A Framework for Solving Logical Topology Design Problems Within Constrained Computation Time

Andrew Zalesky, Hai Le Vu, *Member, IEEE*, Moshe Zukerman, *Senior Member, IEEE*, and Iradj Ouveysi

***Abstract*—We present a framework for solving logical topology design (LTD) problems in a constrained amount of computation time. Our framework uses a search space dimensionality (SSD) reduction technique that exploits a tradeoff between computation time and solution quality. We have demonstrated that our framework offers improved solution quality in comparison to an existing SSD reduction technique reported in the literature.**

***Index Terms*—Logical topology design, mixed integer linear programming, optical networks, routing.**

I. Introduction

In OPTICAL networks, the interconnection of nodes via optical fibers is defined as the *physical topology*. Nodes not adjacent to each other in the physical topology can communicate through wavelength-routed all-optical channels known as *lightpaths*. Data transmitted on a lightpath is carried optically in the physical topology from end to end via a contiguous sequence of wavelengths. Optical cross-connects (OXC) located at each intermediate node along a lightpath transparently couple an incoming wavelength with an outgoing wavelength without requiring any electronic processing of the transmitted data. The interconnection of nodes via lightpaths is defined as the *logical topology*.

The limited number of input/output ports on an OXC usually precludes the design of a fully meshed logical topology. In a partially-meshed logical topology, source–destination (SD) pairs that are not optically connected with a single lightpath can communicate via a contiguous sequence of lightpaths known as a *route*. Data transmitted on a route is electronically switched to each subsequent lightpath enroute to the destination node.

Logical topology design (LTD) is an optimization problem that can be stated as follows. Given: 1) a physical topology in which each node is equipped with an OXC consisting of an integer number of input ports and the same integer number of output ports; and 2) a traffic demand matrix whose elements quantify the average traffic exchanged between SD pairs. Find an optimal: 1) logical topology (routing for lightpaths over the physical topology); 2) routing for the traffic demand over the logical topology; and 3) wavelength assignment for lightpaths. A common optimization objective is to minimize the maximum level of congestion. *Congestion* is defined as the maximum traffic flow on any lightpath established in the logical topology.

The LTD optimization problem is typically formulated as a mixed integer linear program (MILP). Optimally solving the MILP for networks of practical size in a reasonable amount of computation time is numerically intractable.

Heuristics are typically employed to generate approximate solutions to intractable instances of the LTD problem. See [4] for a survey of the available heuristics. The complicatedness of heuristics has motivated the development of alternative solution approaches. For example, in [5], a simple edge disjoint path algorithm is shown to offer solutions comparable to that of more complicated heuristics. Other approaches are discussed in [1], [3] and [6].

We propose a framework for solving LTD problems in a constrained amount of computation time. Our framework entails a search space dimensionality (SSD) reduction technique that exploits a tradeoff between computation time and solution quality. SSD reduction is achieved by excluding certain subsets from the search space. Each subset is identified with a distinct feature that solutions of the LTD problem may exhibit. Based on the maximum amount of computation time permitted, our framework pinpoints subsets of the search space that require exclusion. Excluding these subsets ensures that a higher quality solution is achievable, within a given maximum computation time, in comparison to exploring the entire search space.

In Section II, we discuss cyclic routes—an undesirable feature that solutions of the LTD problem may exhibit. We formulate the LTD optimization problem as a MILP in Section III. For the purposes of this paper, we do not consider the wavelength assignment subproblem; i.e., the number of wavelengths and their lightpath assignment. Section IV is devoted to our framework. In Section V, we demonstrate our framework with an example.

II. Cyclic Routes

Unlike heuristics, our framework allows designs to be tailored to exclude certain undesirable features such as cyclic routes. A *cyclic* route utilizes more than one output port or more than one input port on any particular OXC in the network. For example, in Fig. 1, traffic from nodes 1 to 3 is offered two possible routes. The route consisting of the single lightpath $1 \rightarrow 2 \rightarrow 3$ is noncyclic (*acyclic*), however, the alternative route consisting of two lightpaths $1 \rightarrow 2$ and $2 \rightarrow 1 \rightarrow 3$ is cyclic because it utilizes two output ports on the OXC located at node 1.

Although optimal solutions may exhibit cyclic routes [2], it is likely that optimal solutions will not be found in a limited amount of computation time. Pruning from the search space the

Manuscript received March 11, 2003. The associate editor coordinating the review of this letter and approving it for publication was Dr. I. Fair. This work was supported by the Australian Research Council.

A. Zalesky, H. Le Vu, and M. Zukerman are with the ARC Special Research Centre for Ultra-Broadband Information Networks, Department of Electrical and Electronic Engineering, The University of Melbourne, Melbourne, Vic. 3010, Australia (e-mail: alz@ee.mu.oz.au; h.vu@ee.mu.oz.au; m.zukerman@ee.mu.oz.au).

I. Ouveysi is a Fellow with the Department of Electrical and Electronic Engineering, The University of Melbourne, Melbourne, Vic. 3010, Australia (e-mail: Iradj.Ouveysi@ bigpond.com).

Digital Object Identifier 10.1109/LCOMM.2003.818883

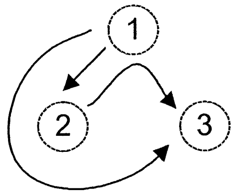

Fig. 1. Example of cyclic and acyclic routes. Each arc represents a lightpath.

subset identified with cyclic routes ensures a better solution for the given computation time in comparison to exploring the entire search space.

III. Problem Formulation

In this section, the LTD optimization problem is formulated as a MILP. The wavelength assignment subproblem is not considered. We introduce the following notation for indexing variables:

- m SD pair needing bandwidth;
- n node in the network;
- l lightpath in the logical topology;
- p route between SD pair.

A. Given Parameters

The following sets are provided as input to the MILP:

- $\mathcal{M}$ SD pairs in the network;
- $\mathcal{N}$ nodes in the network;
- $\mathcal{L}$ all possible lightpaths;
- $\mathcal{P}_m$ all possible routes between SD pair $m \in \mathcal{M}$;
- $\mathcal{H}_m^l$ $(\mathcal{H}_m^l \subset \mathcal{P}_m)$ routes between SD pair m that traverse lightpath $l \in \mathcal{L}$;
- $\mathcal{D}_n^{\text{in}}$ $(\mathcal{D}_n^{\text{in}} \subset \mathcal{L})$ lightpaths incident to node $n \in \mathcal{N}$;
- $\mathcal{D}_n^{\text{out}}$ $(\mathcal{D}_n^{\text{out}} \subset \mathcal{L})$ lightpaths incident from node n.

$\mathcal{L}$ can be determined by originating a breadth first search at each node. For each search, a queue of lightpaths is maintained, initially containing a zero length lightpath from the originating node to itself; continue by repeatedly removing a lightpath from the queue and adding all one fiber extensions of that lightpath to the queue and to $\mathcal{L}$. The search terminates if there are no lightpaths in the queue that cannot be extended by one fiber without resulting in a cycle.

$\cup_{m\in\mathcal{M}}\mathcal{P}_m$ is determined similarly by originating a search at each lightpath in $\mathcal{L}$. Excluding cyclic routes from $\cup_{m\in\mathcal{M}}\mathcal{P}_m$ can be achieved by checking that a route does not traverse a node twice when extending that route in the queue by one lightpath. Hop limits can be imposed to ease the considerable computational burden in determining $\mathcal{L}$ and $\cup_{m\in\mathcal{M}}\mathcal{P}_m$.

The following parameters are also given. Δ is the maximum degree of the logical topology; i.e., the maximum number of input/output ports available on an OXC and Λ_m is the traffic demand between SD pair m.

B. Variables

The MILP is formulated with the following fractional and binary variables:

- $\lambda_{m,p}$ $(0 \le \lambda_{m,p} \le 1)$ is the portion of traffic between SD pair m aggregated to path $p \in \mathcal{P}_m$;
- λ $(0 \le \lambda \le \max_{m\in\mathcal{M}} \Lambda_m)$ is the maximum amount of traffic carried by any logical link in $\mathcal{L}$;
- α_l $\alpha_l = 1$ if lightpath l is used in the optimal logical topology, otherwise $\alpha_l = 0$.

C. Objective Function

$$\min \lambda.$$

The objective seeks to minimize the maximum level of congestion in the network.

D. Constraints

$$\sum_{p\in\mathcal{P}_m} \lambda_{m,p} = 1 \;\forall\, m \in \mathcal{M} \tag{1}$$

$$\lambda_{m,p} \le \alpha_l \;\forall\, l \in \mathcal{L}, \qquad m \in \mathcal{M};\; p \in \mathcal{H}_m^l \tag{2}$$

$$\sum_{m\in\mathcal{M}} \sum_{p\in\mathcal{H}_m^l} \lambda_{m,p}\Lambda_m \le \lambda \;\forall\, l \in \mathcal{L} \tag{3}$$

$$\sum_{l\in\mathcal{D}_n^{\text{in}}} \alpha_l \le \Delta \;\forall\, n \in \mathcal{N} \tag{4}$$

$$\sum_{l\in\mathcal{D}_n^{\text{out}}} \alpha_l \le \Delta \;\forall\, n \in \mathcal{N}. \tag{5}$$

Constraint (1) forces traffic between SD pair m to be partitioned across a subset of routes in $\mathcal{P}_m$; constraint (2) ensures that a lightpath is established if it used by any route that is assigned a portion of traffic; and constraint (3) quantifies network congestion. Note that the term $\lambda_{m,p}\Lambda_m$ is the amount of traffic between SD pair m aggregated to route $p \in \mathcal{P}_m$. Constraints (4) and (5) ensure that the number of lightpaths incident to and incident from node n do not exceed Δ.

IV. Framework

In this section, we describe a framework based on SSD reduction for solving the above MILP in a constrained amount of computation time. Our framework achieves SSD reduction by pruning certain subsets from the search space. Each subset is identified with one of the following three distinct features that a solution to the LTD problem may exhibit:

— Feature 1: cyclic routes;
— Feature 2: nonshortest lightpaths between nodes;
— Feature 3: nonshortest routes between SD pairs.

Note that the term 'nonshortest' refers to the complimentary set of *all* the shortest lightpaths (or routes). Our framework prescribes an appropriate feature to prune from the search space based on the following three criteria:

1) SSD of the MILP;
2) maximum amount of computation time permitted;
3) predefined features the solution must exhibit.

The total number of enumerated routes $|\cup_{m\in\mathcal{M}} \mathcal{P}_m|$ is used to approximate the SSD of the MILP. The MILP is considered *solvable* if the number of routes enumerated is below a certain threshold. This threshold is chosen depending on the maximum amount of computation time permitted. Our computational experience with the MILP we formulate shows that for a given instance of the LTD problem, the time of computation is approximately proportional to the number of enumerated routes $|\cup_{m\in\mathcal{M}} \mathcal{P}_m|$. Although a precise relation between computation time and $|\cup_{m\in\mathcal{M}} \mathcal{P}_m|$ is unlikely to exist, our approximation based on observing the computation time by sufficiently varying the number of enumerated routes is adequate for the purposes of our framework.

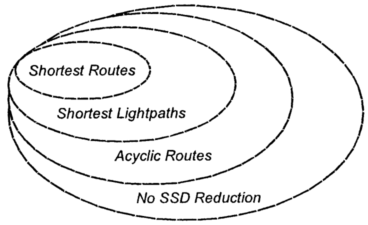

Fig. 2. Partitioning of the search space. Each subset is identified with a distinct feature.

Each of the three features offers a different amount of SSD reduction; i.e., the number of enumerated routes after pruning is different for each feature. This allows our framework to exploit a tradeoff between computation time and solution quality. In particular, pruning Feature 1 reduces the SSD by the least, thus emphasizing solution quality. On the other hand, pruning Feature 3 reduces the SSD by the most, thus emphasizing a rapid computation time. Pruning Feature 2 offers a balance between these two competing factors. Fig. 2 depicts the partitioning of the search space.

Our framework can be summarized into three steps.

Step 1: Prune from the search space all subsets that are identified with undesirable features that the solution must not exhibit.

Step 2: Approximate the SSD and determine if the MILP is solvable; that is, determine if the number of routes enumerated is below the threshold corresponding to the maximum computation time permitted. If so, solve the MILP and stop, otherwise consider the next step.

Step 3: Sequentially prune an additional feature from the search space in the order Feature (1) to Feature (3) and repeat the previous step.

V. Numerical Example

In this section, we demonstrate the usefulness of our framework for solving the LTD problem on the well-known NSFNET, see [2] for the network graph.

The MILP is established as follows. We randomly generate a traffic demand matrix Λ_m from a uniform distribution on [0, 1]. The maximum degree of the logical topology Δ is fixed at six. All lightpaths are limited to a hop limit of two. All routes between SD pairs are limited to three or fewer lightpaths. It follows that all routes traverse no more than six edges in the physical topology. This in itself reduces the SSD.

The CPLEX optimization package is used to compute the best feasible solution in a limited amount of computation time, which is quantified by CPU time. Table I shows the minimum level of network congestion achievable by pruning from the search space each of the three features identified in Section IV. Each numerical entry in Table I is obtained by terminating the branch-and-bound algorithm after the maximum amount of computation time is exceeded, and then finding the solution corresponding to the best feasible node of the branch-and-bound tree. It is likely that this will not be the optimal solution to the MILP. A feasible solution is not recoverable in the specified amount of computation time for entries marked with an asterisk.

Table I shows how our framework can be used to exploit the tradeoff between solution quality and computation time. Each threshold is approximated by noting the time limit when including a certain feature results in a lower level of congestion.

TABLE I
MINIMUM LEVEL OF NETWORK CONGESTION

Time Limit (s)	*No Cyclic Routes*	*Shortest Lightpaths*	*Shortest Routes*
$\lvert \cup_{m\in\mathcal{M}}\mathcal{P}_m \rvert$	3888	2622	536
60	*	*	2.2169
300	*	3.1367	2.2023
1800	3.0343	2.8949	2.2023
3600	3.0343	2.2659	2.2023
9000	2.2659	2.1739	2.2023
10800	1.9073	2.1739	2.2023
36000	1.8659	2.0747	2.2023

For example, our framework considers the MILP solvable for a computation time limit of 3600 s if the approximate threshold $\lvert \cup_{m\in\mathcal{M}} \mathcal{P}_m \rvert \leq 536$. In this case, our framework will exclude feature (1) and (2) leaving a search space consisting of all the shortest routes between each SD pair. By increasing the time limit to 9000 s, our framework will consider the MILP solvable if $\lvert \cup_{m\in\mathcal{M}} \mathcal{P}_m \rvert \leq 2622$. In this case, our framework will only exclude feature (1) from the search space resulting in the level of congestion falling from 2.2023 to 2.1739.

In the existing literature [3], the search space is pruned so only the shortest routes between each SD pair remain, regardless of the time limit. This corresponds to the rightmost column in Table I. By expanding the search space beyond shortest routes in accordance with our framework, we have been able to improve on this level of congestion by about 15% for a time of 36 000 s.

VI. Conclusion

Our approach for solving the LTD problem within a constrained time is less complicated than heuristic methods and allows designs to exclude certain undesirable features such as cyclic routes. We have shown that our framework is an improvement to an existing SSD reduction technique. Future research will entail approximating the expected solution quality as a function of computation time and number of enumerated routes for a general network topology.

References

[1] D. Banerjee and B. Mukherjee, “Wavelength routed optical networks: Linear formulation, resource budgeting tradeoffs, and a reconfiguration study,” *IEEE/ACM Trans. Networking*, vol. 8, no. 5, pp. 598–606, Oct. 2000.

[2] R. M. Krishnaswamy and K. N. Sivarajan, “Design of logical topologies: A linear formulation for wavelength-routed optical networks with no wavelength changers,” *IEEE/ACM Trans. Networking*, vol. 9, pp. 186–198, Apr. 2001.

[3] J. Kuri, N. Puech, and M. Gagnaire, “Resolution of WDM optical network design problem using a decomposition approach and a size reduction method,” in *Proc. 2nd Eur. Conf. on Universal Multiservice Networks (ECUMN)*, Colmar, France, Apr. 8–10, 2002, pp. 187–194.

[4] E. Leonardi, M. Mellia, and M. Ajmone Marsan, “Algorithms for logical topology design in WDM all-optical networks,” *Optical Networks Mag.*, vol. 1, no. 1, pp. 35–46, Jan. 2000.

[5] P. Manohar, D. Manjunath, and R. K. Shevgaonkar, “Routing and wavelength assignment in optical networks from edge disjoint path algorithms,” *IEEE Commun. Lett.*, vol. 6, pp. 211–213, May 2002.

[6] N. Puech, J. Kuri, and M. Gagnaire *et al.*, “Models for the logical topology design problem,” in *NETWORKING 2002: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; and Mobile and Wireless Communications*, E. Gregori *et al.*, Eds.. Berlin, Germany, 2002, Lecture Notes in Computer Science no. 2345.