requirement. Thus,

$$F_3^{min} = \max_{\forall (s,d)} t_{sd}.$$

Lower bounds for the maximum utilization on any link can be found by dividing the lower bounds for maximum flows to the maximum capacity on any link. We used the tightest of these three lower bounds for each case.

4.3 Results

The results of the annealing algorithm with optimized bifurcated routing are given in Table 1. These results are normalized by choosing the link capacity $C = \sum_s \sum_d t_{sd}$. Related columns show the best and the worst solutions found by the annealing algorithm and the lower bounds for each problem. The range of the annealing solutions were small. Final topologies were improved up to 41.8% compared to the perfect shuffle topology. The gaps between the lower bounds and the annealing solutions were between 1% and 39.5%.

The last three columns of Table 1 summarizes the results of the comparison between the solutions of the simulated annealing algorithm and those found in [5]. All example problems in [5] had 8 nodes. In general, both algorithms performed similarly. In three of the four cases with nonuniform traffic requirements, the solutions found by Labourdette and Acampora were less than 1% better. In the case with the disconnected type traffic matrix, simulated annealing performed 6.7% better. This was a special case in which the algorithm in [5] performed poorer. Better performance of the annealing algorithm on this case indicates that the annealing solutions are less dependent on the type of non-uniformity of the traffic requirements.

The results with shortest path routing are summarized in Table 2. Again, the range of the annealing solutions is small. As shown in the fourth column of Table 2 ($\Delta_{PS,min}$), the annealing algorithm improved the performance over the Perfect Shuffle topology by 21.9% to 45.3% for all cases with nonuniform traffic requirements. Annealing solutions were between 2.5% to 46.5% above the lower bounds as shown in the last column of Table 2. As expected, the solutions for the bifurcated case were better

Table 2: Results with shortest path routing

# of nodes Matrix Type	Annealing ρ^M_{min}	Annealing ρ^M_{max}	Δ PS, min	Lower Bound	Δ min, LB
8 Quasi	0.133	0.135	33.5%	0.112	18.7%
8 Ring	0.117	0.121	45.3%	0.096	21.8%
8 Centr	0.203	0.208	21.9%	0.198	2.5%
8 Discon	0.119	0.121	36.7%	0.088	35.2%
16 Random	0.097	0.102	-	0.071	36.6%
32 Random	0.063	0.065	-	0.043	46.5%

Table 3: Summary of the histograms for ρ^M

# of nodes & routing	Annealing		10000 Random Topologies			
	min	max	min	max	mean	σ
16 sh path	0.097	0.102	0.116	0.298	0.172	0.021
16 bifurc	0.091	0.094	0.103	0.285	0.146	0.020
32 sh path	0.063	0.065	0.075	0.189	0.105	0.011
32 bifurc	0.060	0.061	0.070	0.175	0.094	0.010

than those for the shortest path case. However, the difference between the quality of solutions obtained using different routing methods were smaller for larger size problems. This can be explained with the fact that non-bifurcated flows are closer to the optimum bifurcated flows for larger size networks [8].

We compared the solutions of 8 node problems to those found in [5]. For larger problems, we used a statistical goodness measure in which 10000 random logical connectivity diagrams were generated. ρ^M's of these topologies were found using different routing schemes for each case. Histograms for 16 and 32 node topologies with shortest path and bifurcated routing are summarized in Table 3. The last column shows the standard deviation (σ) of ρ^M's. The simulated annealing algorithm found better topologies than the best of 10,000 randomly generated topologies for all cases. Majority of the random topologies had considerably larger ρ^M's than that of the annealing solutions.

The average CPU time required for the 8 node case with shortest path routing was 1 minute on a PC/AT and 0.9 seconds on a CONVEX 120 mini-computer. Doubling the size of the problem approximately twenty folded the running times for the shortest path cases. The ratio of the running times of the same size problem with different routing methods is approximately equal to the average number of iterations of the flow deviation algorithm. We could not compare the algorithms in terms of running times because they were not specified in [5]. However, both algorithms have mechanisms to adjust the compromise between the quality of the final solution and the running time.

5 Conclusion

We have described a method based on simulated annealing for solving the joint logical topology design and routing problem for multihop lightwave networks with the objective of minimizing the maximum utilization on any link. We have implemented simulated annealing algorithms for the cases with bifurcated and shortest path routing. Computational experiments have shown that the simulated annealing algorithm could improve the performance over the Perfect Shuffle topology for nonuniform traffic cases; simulated annealing solutions are as good as or better than the solutions found in [5] and less dependent on the type of non-uniformity of the traffic requirements. Moreover, annealing solutions are not very far from the lower bounds. For larger problems, the algorithm found better topologies than the best of 10,000 random topologies and as the histograms indicate majority of the random topologies had considerably poorer performance compared to the annealing solutions. Other performance criteria and constraints, such as delay and reliability constraints, can be used in the formulation of the problem. The simulated annealing algorithm can easily be modified to handle these variations.

References

[1] A. S. Acampora, M. J. Karol, and M. G. Hluchyj, "Multihop lightwave networks: A new approach to achieve terabit capabilities," in *Proc. of IEEE Int. Conf. on Comm.*, pp. 1478–1484, 1988.

[2] M. Eisenberg and N. Mehravari, "Performance of the multichannel multihop lightwave network under nonuniform traffic," *IEEE JSAC*, pp. 1063–1077, August 1988.

[3] K. Sivarajan and R. Ramaswani, "Multihop lightwave networks based on De Brujin graphs," in *Proc. of the IEEE INFOCOM*, pp. 1001–1011, 1991.

[4] J. A. Bannister, L. Fratta, and M. Gerla, "Topological design of the wavelength-division optical network," in *Proc. of the IEEE INFOCOM*, pp. 1005–1013, 1990.

[5] J. P. Labourdette and A. S. Acampora, "Rearrangeable multihop lightwave networks," *IEEE Tr. on Communications*, pp. 1223–1230, August 1991.

[6] D. Bertsekas and R. Gallager, *Data Networks, 2nd Ed.* Prentice-Hall, 1992.

[7] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, pp. 671–680, May 1983.

[8] L. Fratta, M. Gerla, and L. Kleinrock, "The flow deviation method: An approach to store-and-forward communication network design," *Networks*, pp. 97–133, 1973.

[9] E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines.* John Wiley & Sons, 1989.