Heuristic and optimal techniques for light-trail assignment in optical ring WDM networks

Ashwin Gumaste a, Paparao Palacharla b

a Indian Institute of Technology, Bombay, Bombay 400076, India
b Fujitsu Laboratories of America, 2801 Telecom Pkwy, Richardson, TX 75082, USA

Abstract

In this paper we address the problem of constrained optimization (ILP formulation) and propose a set of heuristic algorithms for assigning light-trails [1–4,7,10] to WDM ring networks to facilitate IP centric dynamic communication at the optical layer. A light-trail is a generalization of a lightpath such that multiple nodes can take part in communication along the path without the need for optical switching. A light-trail represents an opportunistic medium in which multiple spatially distributed sub-lambda flows can be groomed despite without the need for optical switching. A light-trail is analogous to an optical bus such that multiple connections between source–destination pairs can be provisioned under the constraint that no two connections have overlapping time-intervals. This enables traffic grooming at the optical layer. In this paper we first describe a constrained optimization procedure for assigning light-trails in WDM ring networks. We then show five heuristic algorithms that solve the light-trail assignment problem in polynomial time. The heuristic algorithms are based on dynamic (unknown traffic) and static (known traffic) approaches. A simulation study compares the performance of ILP and heuristic algorithms.

Keywords: Light-trail; WDM ring networks; Traffic grooming; Dynamic provisioning

1. Introduction

Fiber optic communication provides a befitting transport mechanism for the near exponential surge of data traffic. The present mode of fiber optic networking is based on lightpaths or optical circuits that are end-to-end all-optical paths residing on a single wavelength. However as we move from circuit switched to IP centric communication [9,8] due to the emergence of IP as a universally accepted protocol, the full granularity – that of an entire wavelength provided by a lightpath is not an efficient means for bandwidth provisioning. The result is that networks using lightpath communication and wavelength routing are over-provisioned leading to expensive network element deployment and not being able to provide the dynamic guarantees of bandwidth provisioning as required by IP centric communication. Light-trails [1–5,10] on the other hand offer a low cost pragmatic alternative by providing dynamic provisioning as compared to rigid lightpath communication. A light-trail is a generalization of a lightpath such that multiple nodes along the trail can take part in communication without the need for optical switching [5,10]. Thus, a light-trail can cater to sub-lambda flows in an efficient manner. The principle of a light-trail is similar to that of an optical bus whereby multiple nodes along the bus can communicate to their downstream nodes (broadcast medium), under the constraint that no two nodes transmit at the same time. To enable non-time overlapping transmissions we make use of an out-of-band control channel that arbitrates communication amongst nodes in a light-trail. A light-trail by itself is semi-permanent, defined by its two extreme nodes – a convener node and an end node (Fig. 1). The light-trail can be further understood as an optical bus exemplified by a wavelength, switched between the convener and the end
nodes and shown in Fig. 1. Though the setting up procedure of light-trails does involve optical switching, the procedure to set up ‘connections’ between nodes within the light-trail does not involve any kind of optical switch configuration. Connections in a light-trail can be of dynamic nature and the out-of-band control protocol does the task of connection management. This means that the out-of-band protocol arbitrates communication and resolves conflicts in the light-trail. The protocol shown in [1] also has the added responsibility of light-trail management in addition to the dynamic connection management, whereby it sets up the light-trails, tears them down as well as dimensions (grows/contracts) the set of light-trails as per requirement. The hardware and protocol for light-trails are shown in [1–5,10] for ring and mesh nodes.

### 1.1. Need for a heuristic algorithm for light-trail assignment in ring networks

A Need for optimization: light-trails present a mechanism that enables dynamic provisioning, optical grooming and optical multicasting. Spatially diverse sub-wavelength flows from multiple nodes are groomed in the wavelength bus (light-trail) by arbitrating the bandwidth in the light-trail to accommodate each flow. A light-trail represents an opportunistic medium for multiple nodes to share wavelength bandwidth. While the out-of-band control channel is fundamental in bandwidth arbitration in the light-trail, it also manifests itself in another important way: the control channel also sets up, tears down and dimensions light-trails in the network. This double abstraction of the control plane in light-trail networks – at the connection (intra-light-trail) level, and at the network-wide light-trail level is fundamental to enable the network to support traffic growth. While setting up and tearing down connections is dynamic (due to absence of optical switching), the setting up and tearing down of light-trails is significantly time-consuming. It was shown in [15] that the time required to set up connections is of the order of 10 ms, while the time to set up light-trails was three orders of magnitude higher at 2.4 ms. To preserve the dynamic provisioning property of light-trails, it is desired to minimize the probability of setting up a light-trail for a new traffic request. Alternatively it is desired to create a topology of light-trails mapped to nodes and wavelengths across the network, such that traffic is readily routed across these light-trails. The problem of grooming spatial sub-wavelength flows into light-trails while considering the temporal aspects of traffic leads to a constrained optimization problem. The fundamental question is how to set up the static light-trails in the best possible manner? In order to do this, we have to minimize the resources (wavelengths) used, minimize the provisioning time and maximize the possibility that a new bandwidth request will find an available light-trail, thus reducing the probability of needing to create an extra light-trail. A factor that has to be taken into account is that light-trail communication is constrained by the bus property, i.e., by the fact that upstream nodes have a higher priority for establishing connections as compared to downstream nodes.

The bus property leads to a situation where, if a node is in the process of transmitting data over a connection, and an upstream node desires to use the light-trail, the node has to interrupt and halt its transmission allowing the upstream node to transmit. As a result, the queuing delay on nodes in a light-trail will depend on their position within the trail. As most broadband applications have stringent latency requirements, the queuing delay is a function of the node’s position with respect to the convener of the light-trails. The goal of the optimization process is to create a system where we assign optimal light-trails across a network graph such that the traffic latency incurred in provisioning connections or new light-trails is within the bounds required by the traffic demands.

In this paper, we will first develop a linear program for optimally assigning light-trails taking into account the temporal aspects of traffic. The constrained optimization will attempt to maximize the wavelength utilization in each light-trail. Subsequently we will build heuristic algorithms for assigning light-trails thereby reducing the apparent complexity of the linear program. Our optimization is significantly different from earlier light-trail optimization approaches shown in [2] and [6]. While [2] is a simple optimization of allocating light-trails to a given traffic matrix, [6] shows how to allocate light-trails based on hop distances. However, this approach is similar to the routing and wavelength assignment (RWA) problem for lightpaths and does not consider the temporal (stochastic) aspect or the sub-wavelength characteristics of light-trail flows. Our approach is a mid-way between the approaches of [2] and [6] whereby we consider factors like light-trail fairness and optimum size both part of our optimization. We first normalize each source–destination flow by taking the average rate of the flow, the latency of the service represented by the flow and the ratio of the flow to the light-trail (wavelength) bandwidth. Then by considering the entire set of possible light-trails, we allocate flows from source–destination pairs by creating an optimum sub-set of chosen light-trails. The resultant optimum set of chosen light-trails is the output of our constrained optimization problem. We constrain our optimum set of light-trails to have maximum utilization in each light-trail. This ensures that the total number of light-trails selected (and hence wavelengths) is minimized. The light-trail assignment problem is analogous to the bin packing problem [2,6] with a slight modification, of aligning the flows within each light-trail (bin) under the constraint that a light-trail is a unidirectional bus and has limited
capacity (packing). Literature from other authors on light-trails is available in [11–13] and the references therein.

1.2. Differentiation from OEO grooming problem, RPR and HORNET architectures

It is important to note that a light-trail is an all-optical wavelength bus. This means that there is no opto-electronic conversion of the signal at intermediate nodes. Signals of sub-wavelength granularity are queued electronically at node transmitters. These transmitters have specific burst-mode operation and called burstponders [17]. An implementation of burstponder is shown in [17]. When a node transmits data it does so in the optical domain. The data travels to the end node of a light-trail all-optically. Hence the problem of grooming in light-trails is different from the ones in lightpaths [16] or SONET/SDH (electronic switching) [14]. Similarly, in IEEE 802.17 Resilient Packet Rings or RPR, grooming is done entirely in the electronic domain (both pass-through signal and added signals are combined in the electronic domain), while for light-trails the pass-through signal is in the optical domain, and only when there is no pass-through signal (statistical multiplexing) does the node add its own signal. A popular optical packet network HORNET [19] deploys optical grooming based on CSMA technique. HORNET makes use of a fast acousto-optic tunable filter (AOTF) for collision avoidance. This technology is limited due to extinction ratio [20] of the filter and hence HORNET is difficult to implement in service provider networks. In light-trails we avoid using fast tuning by using arbitration through an out-of-band control channel. This distinction between control and data plane based on both wavelength diversity and off-set time (control packets for establishing connections are sent prior to the connection [1,2]) differentiates and simplifies the light-trail technology from HORNET.

2. Linear program formulation

We now develop a constrained optimization for assigning light-trails for a given traffic matrix.

We define $C$ as the line rate of any light-trail (typically 1 or 10 Gb/s). For an $N$ node networks we have the following definitions.

2.1. Traffic

Let $T$ represent the traffic flow matrix, such that represents the time-averaged flow from node $N_i$ to node $N_j$ [2]. If $C$ is the light-trail bandwidth and if $t_{ij}$ is the low from node $N_i$ to node $N_j$ in bits per second, then $t_{ij}/C$ represents the time-averaged flow from node $N_i$ to node $N_j$. Hence $T_{ij} = t_{ij}/C$ and $0 < = T_{ij} < = 1$. We note that if, $T_{ij} < C$ signifies the flow is sub-lambda and can be accommodated in a light-trail, $T_{ij} > C$ signifies $T_{ij}$ is a flow greater than the bandwidth offered by a wavelength. This means we can break into two components, one of static lightpaths each having $C$ units of flow, while the other into sub-lambda (<$C$) units of flow that can be provisioned into light-trails. Further, let $K_{ij}$ be the number of lightpaths that can be created from $T_{ij}$ and is given by $K_{ij} = \left\lfloor \frac{T_{ij}}{C} \right\rfloor$ where $\lfloor \rfloor$ indicates division of $T_{ij}$ by $C$ neglecting the remainder. The sub-lambda flow signified by the remainder of the division is $T_{ij}^* = T_{ij} \mod C$ where $\mod$ function gives the remainder after division.

From the traffic matrix $T$, we create a modified matrix $T^*$ such that no element of $T^*$ has value greater than $C$, indicating that all the flows of $T^*$ are sub-lambda in granularity. To obtain $T^*$, we remove all $K_{ij}$ from the original matrix $T$ by performing the $\mod$ and $\lfloor \rfloor$ operations on every source–destination pair $(i,j)$.

Our objective is to groom geographically diverse sub-lambda flows together and isolate the flows that are not sub-lambda (need bandwidth greater than that provided by a single lambda). Flows that occupy full wavelength granularity can be best provisioned by establishing full duration lightpaths [6]. By considering the time-averaged value of the flows, we are able to neglect the effect of upstream priority in light-trails. By considering flows as time averaged quantities, and assigning to a light-trail such that the combined value of all the time-averaged flows in a light-trail is lesser than or equal to the total capacity of the light-trail we are able to ensure that the light-trail is fair to its nodes as long as the assigned time-averaged flows are not altered. Note that to meet service requirements we still require a scheduling algorithm that schedules these connections in the light-trail. One such algorithm is presented in [18].

2.2. Active and passive nodes

The time-averaged flows represent the average value of traffic from a source to a destination. A node on a given light-trail has access rights of the channel, proportional to its position in the trail. In other words, a node further downstream in the light-trail has lesser probability of sending data in the light-trail successfully, as it has to wait to ensure that no upstream node transmits data when it is doing so. In [4], we derived equations that bound the queuing delays as well as the maximum cumulative flow allowed in a light-trail. The summary of that result suggests that the sum of all the sub-lambda flows that can be accommodated in the light-trail is less than or equal to the capacity of the wavelength. Further considering that upstream nodes have priority over downstream nodes, this suggests that for a $n$ node light-trail, if the first $y$ nodes have cumulative sub-lambda flows totaling the capacity of the light-trail, then the remaining $n - y$ nodes will not be able to send data in this light-trail, or will have infinite queuing delay. Based on this observation, we define two types of nodes in a light-trail – active and passive node. Active nodes are sources of data and possible destinations, while passive nodes are purely destinations. Note that an active node in one light-trail may be a passive node in another light-trail due to wavelength
diversity. Hence active and passive aspect of a node is specific to a given light-trail.

2.3. The linear program formulation

We first define the following matrices for an $N$ node ring, with the assumption of 2-fibers supporting the ring in the form of two counter-propagating ringlets.

$P_{\text{max}} = 2N(N - 1)$, the maximum number of possible light-trails in the ring.

Construct $X$ a matrix of dimension $[2N(N - 1), 2^N - 1]$, where $X_{km}$ gives the total flow in the $k$th light-trail with $m$th combination of active nodes.

$L_{\text{max}} = 2^N - 1$, is the total number of combinations of active nodes in the $N$ node network.

Construct matrix $W$, such that $W_{ks}$ is binary and:

$W_{ks} = 1$ if light-trail $k$, contains $s$th source–destination pair,

$W_{ks} = 0$, otherwise.

Create $T^{**}$ as a one-dimensional $1 \times N^2$ array where by $T^{**}_s$ denotes the flow in $s$th source–destination pair. $T^{**}$ is created from $T$.

Then our objective function is

$$\min L = \sum_{k=1}^{P_{\text{max}}} \sum_{m=1}^{L_{\text{max}}} X_{km} Z_{km}$$ (1)

Explanation: We minimize the number of chosen light-trails from the set of all possible light-trails.

2.3.1. Random constraint

With the variables $Z_{km}$ taking the binary values as follows:

$Z_{km} = 1$ if light-trail $k$ with $m$th combination of active nodes chosen, (2)

$Z_{km} = 0$, if otherwise. (3)

2.3.2. Capacity constraint

$X_{km} Z_{km} \leq C$ (4)

Eq. (4) signifies that the total aggregate (groomed) flow in the light-trail is less than the capacity of the wavelength.

2.3.3. Traffic constraint

$$\sum_{k=1}^{P_{\text{max}}} X_{km} W_{ks} Z_{km} \equiv T^{**}_s$$ for $s = 1, 2, \ldots, N^2$ and $m = 1, 2, \ldots, L_{\text{max}}$ (5)

where $T^{**}_s$ represents the traffic flow between source–destination pair $s$.

Eq. (5) ensures that each flow in the matrix $T^{**}$ (and hence in $T$) is routed through a light-trail.

2.3.4. Explanation

The minimization objective function shown in (1) minimizes the number of light-trails by choosing the most optimum fit of light-trails to suffice the traffic matrix $T^{**}$. The variable $Z$ is the decision variable – whether or not to choose a particular light-trail $k$. In (4) we have the capacity constraint which does not allow any light-trail to have a cumulative flow greater than the capacity of a wavelength, $C$. Finally in (5), we show the fulfillment constraint such that, the sum of the flows for all the chosen light-trails with the combination of active nodes will guarantee to have the $s$th source–destination pair to suffice $T^{**}_s$.

2.4. Numerical example

In this section, we will consider an example of assigning linear light-trails using the LP formulation. We use Lingo$^\text{TM}$ to solve the LP proposed in the previous section. For our example we consider a 10-node ring network with an arbitrary number of wavelengths in each direction. The number of possible light-trails for the 10-node network is 180. The size of the matrix $X$, on the other hand for the 10-node network is of the value $180 \times (2^{10} - 1)$, or $180 \times 1023$.

This shows that the LP formulation is very complex for small size networks and hence this is one of the primary motivators for a heuristic approach to assigning light-trails. Given in Table 1 is a traffic matrix for 10 nodes. Here the traffic flows are normalized time-averaged and a flow of unity signifies full wavelength occupancy. Table 2 gives the result of the LP formulation. In this table, we vary the traffic matrix from low loads to higher loads, and

<table>
<thead>
<tr>
<th>Table 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic matrix ($T_{ij}$ is the flow from node $i$ to $j$)</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0.6405</td>
</tr>
<tr>
<td>0.6029</td>
</tr>
<tr>
<td>0.7889</td>
</tr>
<tr>
<td>0.7446</td>
</tr>
<tr>
<td>0.2071</td>
</tr>
<tr>
<td>0.0129</td>
</tr>
<tr>
<td>0.1901</td>
</tr>
<tr>
<td>0.4118</td>
</tr>
<tr>
<td>0.4508</td>
</tr>
</tbody>
</table>
Table 2
(a) Computation results for 10-node ring, and (b) Computation results for 12-node ring and Figure represents graphical plot of results

<table>
<thead>
<tr>
<th>Traffic (Utilization)</th>
<th>Light-trails Utilization (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>27  74</td>
</tr>
<tr>
<td>28</td>
<td>37  76</td>
</tr>
<tr>
<td>36</td>
<td>45  80</td>
</tr>
<tr>
<td>44</td>
<td>51  86</td>
</tr>
<tr>
<td>52</td>
<td>84  62</td>
</tr>
<tr>
<td>60</td>
<td>86  70</td>
</tr>
<tr>
<td>68</td>
<td>92  74</td>
</tr>
<tr>
<td>76</td>
<td>96  79</td>
</tr>
<tr>
<td>84</td>
<td>103 82</td>
</tr>
<tr>
<td>92</td>
<td>107 86</td>
</tr>
<tr>
<td>100</td>
<td>110 91</td>
</tr>
</tbody>
</table>

(b)

<table>
<thead>
<tr>
<th>Traffic (Utilization)</th>
<th>Light-trails Utilization (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>34  59</td>
</tr>
<tr>
<td>28</td>
<td>47  60</td>
</tr>
<tr>
<td>36</td>
<td>56  64</td>
</tr>
<tr>
<td>44</td>
<td>67  66</td>
</tr>
<tr>
<td>52</td>
<td>75  69</td>
</tr>
<tr>
<td>60</td>
<td>86  70</td>
</tr>
<tr>
<td>68</td>
<td>94  72</td>
</tr>
<tr>
<td>76</td>
<td>103 74</td>
</tr>
<tr>
<td>84</td>
<td>110 76</td>
</tr>
<tr>
<td>92</td>
<td>117 79</td>
</tr>
<tr>
<td>100</td>
<td>123 81</td>
</tr>
</tbody>
</table>

3. Heuristic algorithms for light-trail assignment

Heuristic algorithms proposed are of two kinds, one type in which the traffic matrix is known and a quick solution is to be determined (static), while a second type in which the traffic is not known and an adaptable solution (dynamic) is to be determined.

3.1. Static assignment

In this sub-section we propose three heuristic algorithms for static schemes.

3.2. Heavily loaded precedence scheme (HLPS)

We sort the traffic-flow, T, between all source–destination pairs in the ascending order of flows (in terms of their granularity) and call this sorted matrix as S. Define L as the set of light-trails we generate while f(LTij) is the total flow in light-trail from converter node i to end-node j, while t(x − y) as the flow from node x to y. Further {ab} is the set of nodes in the light-trail corresponding to ring arch a ∼ b.

While (S) ≠ 0
Do, SP = max(S)
LTij = SP, L ← L∪{SP}
s′d′ = {SP}, f(LTij) = v(SP)
S ← S − SP
While f(LTij) ≤ C or s′d′ ≠ φ
Find s′d′, such that s′d′ ∈ LTij and s′d′ ≠ SP and v(s′d′) > all v(s′d′) ∈ LTij
If v(s′d′) + v(SP) ≤ C
S ← S − v(s′d′),
LTij ← LTij − s′d′
End

The algorithm begins by finding out if there are any flows still to be assigned within S. It then selects the largest flow (by using the max function) and assigns a light-trail to this flow (SP). While the value of the flow in this new light-trail is lesser than the capacity C, and there are nodes that have yet to be searched within SP, the algorithm finds the next highest loaded source–destination pair (s′d′) within this new trail LTij. It keeps repeating until either the capacity of the light-trail is exhausted (=C) or there are no source–destination pairs to find within SP. In this algorithm as we allocate a flow to a light-trail between any given source–destination node, we deduct this flow from S.

3.3. Per node precedence assignment (PNPS)

In PNPS, we create one light-trail at every node that begins from the node, and then we map the flows beginning from that node to all the other nodes in the ring. If the total nodal flow from which the light-trail begins, destined for all other nodes cannot be met by a single light-trail, then we form two light-trails, and so on till the flow is routed successfully. The algorithm is shown below:

For each n ∈ N
Calculate Tn from T, such that:
Tn = Σj=1Af(n, j), j ≠ n for all n
Do create \( K_n \) light-trails of \( N - 1 \) hops (spans) originating at \( n \).

If \( T_n \leq C; K_n = 1 \)
Elseif \( T_n > C; K_n = \left\lceil \frac{C}{C_0} \right\rceil + 1 \)
End

For each node \( n \), part of the \( N \)-node ring, we calculate the total flow emanating from node \( n \) as \( T_n \). Further we set \( K_n \) light-trails at \( n \) where \( K_n \) depends on the value of \( T_n \). Larger the value of \( T_n \), higher the number of light-trails required. This scheme guarantees fairness but results in higher number of light-trails.

### 3.4. Longest pair heaviest load assignment (LPHL)

This scheme is similar to HLPS, except that the sorting algorithm gives priority to heavy flows, which are also the farthest apart from each other in terms of hop distances. The idea is to incorporate maximum number of (diverse) flows by considering longer light-trails first. The algorithm is as shown:

Create \( \{S\} \), such that \( S_{ij} = \max(T) \) and so on (sorting), and \( |i - j| > |i' - j'| \) for \( S_{ij'} = S_{ij} \)
While \( (S) \neq 0 \)
Do, \( SP = \max(S) \)
\( LT_{ij} = SP \), \( L \leftarrow L + \{SP\} \)
\( s'd' = \{SP\}, f(LT_{ij}) = v(SP) \)
\( S \leftarrow S - SP \)
While \( f(LT_{ij}) \leq C \) or \( s'd' \neq \phi \)
Find \( s'd' \), such that \( s'd' \in LT_{ij} \) and \( s'd' \neq SP \) and \( v(s'd') > \) all \( v(s'd') \in LT_{ij} \)
If \( v(s'd') + v(SP) \leq C \)
\( S \leftarrow S - v(s'd'), LT_{ij} \leftarrow LT_{ij} - s'd' \)
End

The algorithm is similar to HLPS except in the first step, when we create the sorted traffic matrix \( S \), based on not one, but two factors – heaviest flow and longest path in that order of precedence. Subsequently, the next maximum flow from \( S \) is selected and a light-trail is created to accommodate this flow. All flows that have sources and destinations within the newly created light-trail are then mapped into the light-trail under the constraint that the total flow of the light-trail is lesser than the maximum wavelength bandwidth \( C \). This algorithm performs slightly better than HLPS but also has longer running time.

Both HLPS and LHPL have complexity of the order \( O(n^3) \) (the second while loop in each case is \( O(N^2) \)) and the first while loop is \( O(N) \).

### 3.5. Dynamic assignment

In this sub-section we propose two dynamic assignment algorithms, for which the traffic matrix is not known before hand.

#### 3.6. The Cantor set growth method (CSGM)

In CSGM for a ring network of \( N \) nodes, we create two arbitrary light-trails \( L_1 \) and \( L_2 \) such that convener of \( L_1 \) is the end node of \( L_2 \) and vice versa, with the constraint that \( L_1 \) and \( L_2 \) have \( N - 1 \) hops. Then we have a system with 2 light-trails in opposite direction able to cover \( N^2 \) connections. Let \( CS \) be the Cantor set dividing value, and \( CS \) here is 2. Begin by placing the traffic in these two light-trails, and if the total traffic flow exceeds that supported by the two light-trails, then create Cantor segments of each light-trail to make new light-trails as shown in Fig. 2. The logic is to consider the existing light-trails as an ancestor of the next Cantor segments. The relation between the ancestor and descendant is the Cantor set dividing value \( CS \), which is 2 in this case. As shown in Fig. 2, we begin by having one light-trail of \( N - 1 \) spans (hops). As the traffic flow increases, we create 2 new light-trails each of \( (N - 1)/2 \) hops and continue to repeat the Cantor set procedure of creating fractals of the previous light-trails until the traffic flow is routed. The disadvantage of this scheme is that it results in re-routing of existing connections to facilitate good utilization. Without re-routing a situation can be reached where a combination of shorter light-trails have enough capacity to route a connection between nodes far apart but an existing longer light-trail is busy accommodating shorter connections.

Let \( X \) be the number of LT already created, and let \( Y \) be the hop length of the light-trail last created, with \( CS \) the Cantor set dividing ratio, and \( p \) the number of times the original light-trail (of \( N - 1 \) nodes) has been divided, then,

While \( T \neq 0 \)
Create \( CS \times X \) light-trails each of \( Y/CS \) hops
Route traffic:
For \( h = 1:p \)
For each of the \( h \times CS \) light-trails (with \( N/CS \times h \) hops)
Do:
If traffic \( T_{ij} \in \) of light-trail, and \( |i - j| \leq N/CS \times h \)
Assign \( T_{ij} \) to this light-trail
\( T \leftarrow T - T_{ij} \)
End
End

![Fig. 2. Cantor Set Growth Method.](image-url)
3.7. Decrement wrapping

Consider a ring of \( N \) nodes. Let \( N_A \) be an arbitrary node in the ring, then set up a clockwise light-trail beginning at \( N_A \) and of \( N - 1 \) hops (spans). The convener node for this light-trail is \( N_A \), and the end node is \( N_{A+N-1} = N_B \) (say). Now set up a light-trail in the counterclockwise fiber, beginning from \( N_B \) and of \( N - 1 \) hops, thus ending at \( N_A \). Now these two light-trails can provide complete \( N^2 \) connectivity under the constraint that the total capacity required by the \( N^2 \) connections is lesser than \( 2C \), where \( C \) is the capacity of each light-trail (wavelength). However, if the load is higher than \( 2C \) then we cannot suffice with just two light-trails and need to create more light-trails. We do so using decrement wrapping. Each time, the existing set of light-trails is insufficient for the traffic flow we build a new set of light-trails with the longest light-trail in the new set, decremented by one span (hop) from the longest light-trail in the previous iteration. The logic is to route shorter connections in the shorter light-trails and longer connections in the longer trail while still creating a large set of very small light-trails to suffice very short duration traffic as shown in Fig. 3. The position of decrementing the next light-trail is random.

4. Numerical simulations

4.1. Simulation model

We simulate and compare performances of the proposed five heuristic schemes and optimal scheme. For the simulation model, we consider 24-node 2-fiber ring network. Traffic flows values are random and we consider only normalized values (\( C = 1 \)). Three performance parameters were considered: the total number of light-trails required, the wavelength (bandwidth) utilization of each light-trail, and the residual capacity of the entire network (which indicates the ability to accommodate new light-trails).

4.2. Utilization comparison

Shown in Fig. 4, we compare the utilization of light-trails for all the schemes shown previously. On the \( X \)-axis, we show network load as a percentage of full load and we compare the light-trail utilization on the \( Y \)-axis. The LPHL scheme performs second best after the optimal scheme as expected, while the dynamic schemes (Cantor set and decrement wrapping) perform quite poorly as can be seen. Amongst the dynamic scheme, the Cantor set method is better suited because of the reduction of light-trail lengths by half in each iteration.

4.3. Number of light-trails required

In Fig. 5 we compare the number of light-trails required for network load. Of all the heuristic schemes, the LPHL performs best and is in fact within 15% of the optimal scheme. The HLPS and PNHS are also somewhat acceptable schemes within a mean of 25% and 32% of the optimal schemes. The dynamic schemes need a large number of light-trails and perform 35 ~ 50% worst than the optimal scheme for the same traffic considerations.
4.4. Residual capacity comparison

In Fig. 6 we compare residual capacity of the schemes as a function of load. Here we observe that the residual capacity of the dynamic schemes (Cantor set and decrement wrapping) is better than the residual capacities of the static schemes (LPHL, PNHS and HLPS). The lesser utilization of the dynamic schemes means that there is free capacity in the light-trails to accommodate any new ‘churns’ (increases) of traffic.

5. Conclusion

We have shown constrained optimization of light-trails as a solution for IP centric communication at the optical layer. A light-trail is an opportunistic medium whereby multiple sub-lambda flows are groomed together in the optical domain. The optimal way to do so is by creating an ILP formulation. The formulation is shown in this paper. The ILP is difficult to solve at higher node counts due to limitations posed by commercial LP solvers. Hence we propose heuristic algorithms for static and dynamic assignment of light-trails in optical ring networks, and compare the performance of these algorithms. Five such algorithms are proposed. We evaluate the algorithms based on criteria such as utilization, number of light-trails generated and residual capacity of the light-trails.

Acknowledgement

Authors wish to thank and acknowledge Prof. Krithi Ramamritham at IIT Bombay, Dr. Hideo Kuwahara and Dr. Susumu Kinoshita at Fujitsu Laboratories Ltd and Dr. Takao Naito at Fujitsu Laboratories of America USA for their encouragement and support.

References


Ashwin Gumaste received a Ph.D. from The University of Texas at Dallas and is currently with Indian Institute of Technology Bombay. He was previously with Fujitsu Laboratories (USA) Inc as a Member of Research Staff in the Photonics Networking Laboratory. Prior to that he worked in Fujitsu Network Communications R & D and prior to that with Cisco Systems under the Optical Networking Group. He has over forty pending U.S. and EU patents, and in 1991, Ashwin was awarded the National Talent Search Scholarship (NTSE) in India. His research interests include optical networking and access networks and broadband technology. He proposed an optical solution called Light-trails to solve the problem of dynamic provisioning, sub-wavelength grooming and low-cost implementation, and also proposed the Light-frame framework, a conceptual model for future packet mode optical communication. The light-trail concept is well cited and currently under industrial development. It has also attracted attention as a paper session in conferences. He has authored three books in broadband networks called DWDM Network Designs and Engineering Solutions, and First Mile Access Networks and Enabling Technologies (for Pearson Education/Cisco Press) and Broadband Services: User Needs, Business Models and Technologies for John Wiley.

Paparao Palacharla received the B.Tech degree in Electronics Communication Engineering from Indian Institute of Technology, India, in 1986 and the Ph.D. degree in Electrical Engineering from University of Iowa,
Iowa City, IA in 1991. He was a research associate at National Research Council, Ottawa, Canada working in the area of optical interconnects and optical signal processing. He then worked at Nortel Networks, Ottawa, Canada and Fujitsu Network Communications, Richardson, TX in the research and development of optical transport products. In 2004, he joined the Photonics Networking Laboratory of Fujitsu Laboratories of America, where he is currently working in the area of optical networking.