

ON SURVIVABLE DESIGN IN LIGHT TRAIL OPTICAL NETWORKS

Wensheng He, Jing Fang, Arun K. Somani

*Dependable Computing & Networking Laboratory
Department of Electrical & Computer Engineering
Iowa State University, Ames, Iowa 50011, USA*

whe, jfang, arun@iastate.edu

Keywords: Light trail, wavelength division multiplexing (WDM), survivability, optimization, integer linear programming (ILP)

Abstract The recently proposed *light trail* architecture offers a promising candidate for carrying IP centric traffic over optical networks. In this paper, we first give a brief introduction to the light trail architecture, then focus on the optimal design of survivable light trail optical network. Two protection schemes, namely *connection based protection* and *link based protection*, that can achieve 100% protection against single link failure are proposed and compared. The survivable light trail design problem using connection based protection model is formulated as an *integer linear programming (ILP)* optimization problem. The numerical results obtained from solving our ILP formulation are presented and show that the design achieves high wavelength utilization as well as 100% protection against single link failure.

1. Introduction

The explosive growth in IP traffic in the last decade has triggered a lot of research activities in devising new high-speed transmission and switching technologies. Wavelength division multiplexing (WDM) has emerged as a dominating transmission technology for the next generation IP backbone network with the capability of supporting a number of gigabit wavelength channels in a single fiber. In a typical WDM optical network, the connection between end users are supported by establishing an all-optical channel, namely *lightpath*, from the source to the destination. Signals are delivered transparently between end terminals without being terminated in the core network. This bit-rate and protocol transparency is a key feature for any backbone network. One challenging

problem for this wavelength switched optical network is the huge opto-electronic bandwidth mismatch. Once a lightpath is established, the entire wavelength is used exclusively by its source and destination node-pair (s-d pair), and no wavelength multiplexing between multiple nodes along the lightpath is allowed. Therefore, the wavelength capacity could be severely underutilized for IP bursts unless the wavelength is filled up by the efficiently aggregated IP traffic. A recently proposed concept named *light trail* [8] offers a strong candidate for supporting IP traffic over optical networks. Light trail architecture can be implemented using mature components that allows fast provisioning of network resource. Hence, in comparison to optical packet switching (OPS) [1-3], light trail requires neither the high speed electrical header processing for each packet, nor big optical buffering at a node. Moreover, the exclusion of fast switching at packet/burst level, combined with the flexible provisioning for diverse traffic granularity make the light trails superior to conventional circuit and burst switched architecture. Due to the huge bandwidth involved in WDM optical transporting networks, any link failure that leaves fiber unusable will have catastrophic results. Survivability is more predominant in light trail networks because one single link failure could cause failures of a set of light trails, each of which carries multiple connections. This paper is devoted to the study of survivable light trail design. The rest of paper is organized as follows. Section 2 is a brief introduction to light trail concept. Two protection schemes that can provide 100% protection in optical layer are proposed and compared in Section 3. A formal statement of light trail design problem is given in Section 4, followed by an integer linear programming (ILP) formulation for solving this optimization problem. Section 5 presents numerical results obtained from our experiments. Section 6 concludes the paper.

2. Light Trail Introduction

A light trail is a unidirectional *optical trail* between the start node and the end node. It is similar to a *lightpath* with one important difference that the intermediate nodes can also access this unidirectional trail. In light trails, the wavelength is shared in time and the medium access is arbitrated by control protocol among the nodes that try to transmit data simultaneously, that is, upstream nodes have higher priorities than lower stream nodes. The readers are referred to [8] for the details of light trails architectures. For the completeness of this paper, here we give a brief introduction to light trails.

2.1 Illustration Example

Consider a 4-node light trail shown in Figure 1, which starts from node 1, passes through node 2, node 3 and ends at node 4. Each of the nodes 1, 2 and 3 are allowed to send data to any of their respective downstream nodes without the need for optical switch reconfiguration. Every node receives the data from the upstream nodes, but only the corresponding destination node(s) will accept the data packets while other nodes will ignore them. An out-of-band control signal carrying information pertaining to the set up, tear down and dimensioning of light trails is dropped and processed at each node in the light trail. Since a light trail is unidirectional, a light trail with N_T nodes offers up to $\binom{N_T}{2}$ optical connections along the trail. This example shows that

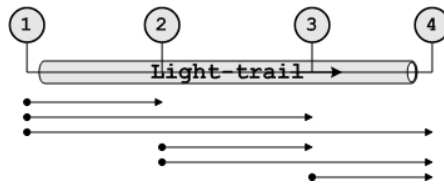


Figure 1. Illustrative example of traffic streams in a light trail.

light trails offer a method to group a set of nodes at the physical layer. Therefore, in contrast to optical burst switching (OBS) [4–7], there is no need to configure switches for each IP burst when using light trails to transport IP traffic. In fact, this leads to an excellent provisioning time and an order of magnitude better utilization than OBS under the similar situation [8].

2.2 How Does It Work

Figure 2 provides a typical node structure in light trail framework. In Figure 2, the multiple wavelengths from the input link are de-multiplexed and then sent to corresponding light trail switches. A portion of the signal power goes to the local receiver, the remaining signal power passes through an optical shutter which is typically an AOTF (Acousto-Optic Tunable Filter). Figure 3 gives a connection of four light trail nodes and the corresponding ON/OFF switch configurations. The direction of communication is from node 1 to node 4. The optical shutter is set to *OFF* state at the start and end nodes of the light trail, such that the signal is blocked from travelling further. For an intermediate node along the light trail, the optical shutter is set to *ON* state to allow the

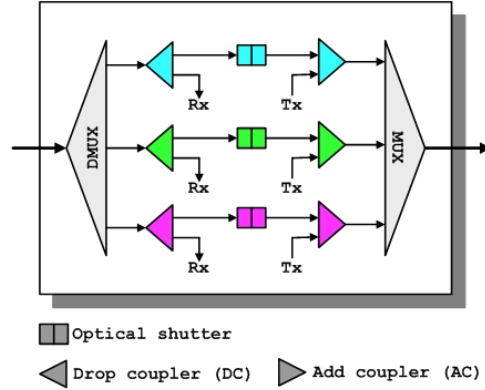


Figure 2. An example node structure in light trail framework.

signal to pass through the node. We thereby obtain a unidirectional light trail from the start node to the end node. No switch reconfiguration is required after the initial light trail setup. Due to the power loss within the light trail, which mainly comes from the power splitting at each node, the length of a light trail is limited and can be estimated in terms of hop-length. The expected length of a light trail is 5 hops [8].

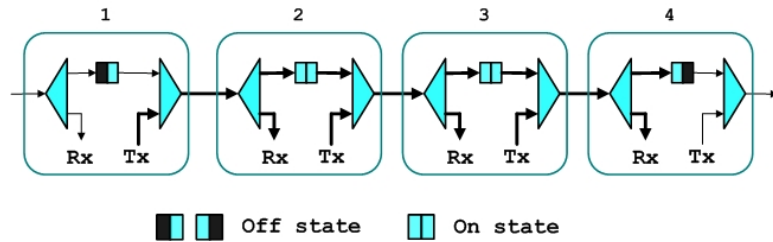


Figure 3. An example connection of four light trail nodes.

2.3 Why Light Trails

Current technologies that transport IP centric traffic in optical networks are often too expensive, due to their reliance on expensive optical and opto-electronic approach. Consumers generate diverse granularity traffic and service providers need technologies that are affordable and seamlessly upgradable. The light trail offers a technologically exclusive solution that enables a number of salient features and is practical. It exhibits a set of properties that distinguishes and differentiates from other

platforms. The following three characteristic properties of light trails make possible this differentiation:

- Light trails are built using mature components that are configured in such a way that allows extremely fast provisioning of network resources. This allows for dynamic control for the fluctuating bandwidth requirements.
- Light trails offer a method to group a set of nodes at the physical layer to create optical multicasting - a key feature for the success of many applications.
- The maturity of components leads to the implementation of light trails in a cost effective manner resulting in economically viable solutions for mass deployment.

3. Restoration Model in Light Trail Architecture

As stated above, survivability is a critical issue in the design of light trail optical network due to the fact that single link failure will disrupt all the light trails that use this link. Each of these light trails carries multiple connections. Therefore the failure effects would be catastrophic. For instance, if a failed link has W wavelength, it can carry up to W light trails. Each light trail contains up to $\binom{N_T^w}{2}$ s-d pairs, where N_T^w denotes the number of nodes in the w th light trail, $w = 1, 2, \dots, W$. Therefore, the worst case for this link failure is a service disruption of $\sum_{w=1}^W \binom{N_T^w}{2}$ connections. To provide 100% protection in WDM layer of light trail architecture, we need to provide backup at the time of establishing light trail. Recall that the key difference between the light trail and lightpath architecture is that the intermediate nodes in the light trail can also transmit or receive information. Thus the restoration model in light trail architecture is different from that in lightpath architecture [9–11]. Two protection schemes are proposed, namely *connection based protection* and *link based protection*. We assume that there is no more than one link failure at any time.

3.1 Connection Based Protection

For each connection request R_{s-d} , the resources are allocated to a primary connection in a light trail LT_1 and a backup connection in another light trail LT_2 . LT_1 and LT_2 are link-disjoint. The primary connection is the working connection when there is no link failure. If a link on LT_1 fails, the failure information is propagated through the control

channel. When the source node s of the request receives the failure information, it starts to transmit the data on LT_2 to the destination d through backup connection. We use an example to illustrate the scheme. In the example network in Fig 4, suppose there are two light trails, LT_1 : $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$, and LT_2 : $2 \rightarrow 6 \rightarrow 5 \rightarrow 4$. LT_1 and LT_2 are link-disjoint. There is a connection request from node 2 to node 4. The primary connection can be established on light trail LT_1 with backup connection on light trail LT_2 . Suppose link $2 \rightarrow 3$ on LT_1 fails, then light trail LT_1 cannot be used. When this failure information reaches source node 2, the source node will start to transmit data use backup connection on LT_2 .

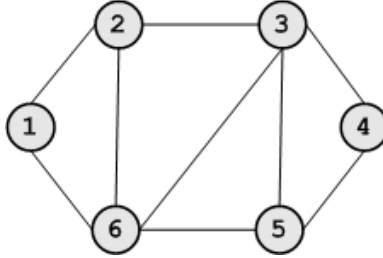


Figure 4. An example network.

3.2 Link Based Protection

For each link on a light trail, we provide a backup sub-light trail. When a link on a light trail fails, the light trail will be rerouted around the failed link and use backup sub-light trail. To do this, the failure information will be sent along the control channel. The information about what are the nodes on the backup sub light trail will be attached to the message. When the message arrives the source node of light bus, the source will send setup message along the nodes (including the nodes on the backup sub-light trail) to setup a new light trail, which basically is the remaining part of failed light trail plus the backup sub-light trail. Consider the above example again. Suppose for each of link on LT_1 , there is a backup sub-light trail. The backup sub-light trail for link $3 \rightarrow 4$ is $3 \rightarrow 5 \rightarrow 4$. If link $3 \rightarrow 4$ fails, the information about the failure and the sub-light trail of link $3 \rightarrow 4$ is sent along the control channel (there exists bidirectional control channel). When the message reaches source node 1, node 1 will send a fault management message, which is similar to light trail set up message, along the control channel to intermediate

nodes (node 2, 3, and 5) and end node 4. These nodes then configure the optical shutter and form a new light trail $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 4$.

3.3 Comparison of Connection Based and Link Based Protections

The connection based protection has following advantages over link based protection.

- Restoration time: In connection based protection, as soon as the failure information message reaches the source node of a connection that is using the light trail, the source node can immediately use backup connection in another light trail to continue the transmission. The maximum restoration time is transmission time of the control message. In contrast, in link based protection, after the failure information reaches the source node of the failed light trail, the source node will have to initiate a light trail setup process, i.e. setting up a light trail that includes the remaining part of the original light trail and the nodes on the backup sub-light trail of failed link. This takes much more time than restoration in connection based protection.
- The length of the light trail: As shown in the illustrative example, the restored light trail in link based protection is longer than original light trail. As we have discussed in Section 2.2, the length of the light trail is an important parameter that is related to signal-to-noise and bit-error-rate.

From the above discussion, we conclude that the connection based protection is more practical for light trail architecture. Therefore only connection based protection is considered in the rest of the paper.

4. Survivable Network Design

The major issue in the design of survivable light trail network is to identify a set of light trails to carry the given traffic and provide 100% protection against single link failure. The survivable light trail network design problem can be defined as follows. *Given graph $G(V, E)$, where $|V| = N$, and traffic matrix $T_{N \times N}$, to identify a minimum number of light trails to carry the given traffic in such way that for each connection request, there is a primary connection established in one light trail and resources are reserved in another light trail for backup connection. Two light trails are link-disjoint.* As stated earlier, due to the power losses on the lines, a long light trail may not be advisable. The length of a light trail is limited and can be estimated in terms of hop-length, denoted by

BL. According to the study in [8] the expected hop-length of a light trail is 5. Hence, an initial processing on the traffic matrix is needed. This can be achieved at higher layer in electrical domain. In this initial step, a single long hop is recursively divided into multiple hops in order to satisfy the hop-length constraint of light trail networks. The detail of this algorithm can be found in [12]. For the sake of completeness, this algorithm is included in **Appendix**. The next step is to develop an ILP formulations to optimize the capacity utilization in terms of number of light trails, with the given network topology and refined traffic matrix obtained from *traffic matrix preprocessing*. The objective is to find a minimum number of light trails that are required for the system.

4.1 ILP Formulation: Connection Based Protection

Given the network topology $G(V, E)$, and the traffic matrix obtained from *traffic matrix preprocessing*, we first list all possible paths with the hop-length limit constraint for each s-d node pair. This can be accomplished by applying *breath first search* for each node. These eligible paths form a set of all possible light trails. Among all these possible choices, we then chose an optimal set of paths to form the light trail network, such that the total number of light trails are minimized and the demand constraint and protection constraint are met. This problem is formulated as an ILP optimization problem. We also assume that each request cannot be divided into different parts and transferred separately.

4.2 Notation

The network topology is represented as a directed graph $G(N, L)$ with N nodes and L links with W wavelengths on each link. The following notations are used.

- $n = 1, 2, \dots, N$: Number assigned to each node in the network.
- $p, p_1, p_2 = 1, 2, \dots, P$: Number assigned to a path in the network.
- $i, j, k = 1, 2, \dots, N(N - 1)$: Number assigned to a node pair. The source and destination nodes of a connection request forms a node pair.

The following notations are used for path related information.

- δ_p^i : Path indicator which takes a value of one if primary connection for request i is established on light trail p ; zero otherwise (binary variable).

- ν_p^i : Path indicator which takes a value of one if backup connection for connection request i is established on light trail p ; zero otherwise (binary variable).
- ψ_p^i : Node pair indicator, which takes a value of one if node pair i is on path p ; zero otherwise (data).
- h_p : Takes a value of one if path p is used by some primary connection or (and) backup connection; zero otherwise (binary variable).
- d^i : Demanded capacity of connection request i (data).
- I_{p_1, p_2} : Takes a value of one if p_1 and p_2 are link-disjoint; zero otherwise (binary data).

4.3 ILP Formulation

4.3.1 Objective. Minimize number of light trails:

$$\min \sum_{p=1}^P h_p \quad (1)$$

4.3.2 Constraints.

- 1 On demand constraint for each node pair: For each request, there is one primary connection in one light trail, and a backup connection in another light trail.

$$\sum_{p=1}^P \delta_p^i \psi_p^i = 1 \quad 1 \leq i \leq N(N-1) \quad (2)$$

$$\sum_{p=1}^P \nu_p^i \psi_p^i = 1 \quad 1 \leq i \leq N(N-1) \quad (3)$$

- 2 On topology diversity of primary and backup connections: The primary and backup connection for a request are established in two link-disjoint light trails.

$$(\delta_p^i \psi_p^i + \nu_p^i \psi_p^i)(1 - I_{p_1, p_2}) \leq 1 \quad 1 \leq i \leq N(N-1), 1 \leq p_1, p_2 \leq P \quad (4)$$

- 3 On link capacity constraints: The total demand of all the connections on one light trail cannot exceed one wavelength capacity.

$$\sum_{i=1}^{N(N-1)} \delta_p^i \psi_p^i d_i + \sum_{i=1}^{N(N-1)} \nu_p^i \psi_p^i d_i \leq C \quad 1 \leq p \leq P \quad (5)$$

- 4 On light trail identification constraints: If one or more of the primary or backup connections use a path, then this path is a light trail.

$$2h_p \leq \sum_{i=1}^{N(N-1)} \delta_p^i \psi_p^i + \sum_{i=1}^{N(N-1)} \nu_p^i \psi_p^i \quad 1 \leq p \leq P \quad (6)$$

$$2N(N-1)h_p \geq \sum_{i=1}^{N(N-1)} \delta_p^i \psi_p^i + \sum_{i=1}^{N(N-1)} \nu_p^i \psi_p^i \quad 1 \leq p \leq P \quad (7)$$

5. Numerical Results

In this section, we present numerical results obtained by solving the above ILP formulation. We use CPLEX Linear Optimizer 7.0 [13] to solve the ILP formulation proposed in Section 4.3. First, we use a simple example to illustrate the solutions obtained from solving our ILP formulation. We then present results on the example 6-node network shown in Figure 4, and a 10-node network shown in Figure 5. To simplify the problem, we assume each physical link is bidirectional with the same length.

5.1 A Simple Illustrative Example

We present a simple illustrative example that helps to understand the solution obtained from solving the ILP formulation, which optimally identify the light trails covering all the working connections and backup connections. Consider the example network in Figure 4 again. Table 1 is an traffic matrix for this illustration example. The integer numbers indicates the request capacity in unit of OC-1 (51.84 Mbps), while the entire wavelength capacity is OC-48. Here we only take fractional wavelength capacity into consideration for the study of grooming. The study in [8] shows the average length of light trails is 5 hops. By taking the size of the example network into consideration, we choose hop length limit $BL = 3$. Table 2 gives the resulting light trails that covers all the connection requests and their corresponding backup connections. The notation (s, d) and $(s, d)_b$ in column 4 denote the accommodated primary and backup connection, respectively. As described in Section 3, to provide 100% protection for single link failure, for each request we allocate resource for a working connection in one light trail and reserve resource for the backup connection in another light trail. These two light trails must be disjoint. In Table 2, the working connection for request (1, 2) uses light trail $1 \rightarrow 2 \rightarrow 3$, while backup connection for this request is accommodated in light trail $1 \rightarrow 6 \rightarrow 2$. These two light trails are link-disjoint. Similarly, for request (1, 3), two link disjoint light trails

Table 1. Requests matrix for a 6-node network.

	1	2	3	4	5	6
1	0	11	6	0	14	8
2	0	0	0	0	5	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	19	0

Table 2. Resulting light trails for example request matrix I.

No.	L_B	$ L_B $	Accommodated $s - d$ pairs	load
1	{1, 2, 3}	2	(1, 2), (1, 3) _b	17
2	{1, 2, 6, 5}	3	(2, 5), (6, 5), (1, 5) _b , (1, 6) _b	46
3	{1, 6, 2}	2	(1, 6), (1, 2) _b	19
4	{1, 6, 3, 5}	3	(1, 3), (1, 5), (6, 5) _b	39
5	{2, 3, 5}	2	(2, 5) _b	5

$1 \rightarrow 6 \rightarrow 3 \rightarrow 5$ and $1 \rightarrow 2 \rightarrow 3$ are used. 5 light trails and total 12 wavelength-links are used for this request matrix.

5.2 Results

We demonstrate result on the example network given in Figure 4 and 5. Table 3 provides a randomly generated traffic matrix for 6 node network. Assume the hop-length limit $BL = 3$, from the topology we can observe that all s-d pairs have paths within this hop-length limit, hence, the *traffic matrix preprocessing* will not make any change of the given traffic matrix. Since we perform experiments mainly on small fractional wavelength requests, the number of wavelengths on each link is not a critical constraint. For this example, $W = 4$ is sufficient, although we do not put constraint on number of wavelengths. Table 4 presents the results from solving the ILP formulation with hop-length limit $BL = 3$. The solution is obtained by running CPLEX for 10 minutes on Pentium III 400MHZ processor with 256MB RAM.

Table 4 shows the 21 light trails that are needed to carry the primary and backup connections. The traffic assignment obtained from solving ILP formulation is also listed. For each light trail, the summation of all the traffic requests on it calculated as shown in the right most column in Table 4. It can be seen that most of the light trails are fully or almost fully occupied, hence, the resource utilization is quite high.

Table 3. Requests matrix for a 6-node network.

	1	2	3	4	5	6
1	0	7	6	9	19	17
2	27	0	3	6	28	2
3	14	3	0	19	31	9
4	26	5	29	0	5	23
5	27	20	20	17	0	14
6	9	30	1	1	1	0

Table 4. Resulting light trails.

No.	L_B	$ L_B $	Accommodated $s - d$ pairs	Load
1	{1, 2, 3, 4}	3	(1, 2), (1, 4), (1, 3) _b , (2, 4) _b , (3, 4) _b	47
2	{1, 6, 3, 2}	3	(1, 3), (1, 2) _b , (6, 2) _b	43
3	{1, 6, 5, 4}	3	(1, 4) _b , (1, 5), (1, 6), (6, 4), (6, 5)	47
4	{2, 3, 6, 1}	3	(2, 1), (2, 6), (3, 1), (2, 3) _b	46
5	{1, 2, 6, 3}	3	(2, 3), (1, 6) _b	20
6	{2, 6, 5, 4}	3	(2, 4), (2, 5) _b , (2, 6) _b	36
7	{1, 2, 3, 5}	3	(2, 5), (1, 5) _b	47
8	{3, 2, 6, 5}	3	(3, 2), (3, 5), (3, 6)	43
9	{4, 3, 2, 1}	3	(3, 1) _b , (4, 1), (4, 2) _b	45
10	{4, 5, 6, 2}	3	(4, 2), (4, 6), (5, 2) _b	48
11	{4, 3, 5, 6}	3	(4, 3), (4, 5), (5, 6) _b	48
12	{5, 6, 2, 1}	3	(5, 1), (6, 1)	36
13	{5, 3, 2, 1}	3	(5, 2), (2, 1) _b	47
14	{5, 6, 3, 4}	3	(5, 3), (6, 3), (5, 4) _b , (6, 4) _b	39
15	{3, 5, 4}	2	(3, 4), (5, 4)	36
16	{4, 5, 3, 6}	3	(5, 6), (4, 5) _b , (4, 3) _b	48
17	{4, 3, 6, 2}	3	(6, 2), (3, 2) _b	33
18	{6, 2, 3, 5}	3	(3, 5) _b , (6, 3) _b , (6, 5) _b	33
19	{4, 3, 6, 1}	3	(3, 6) _b , (6, 1) _b , (4, 6) _b	41
20	{4, 5, 6, 1}	3	(4, 1) _b	26
21	{5, 3, 6, 1}	3	(5, 3) _b , (5, 1) _b	47

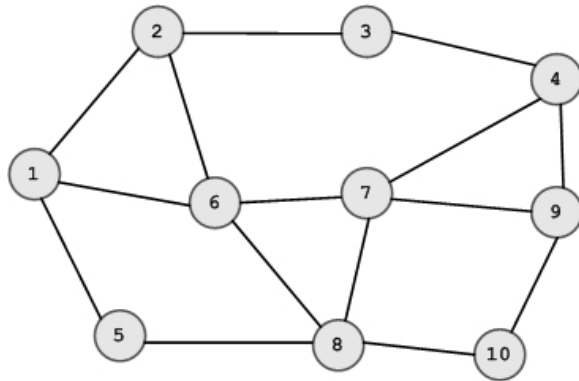


Figure 5. A 10-node 15-link network.

Table 5. Traffic matrix for a 10-node network.

	1	2	3	4	5	6	7	8	9	10
1	0	10	0	20	8	0	8	0	0	20
2	0	0	10	10	0	10	2	22	0	0
3	6	0	0	6	0	4	0	8	20	0
4	8	0	11	0	0	12	11	0	0	8
5	11	0	0	12	0	11	0	4	0	18
6	16	0	14	0	8	0	11	16	0	0
7	0	14	0	18	2	0	0	4	11	0
8	12	0	10	0	4	8	0	0	0	9
9	4	19	0	4	12	0	0	16	0	0
10	11	0	20	0	0	10	0	22	8	0

Table 5 is a randomly generated traffic matrix for a 10-node network. Assume the hop-length limit $BL = 4$, Table 6 shows the resulting light trails. The solution is obtained by running cplex for 5 hours on a Pentium III 400MHZ processor with 256MB RAM.

Compare to Table 4, some of the light trails in Table 6 are not very efficiently occupied. This is due to the fact that the average number of requests per node-pair ($50/(10 \times 9) \approx 0.55$) in this traffic matrix is smaller than the one in the traffic matrix for the 6-node network ($30/(6 \times 5) = 1$). Since the resulting light trail networks shown in Table 6 still have spare capacities, even if the number of requests increases, some of new requests could still be accommodated using the existing light trails. Solving larger network with more requests by ILP is time consuming, we are working on heuristic algorithms to solve this problem.

6. Conclusions

The concept of light trails has been proposed as a novel architecture designed for carrying finer granularity bursty IP traffic. The exclusion of fast switching at packet/burst level, combined with the flexible dynamic sub-wavelength provisioning make light trail architecture a strong candidate for transporting IP traffic over optical networks.

Due to the nature of light trails, survivability is a more predominant issue in the design of light trail optical networks than it is in traditional wavelength routed optical networks using lightpaths. We propose two protection schemes to provide 100% protection against single link failure, namely *connection based protection* and *link based protection*. Connection based protection scheme has advantages over link based protection scheme, and is more practical for light trail architecture where the hop-length is limited due to power loss. Hence, we adopted connection based protection model and formulated the survivable design problem with the objective to minimize the number of required light trails as an ILP optimization problem. The numerical results obtained from solving our ILP formulation are presented and show that the resulting light trail network achieves good wavelength utilization as well as 100% protection against single link failure.

The ILP formulation produces the optimal solution to the survivable light trail network design problems with static traffic demands. However, the computation time for solving ILP formulation is quite large, and it becomes unmanageable as the size of the network or the number of requests increases. We are currently working on the design of efficient heuristic approaches for solving survivable design problems in light trail networks.

Table 6. Resulting light trails.

No.	L_B	$ L_B $	Accommodated $s - d$ pairs	Load
1	{1, 2, 6, 8, 5}	4	$(1, 5)_b, (6, 5)_b, (6, 8)_b, (8, 5)_b$	36
2	{1, 5, 8, 7, 4}	4	$(5, 4), (1, 7)_b$	20
3	{1, 6, 7, 4, 3}	4	$(6, 3)_b, (1, 4)_b$	34
4	{1, 6, 7, 9, 10}	4	$(1, 7), (1, 10)$	28
5	{2, 3, 4, 7, 6}	4	$(2, 4), (2, 3)_b, (2, 7)_b, (3, 4)_b, (2, 6)_b$	38
6	{2, 3, 4, 7, 8}	4	$(3, 8)_b, (2, 8)_b$	30
7	{3, 4, 9, 7}	3	$(4, 7)_b, (3, 9)_b$	31
8	{2, 6, 7, 4, 3}	4	$(2, 3), (2, 6)$	20
9	{3, 2, 6, 7, 4}	4	$(3, 4), (2, 4)_b, (6, 7)_b, (7, 4)_b$	45
10	{3, 2, 6, 7, 9}	4	$(3, 6)_b, (3, 9), (2, 7)$	26
11	{3, 4, 7, 6, 1}	4	$(3, 6), (4, 1), (4, 6), (3, 1)_b$	38
12	{4, 3, 2, 6, 8}	4	$(2, 8), (3, 8), (4, 6)_b$	42
13	{4, 7, 6, 2, 3}	4	$(6, 3), (4, 3)_b, (7, 2)_b$	39
14	{4, 7, 6, 8, 10}	4	$(4, 10)_b, (8, 10), (7, 8), (4, 7)$	32
15	{4, 9, 10}	2	$(4, 10)$	8
16	{5, 1, 2, 3, 4}	4	$(5, 4)_b, (1, 4)$	32
17	{5, 1, 6, 2}	3	$(5, 1), (5, 6), (1, 2)_b$	32
18	{5, 1, 6, 8, 10}	4	$(5, 8)_b, (5, 10)_b$	22
19	{5, 8, 7, 6, 1}	4	$(5, 6)_b, (5, 1)_b, (8, 6), (8, 1)$	42
20	{5, 8, 7, 9, 10}	4	$(5, 8), (5, 10), (7, 9), (8, 10)_b$	42
21	{6, 1, 5, 8, 10}	4	$(1, 10)_b, (6, 8)$	36
22	{6, 8, 7, 9, 4}	4	$(7, 4), (6, 7)$	29
23	{7, 8, 5, 1, 2}	4	$(7, 8)_b, (7, 2)$	18
24	{7, 9, 10, 8}	3	$(10, 8)_b$	22
25	{8, 6, 2, 1, 5}	4	$(8, 1)_b, (8, 6)_b, (8, 5)$	24
26	{7, 4, 9}	2	$(7, 9)_b$	11
27	{8, 10, 9, 4, 3}	4	$(10, 3)_b, (9, 4)_b, (8, 3), (4, 3)$	45
28	{9, 4, 3, 2, 1}	4	$(9, 1)_b, (9, 2)_b, (4, 1)_b, (3, 1)$	37
29	{9, 4, 7, 8, 5}	4	$(9, 8), (9, 5)_b, (7, 5)_b$	30
30	{9, 7, 4}	2	$(9, 4)$	4
31	{9, 7, 6, 1, 2}	4	$(9, 1), (9, 2), (1, 2)$	33
32	{9, 7, 6, 1, 5}	4	$(1, 5), (9, 5), (7, 5), (6, 5)$	22
33	{10, 8, 6, 2, 3}	4	$(8, 3)_b, (10, 3), (10, 6)_b$	40
34	{10, 8, 6, 2, 1}	4	$(10, 1), (6, 1)_b$	27
35	{10, 8, 7, 4, 9}	4	$(10, 9)_b$	8
36	{10, 9, 7, 6, 1}	4	$(10, 6), (10, 1)_b, (6, 1)$	37
37	{10, 9, 7, 6, 8}	4	$(9, 8)_b, (10, 8), (10, 9)$	46

Acknowledgments

This research is in part supported by NSF grant ANI 9973102, ANI 0007746, and ANI 0323374.

References

- [1] P. Gambini *et al.*, “Transparent optical packet switching: network architecture and demonstrators in the KEOPS project,” *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 7, pp. 1245–1259, Sept 1998.
- [2] Y. Yamada *et al.*, “Optical output buffered ATM switch prototype based on FRONTIERNET architecture,” *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 7, pp. 1298–1307, Sept 1998.
- [3] M. Mahony, D. Simeonidou, D. Hunter, and A. Tzanakaki, “The application of optical packet switching in future communication networks,” *IEEE Communication Magazine*, pp. 128–135, March 2001.
- [4] M. Yoo and C. Qiao, “Optical burst switching (OBS) - a new paradigm for an optical internet,” *J. High Speed Networks (JHSN)*, vol. 8, no. 1, pp. 69–84, 1999.
- [5] J. Turner, “Terabit burst switching,” *J. High Speed Networks (JHSN)*, vol. 8, no. 1, 1999.
- [6] A. Ge, F. Callegati, and L. Tamil, “On optical burst switching and self-similar traffic,” *IEEE Communication Letters*, vol. 4, no. 3, pp. 98–100, March 2000.
- [7] S. Verma, H. Chaskar, and R. Ravikanth, “Optical burst switching: A viable solution for terabit ip backbone,” *IEEE Network*, pp. 48–53, November/December 2000.
- [8] Chlamtac, I. and Gumaste, A. Light-trails: A solution to IP centric communication in the optical domain. Online publication: February 17, 2003. Springer-Verlag Berlin Heidelberg 2003.
- [9] T.H. Wu, *Fiber Network Service Survivability*, Norwood, MA: Artech House, 1992.
- [10] S. Ramamurthy and B. Mukherjee, “Survivable wdm mesh networks, part i: protection,” *IEEE INFOCOM*, vol. 2, pp. 744–751, March 1999.
- [11] B.T. Doshi, S. Dravida, P. Harshavardhana, O. Hauser, and Y. Wang “optical network design and restoration”, Bell Labs Technical Journal, pp. 58–83, January-March 1999.
- [12] Fang, J. and Somani, A. K. IP traffic grooming in light trail optical networks. *Submitted to IEEE INFOCOM 2004.*
- [13] <http://www.cplex.com>.

Appendix

In the preprocessing of the traffic matrix, a single long hop traffic is divided into multiple hops to satisfy the hop-length constraint. For a given a network physical topology $G(V, E)$, with N nodes and E links,

we apply Dijkstra's shortest path algorithm to find the shortest path between all s-d pairs. This forms a distance matrix $D_{N \times N} = \{d_{ij}\}$, where d_{ij} denotes the physical distance from node i to node j . The length of a light trail is a main constraint due to the loss both at nodes and over the links. Let BL be the maximum length of a light trail. For traffic between s-d pair (i, j) , where $d_{ij} > BL$, it is not possible to accommodate this traffic on a direct light trail. Thus this traffic will need to go through multiple hops. Here one light trail is counted as one "hop". This necessitates the initial step in our approach, namely *traffic matrix preprocessing*. Let $T_{N \times N} = \{t_{ij}\}$ denote the estimated traffic matrix. Traffic matrix pre-processing will return a modified traffic matrix that satisfies: $T_{N \times N} = \{t_{ij} : d_{ij} \leq BL, \forall t_{ij} > 0\}$. Figure 6 provides the pseudo code for traffic matrix preprocessing algorithm. In

<p>INPUT: Graph $G = (V, E)$ and a traffic matrix $T_{N \times N}$. OUTPUT: Rearranged traffic matrix $T_{N \times N}$ and the distance matrix $D_{N \times N}$. ALGORITHM: <i>Step 0:</i> Apply Dijkstra's shortest path algorithm, calculate distance matrix $D_{N \times N}$. <i>While</i> (find $(i, j) : t_{ij} > 0, d_{ij} > BL$) {</p> <ol style="list-style-type: none"> 1 Pick an intermediate node k: $k = \arg \min_{v \in V} \{d_{vj} d_{iv} \leq BL\}$; 2 Update traffic matrix $T_{N \times N}$: <ol style="list-style-type: none"> (a) $t_{ik} \leftarrow t_{ik} + t_{ij}$; (b) $t_{kj} \leftarrow t_{kj} + t_{ij}$; (c) $t_{ij} \leftarrow 0$. <p>}</p>

Figure 6. L-bus establishment step 1: Traffic matrix preprocessing

this step, the traffic on s-d pair (i, j) with $d_{ij} > BL$, will be reallocated on multiple hops. The goal is to find a node k such that path from node i to node k forms the first hop which is less than BL in distance. A next intermediate node k is found recursively for the source node. Among all possible intermediate nodes, k is chosen to be as close to the destination node as possible, as shown in step 1 in Figure 6. This is done in order to reduce the number of hops that the original traffic has to take. After the preprocessing of the traffic matrix, each non-zero element in the modified traffic matrix would have corresponding distance less than BL , which is the maximum length allowed for a light trail.