

On Dynamic Multicast Traffic Provisioning with Grooming Capabilities Support in Light-Trail Networks

Ching-Fang Hsu

*Department of Computer Science and Information Engineering, National Cheng Kung University,
1 Ta-Hsueh Road, Tainan, Taiwan*

Tzu-Huan Tang

*Department of Computer Science and Information Engineering, National Cheng Kung University,
1 Ta-Hsueh Road, Tainan, Taiwan
E-mail: cfhsu@mail.ncku.edu.tw*

Received 11 April 2013

Accepted 19 August 2013

Abstract

In recent years, because of the popularity of Internet and the rapid growth of multimedia applications, the requirements of bandwidth from users increase violently in optical networks. Various multicast multimedia services also bring challenges to next generation optical networks. As the rapid advance of DWDM, the bandwidth of a single fiber link is significantly improved. However, the traditional circuit switching architecture, lightpath, has the disadvantage with lower bandwidth utilization of a single wavelength. Light-trail has been proposed as an efficient solution to support optical networks. Compared with lightpath approach, a light-trail can achieve fast provisioning for multiple connections without optical switching. The support of sub-wavelength granularity in light-trail network increases the utilization of wavelength bandwidth effectively. The most distinctive feature is that light-trail allows multiple node access, so it possesses natural multicasting capability inherently. In a light-trail, the single send by one node will be received by the downstream nodes. In other words, light-trail has the advantage of natural multicasting. Furthermore, incorporating traffic grooming into light-trail, it can obviously become a strong candidate to support multicasting and multi-granularity bandwidth demands in the future. In this article, we propose a dynamic multicast routing algorithm with traffic grooming consideration in light-trail networks, named Dynamic Light-Trail-based Multicast Routing algorithm (DLTMR). For static traffic, we use a set of ILP formulations to calculate the optimal solution for minimizing the number of established light-trails. In the simulation, we use various network topologies and variant limitation of network resource to verify that DLTMR can efficiently improve blocking performance in the light-trail networks.

Keywords: light-trail; multicast; traffic grooming

1. Introduction

With the great advances of DWDM (Dense Wavelength-Division) technology in the past years, a fiber link can offer much more wavelengths to various optical switching paradigms. Moreover, the bandwidth per wavelength became much huger as well. As we know, applications aiming at multimedia data

transmission via Internet are widely developed. Although user demands for bandwidth are rapidly growing, a significant gap still exists between their bandwidth requirement and per wavelength capacity. In wavelength-routed networks without conversion capability, a connection is assigned to an all-optical channel from source node to destination node. Such a communication channel is called a lightpath and its

granularity is the whole wavelength. With lightpaths, per wavelength capacity is dedicated to the source-destination node pair and it is expectable that such a coarse granularity would result in a lot of idle but not useable bandwidth within a lightpath. For better bandwidth utilization, light-trails have been proposed as a promising solution.¹⁻¹⁶ Moreover, it is seldom that a traffic flow in IP-centric networks requires bandwidth around the entire per wavelength capacity. Based on this rationale, it is evident that lightpath is not flexible enough and would cause vital bandwidth wastage under such a dynamic regime. For higher bandwidth sharing, a mechanism called traffic grooming has been addressed in the past ten years.¹⁷ Traffic grooming is a technique to aggregate several low-speed connection flows to a single high-speed trunk and to provide resource sharing among multiple entities. In DWDM networks, more than two flows which destine to the same target node can be allocated to the same wavelength if the total required bandwidth of these flows does not exceed the capacity of a wavelength. With traffic grooming capabilities support, bandwidth utilization can be more effective and certainly reach higher degree of resource saving.^{1, 2, 12, 15, 17}

Light-trail is a unidirectional optical path occupying single wavelength. The major advantage of light-trail over lightpath is its higher bandwidth sharing. In other words, not only the source-destination node pair but also all intermediate nodes on a light-trail can access the channel as well. Moreover, a light-trail can provision multiple connections without optical switching and apparently the cost of data transmission can be significantly reduced. Fast provisioning indicates that light-trails would be more suitable and potential for dynamic transmission in the optical networks. In addition, light-trails provide sub-wavelength granularity to efficiently utilize the capacity of a wavelength. In this article, we aim at the mechanism dealing with multicasting and traffic grooming problems in dynamic light-trail networks. With the objective to minimize the network resource including the number of wavelength-links and the established light-trails, we propose an auxiliary graph model and an algorithm named Dynamic Light-Trail-based Multicast Routing algorithm (DLTMR) to find the appropriate route for each incoming multicast request. In order to demonstrate the performance and the improvement of proposed scheme, we use the Integer Linear Programming (ILP)

formulations to find the optimal solution for the static case, which has a given set of multicast requests. In the dynamic case, we compare the performance of DLTMR with previous schemes in terms of blocking probability as well as the number of established light-trails.

The rest of this article is organized as follows. In the next section, we introduce the concept of light-trails such as node architecture traffic grooming. Section 3 gives an overview of related work. In section 4, we elaborate on DLTMR in detail. We first show the ILP formulation for static traffic pattern and then discuss the proposed auxiliary graph model and heuristic algorithm for the dynamic case. Numerical results are demonstrated and discussed in section 5. Finally, section 6 gives a brief conclusion of this work.

2. The Concept of Light-trails

A light-trail is quite analogous to a lightpath since both of them set up an optical path on a single wavelength. The major differences between the two paradigms involve the limitations on transmission directions, the degree of bandwidth sharing, and the efficiency of bandwidth provisioning. First, light-trails offer unidirectional optical channel only while lightpaths support both unidirectional and bi-directional data delivery. Moreover, all nodes except the end node of a light-trail are allowed to access it but the bandwidth of a lightpath is dedicated to its source-destination pair only. More specifically, any node in a light-trail can perform data delivery to its downstream nodes and receive data from its upstream nodes. In a light-trail, the first node is called the convener node and the last node is called the end node.¹⁻¹⁶ If a light-trail is composed of n nodes, it can support at most $\binom{n}{2}$ connections which are non-overlapping in time-domain without extra optical switching.

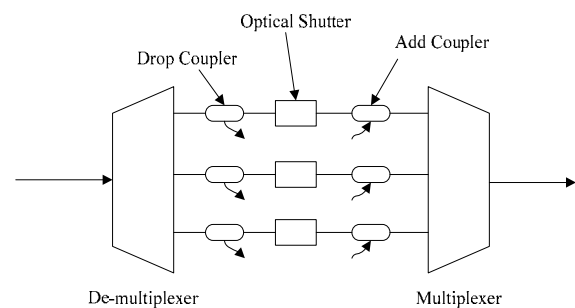


Fig. 1. The node architecture of light-trails

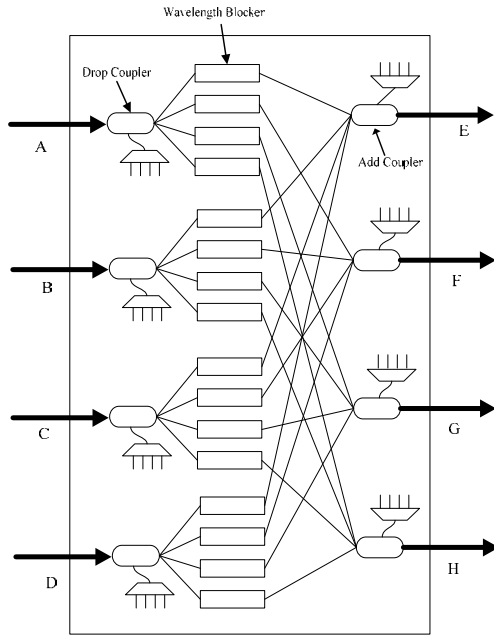


Fig. 2. A node configuration of light-trail for mesh networks ($D=4$)

Fig. 1 shows the node architecture of light-trails.¹⁻¹⁶ A light-trail node is composed of add couplers, drop couplers and optical shutters. The functionalities of add couplers and drop couplers are data uploading and data downloading respectively. As to the optical shutter, it performs just like an ON/OFF switch. When an optical shutter is in ON state, data can go through the light-trail node continuously to the next node. Otherwise, optical signal on the light-trail would be blocked directly if the optical shutter is set to OFF state. In a light-trail, optical shutters at the convener node and the end node must be set to OFF state to prevent interference from irrelevant signal and unnecessary data expose. On the other hand, optical shutter at intermediate nodes are configured to ON state to allow traffic from upstream nodes to be passed to downstream nodes.

In Ref. 8, the authors proposed a new architecture for meshed light-trail networks as shown in Fig. 2. They extended the architecture in Fig. 1 to handle cases with arbitrary input/output port degree. The node consists of three types of components: add couplers, drop couplers and wavelength blockers. The first two types remain identical to those in Fig. 1 in both functionalities and names. The obvious distinction is that optical shutter is replaced with wavelength blocker. Actually, authors

merely used an alternative term to name the same component. Let D denote the degree of a switch node and we illustrate a node with $D=4$ in Fig. 2. As we can see in Fig. 2, the WDM signal entering the node is split into $(D+1)$ outputs at the drop coupler, where one is dropped to the local node. Similarly, the add coupler is realized by a $(D+1):1$ combiner.

3. Related Work

In Ref. 2, due to the power loss at each optical splitter and the optical noise within each fiber link, the hop length of a light-trail should be limited to ensure the signal quality. If the distance between a source node and the corresponding destination node is longer than the hop length limitation, the traffic must be transmitted through multiple light-trails to reach the destination node. The node connecting one light-trail to another light-trail is called a hub node. The authors of Ref. 2 addressed the unicast routing problem with sparsely hub nodes in the light-trail network by forming ILP formulations and a heuristic algorithm. In Ref. 20, the authors proposed an auxiliary graph to solve unicast routing problem for traffic grooming. The concept of auxiliary graph proposed in Ref. 20 is based on the concept proposed in Ref. 21. The information of available resource in network can be all described on the auxiliary graph and therefore the routing problem can be solved directly and efficiently by selecting the shortest path on the auxiliary graph. By differentiating edge weights in auxiliary graph, it can result in varied grooming policies for various purposes. For example, the goal might be to minimize the physical hop length or to increase the utilization of established light-trails.

In Ref. 16, authors have proven that the problem of static multicast routing problem with minimization of established light-trails is a NP-hard problem. They also solved multicast routing problem by reducing it to minimum Steiner tree problem in general light-trail networks and exploit existing efficient heuristic algorithms to the Steiner tree problem to find the multicast tree with minimum number of light-trails.¹⁸ However, these solutions are based on a critical assumption that there exist some light-trails in the network. In other words, these algorithms cannot work in a preliminary stage, i.e., there is no existing light-trail yet. We would design a generalized solution, suitable to both scenarios with and without existing light-trails.

The Tune-In Light-Trail (TILT) work was proposed in Ref. 19. The authors proposed a light-trail node architecture with traffic grooming capabilities support as shown in Fig. 3. Let W be the number of wavelengths per fiber link and N_i stand for arbitrary intermediate node of an existing light-trail. As depicted in Fig. 3, a light-trail node equips with W tunable transmitters and W tunable receivers. When a light-trail is established, only one transmitter in the convener node and one receiver in the end node (operates on the same wavelength of the transmitter in the convener node) are allocated. If N_i simply acts as a relay node, data on the light-trail bypasses via the optical shutter and free transceivers can be assigned to other light-trails. On the other hand, if node N_i needs to send data to its downstream node(s) in an existing light-trail, it first tunes its receiver to listen to the light-trail channel. If the channel is idle, N_i then tunes one of its free transmitters to the light-trail channel for data delivery. Conversely, in case N_i needs data from some upstream node, it would tune one of its free receivers to receive data.

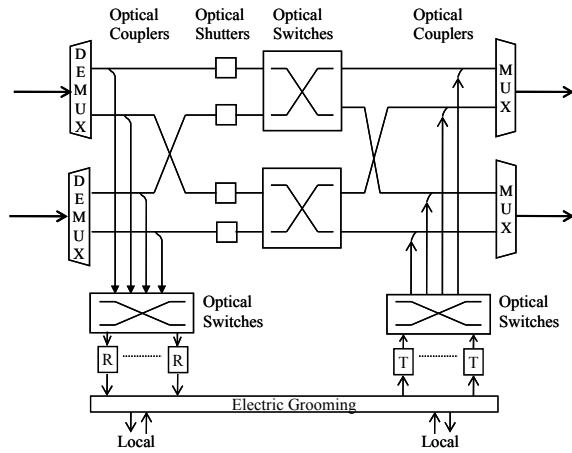


Fig. 3 The node architecture for traffic grooming¹⁹

In addition to the node architecture, the authors of Ref. 19 proposed an auxiliary graph model for dynamic traffic grooming algorithm in TILT networks. Let W denote the number of wavelengths on each fiber and $G_0 = (V_0, E_0)$ stand for the given physical network topology, where V_0 is the set of nodes and E_0 is the set of edges. Then the model generates the following three auxiliary graphs.

- ◆ Sub-auxiliary Graph 1 (SG1): SG1 is used to represent the free wavelengths in the

networks. It is composed of W layers called Wavelength Layers (WL's). If the wavelength λ_w in the fiber between node pair (i, j) in G_0 is free, there is an arc connecting this node pair with full capacity in SG1. Because the authors do not consider the wavelength conversion, there is no arc connecting between wavelength layers.

- ◆ Sub-auxiliary Graph 2 (SG2): Similar to SG1, SG2 has W layers and each layer corresponds to a wavelength uniquely. These layers called Light-trail Layers (LLs). SG2 is used to record the established light-trails in the network. Traffic grooming can be realized among existing light-trails by finding the least-weighted path in SG2. Authors defined three vertex types for SG2: (i) Transmitting vertex (T): A T vertex is used to represent a transmitting port of one wavelength in a node. There are W T vertices within a node in SG2, one for each wavelength. A T vertex connects only to an R vertex (described below) within the neighbor node in the current light-trail. (ii) Receiving vertex (R): This vertex is used to represent a receiving port of one wavelength in a node. There are W R vertices within a node, one for each wavelength. (iii) End vertex (EX): An EX vertex is used to abstract the virtual source (S) and virtual destination (D) nodes.
- ◆ Integrated auxiliary graph IG: The integrated auxiliary graph IG is composed of SG1 and SG2. Therefore, there are $2W$ layers in an IG. Authors added some arcs with weight E_G between WLS and LLS in IG. Connections via these arcs can be groomed from WLS to LLS.

With above auxiliary graphs, the algorithm proposed in Ref. 19 can be described as follows.

Input	Network topology $G_0=(V_0, E_0)$ and an connection request $R(S, D, d_size)$, where d_size is the traffic demand size, S is the source node and D is the destination node
Step 1	If we can find a light-trail which residual capacity is more than or equal to d_size and S and D are both on the light-trail, go to Step 5; otherwise, go to Step 2.
Step 2	Try to establishing the connection by traffic grooming through existing light-trails. Find a least-weighted path from S to D in SG2 by Dijkstra's algorithm. If the least-weighted path can be found, the connection can

- be supported by traffic grooming with some existing light-trail, go to Step 5. Otherwise go to Step 3.
- Step 3 Try to establish a new light-trail to support this connection. Run Dijkstra's algorithm in SG1 and choose one of the paths with shortest hops to establish a new light-trail. If a new light-trail can be set up successfully, go to Step 5. Otherwise Step 4.
- Step 4 If the least-weighted path from S to D in integrated graph IG is found, we can establish this connection with traffic grooming and set up new light-trails, and then go to Step 5. Otherwise go to Step 6.
- Step 5 Update the residual capacity of the used light-trails. Wait for next connection request.
- Step 6 Block the request and wait for next connection request.
-

Let N be the number of nodes and M be the maximum nodal degree in $G_0 = (V_0, E_0)$. If V_{IG} denotes the set of vertices in IG, the worst-case time complexity of the above algorithm is dominated by the time complexity of Dijkstra's algorithm, i.e., $O(|V_{IG}|^2)$, where $O(|V_{IG}|) = O(NMW)$.

4. Proposed Scheme

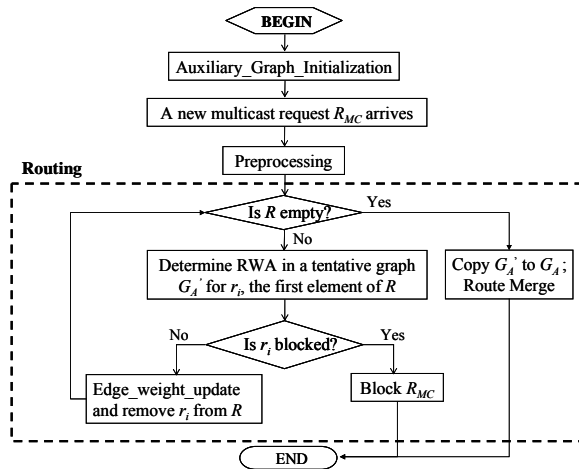


Fig. 4. The flowchart of DLTMR

Given a network topology $G = (V, E)$, where V is the set of nodes and E is the set of fiber links. Let W be the number of wavelengths per fiber link and C represent the capacity per wavelength. A multicast request is denoted as $R_{MC} = \{s, D_set, d_size\}$, where s is the source node of the multicast request, D_set is the set of destination (member) nodes, and d_size is the size of requested bandwidth. The objective of this work is to determine the multicast route for R_{MC} such that the number of light-trails used is minimized and the blocking performance can be improved. In this section,

we propose a dynamic multicast routing algorithm with traffic grooming consideration, which is named Dynamic Light-Trail-based Multicast Routing algorithm (DLTMR). DLTMR adequately exploits two important properties of light-trails, i.e., optical multicasting and optical grooming, to solve the multicast routing problem in light-trail networks. The flowchart of DLTMR is given in Fig. 4. We would elaborate on individual stages of the flowchart in the following subsections.

4.1. Preprocessing

The first stage of DLTMR is the preprocessing. The initialization of the preprocessing is to split R_{MC} into several unicast requests, say $r_1, r_2, \dots, r_{|D_set|}$, and collect them to R . The node d_i corresponds to the element in D_set for each r_i , where $1 \leq i \leq |D_set|$. For the given network topology, the preprocessing procedure creates a database, called candidate database, to incorporate all feasible light-trails whose hop lengths are within the hop limitation. And then DLTMR checks whether all or partial of these unicast requests can be supported in a single light-trail. If such light-trails exist, DLTMR preferentially takes them into account because the establishment of dispensable light-trails can be eliminated as possible. Finally, elements in R are sorted in decreasing order according to their hop length of shortest path. The rationale behind the sorting operation is that in case establishing a new light-trail is inevitable, a longer light-trail can support more connections and better wavelength sharing can be achieved. The pseudo code of the preprocessing is given below.

Preprocessing stage

Input: $R_{MC} = \{s, D_set, d_size\}$

Output: R

```

1  begin
2  R ← ∅;
3  for each destination node d_i ∈ D_set do
4  r_i ← (s, d_i, d_size);
5  R ← R ∪ {r_i};
6  end for
7  Check whether more than one unicast request can
   be supported in a single light-trail whose hop
   length is within the hop limitation. If such light-
   trails exist, take them into account preferentially in
   the following stages of DLTMR;
8  Sort all elements in R in decreasing order
   according to their hop lengths of shortest paths;
9  return R;
10 end
  
```

4.2. Auxiliary Graph Model and Routing Issue

As shown in Fig. 4, the step after preprocessing is to route each unicast request splitting from R_{MC} . DLTMR generates an auxiliary graph to accomplish routing problem efficiently. If all of the unicast requests can be provisioned by any existing light-trails or setting up the new light-trails, updates the resource available and wait for next multicast request incoming. On the other hand, if no route is found for one of the unicast requests, block the request R_{MC} .

TABLE 1. Notations used in auxiliary graph generation

N	The number of nodes in the light-trail network, i.e., $N = V $
Λ	The set of wavelengths in the network, $\Lambda = \{\lambda_k \mid 1 \leq k \leq W\}$
L	The set of existing light-trails in the network, $L = \{l_i\}$, where l_i is light-trail i
G_A	The auxiliary graph $G_A = (V_A, E_A)$, where V_A is the set of auxiliary nodes and E_A is the set of edges in G_A
$AGnode_i$	The node i in V_A , i.e., $AGnode_i \in V_A$, where $1 \leq i \leq V_A $
$degree_i$	The nodal degree of node i

As to the auxiliary graph construction, let us summarize required notations in TABLE 1 for better readability. The proposed auxiliary graph model is based on the concept of layer-graph. The layer i contains network status in the wavelength λ_i . We define four types of auxiliary nodes in auxiliary graph: (1) Transmit (TX) node: A TX node $T_{n,i}^j$ is used to represent λ_j on the output port i of node n . If the wavelength is free, $T_{n,i}^j$ can connect to the RX node $R_{m,k}^j$ if $(m, n) \in E$ from port i of node n to port k of node m . (2) Receive (RX) node: A RX node $R_{m,k}^j$ stands for λ_j on the input port k of node m . In addition to the connection to TX nodes, a RX node $R_{m,k}^j$ also can connect to VDEST node which has the same physical node id m . (3) Virtual source (VSRC) node: A VSRC node S_n is used to represent the physical source node, say node n , of the unicast request and is the starting point for finding the least-weighted route. The virtual source node S_n can connect TX node $T_{n,i}^j$ for each output port i of node n , $1 \leq j \leq W$. (4) Virtual destination (VDEST) node: A VDEST node D_m represents the physical destination node, say node m , of the unicast

request and is the ending point of the least weight route. The virtual destination node D_m can connect to RX node $R_{m,k}^j$ for each input port k of node m , $1 \leq j \leq W$.

TABLE 2. Types of edges in E_A and the corresponding weights

Edge type	Description	Weight
LT_edge	Links which are used by existing light-trails	w_{e1} : for 1~2-hop light-trails w_{e2} : for 3~5-hop light-trails $w_{e1} \gg w_{e2}$
$Free_wavelength_edge$	At least one free wavelength exists in the corresponding physical link	w_{free}
$Bypass_edge$	Edges from a RX node to a TX node within a switch node	w_{bypass}
$LT_multihop$	The traffic can be groomed between different light-trails	w_{mh-cl} : for cross-layered arcs w_{mh-ncl} : for non-cross-layered arcs $w_{mh-cl} \gg w_{mh-ncl}$

In addition to nodes, we define four types of edges to reflect the resource usage and the establishment of light-trails in the physical network. The four types of edges and their corresponding edge weights are summarized in TABLE 2. In general, the relationship among these edge weights is $w_{free} \gg w_{mh-cl} \gg w_{mh-ncl} \gg w_{e1} \gg w_{e2} \gg w_{bypass}$. For higher resource utilization, our objective is that existing light-trails can accommodate as many requests as possible. Hence, the edge weights w_{e1} , w_{e2} , and w_{bypass} must be much smaller than other weights. The reason why $w_{e1} \gg w_{e2}$ is that longer light-trails can support more connections than shorter ones. Shorter light-trails therefore can be relinquished for later requests. As to $w_{mh-cl} \gg w_{mh-ncl}$, it is because we would like to eliminate wavelength-discontinuous assignment as possible so that longer wavelength-continuous paths can be left for pending multicast requests. The initialization of auxiliary graph is given as below.

Algorithm: Auxiliary_Graph_Initialization

Input: $G = (V, E)$, W , $G_A = (V_A, E_A)$, $degree[1..|V|]$

Output: $G_A = (V_A, E_A)$

```

1  begin
2   $V_A \leftarrow \emptyset$ ;  $E_A \leftarrow \emptyset$ ;
3  for each  $AGnode_i \in V_A$  do
4  if  $AGnode_i$  is a VSRC node then
5  Add an unidirectional arc from  $AGnode_i$  to
   all TX nodes which have the same
   physical node id as  $AGnode_i$ ;
6  else if  $AGnode_i$  is a TX node then
7  Add an unidirectional arc with weight
    $Free\_wavelength\_edge$  from  $AGnode_i$  to
   the RX node within the same wavelength

```

```

layer such that the physical node of the
RX node connects to the physical node of
AGnodei in G;
8   else if AGnodei is a RX node then
9   Add unidirectional arcs from AGnodei to
the TX nodes that within the same
physical node in G;
10  Add an unidirectional arc from AGnodei to
the VDEST node which has the same
physical node id;
11  end if
12  end for
13  return GA = (VA, EA);
14  end

```

As routing failure encountered by any unicast request $r_i \in R$ would lead to the blocking of the corresponding R_{MC} , we tentatively deal with the routing issue for all $r_i \in R$ in a copy of G_A , say G_A' . If all r_i 's can be setup successfully, G_A' is then copied back to G_A and DLTMR performs route merging; otherwise, just ignore G_A' and terminate DLTMR. In the routing policy, DLTMR prefers to find a single light-trail to accommodate r_i . In other words, the least-weighted route in G_A' is DLTMR's second choice. If such a route does not exist, block R_{MC} and terminate DLTMR. The algorithm used to update edge weights is shown as below and an example is given in Fig. 5.

Algorithm: Edge_Weight_Update

input: $G_A' = (V_A', E_A')$, L , $r_i = (s, d_i, d_size)$
output: $G_A = (V_A, E_A)$

```

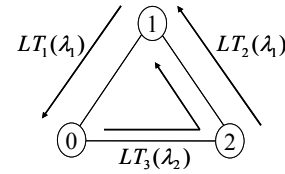
1  begin
2  for each light-trail  $l_k \in L$  do
3  if  $d\_size \leq$  the residual capacity of  $l_k$  then
4  for each arc from TX node to the remote
RX node of the downstream node on  $l_k$  do
5  if the hop length of  $l_k \geq 3$  then
6  Set the arc to  $LT\_edge$  with weight
 $w_{e2}$ ;
7  else
8  Set the arc to  $LT\_edge$  with weight  $w_{e1}$ ;
9  end if
10 end for
11 for each physical node  $n$  in  $l_k$  do
12 if  $l_k$  is the only light-trail that node  $n$ 
joins then
13 Set the edge from RX node which
 $l_k$  enters node  $n$  to TX node which
 $l_k$  continues from node  $n$  to
 $Bypass\_edge$  with weight  $w_{bypass}$ 
14 else if  $n$  joins some other light-trail  $l_l$ 
which is in the same wavelength layer
as  $l_k$  then
15 Set the edge from RX node which

```

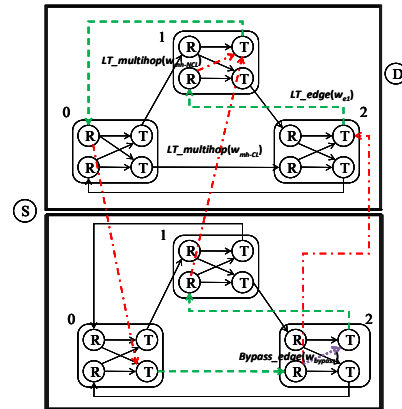
```

 $l_k$  enters node  $n$  to TX node which
 $l_l$  continues from node  $n$  to
 $LT\_multihop$  with weight  $w_{mh-NCL}$ ;
16 else  $n$  joins some other light-trail  $l_l$ 
established in a wavelength different
with that  $l_k$  operates in then
17 Set the edge from RX node which
 $l_k$  enters node  $n$  to TX node which
 $l_l$  continues from node  $n$  to
 $LT\_multihop$  with weight  $w_{mh-CL}$ ;
18 end if
19 end for
20 else
21 Find all arcs by which  $l_k$  passes and set
their edge weights to  $\infty$ ;
22 end if
23 end for
24 end

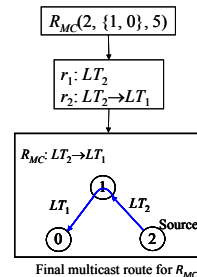
```



(a) A three-node light-trail network



(b) The auxiliary graph of (a)



(c) An example of m/c route

Fig. 5. An example of DLTMR in a light-trail network

As to time complexity analysis, the worst case of the total number of nodes in G_A is $O(W|E|)$. Because DLTMR uses Dijkstra's algorithm to find the least-weighted route in G_A , the worst case complexity is $O(W^2|E|^2)$. The number of destination nodes is $|V|$ at most. Thus, in the worst case, the time complexity of DLTMR to solve the multicast routing for R_{MC} is $O(|V|W^2|E|^2)$, which is more efficient than the method proposed in Ref. 19.

4.3. ILP Formulation

In this subsection, we formulate the ILP in order to find the optimal solution with minimum number of established light-trails for the given multicast requests. As the operation of add couplers (one component of a light-trail node) would result in inevitable power loss, we do not use the traditional method, i.e., the tree topology, for avoiding additional power splitting. Following our previous work in Ref. 15, to solve the multicast routing problem, we first trigger a preprocessing procedure to split a multicast request into several unicast requests. Then the ILP formulation focuses on dealing with the routing problem of these unicast requests. Given a set of multicast requests, after the preprocessing terminates, the light-trail assignment of the generated unicast requests would be calculated by ILP. Roughly speaking, this is not the first work to discuss the traffic grooming problem in light-trail networks. For example, as aforementioned, Balasubramanian, Somani, and Kamal investigated the ILP formulation and heuristic design of the unicast routing problem in light-trail networks with sparsely grooming capabilities in Ref. 2. However, the formulation in Ref. 2 involves many constraint inequalities and certainly results in high computation complexity. For computational simplicity, we assume that all network nodes equip with grooming capabilities and use a simplified ILP formulation. Once all of the unicast requests can be assigned in the light-trail network, the optimal solution to the original multicast problem with the minimum number of established light-trails can be obtained by ILP. The notations used in the following ILP formulation are briefly summarized in TABLE 3.

Given a network topology $G = (V, E)$, where V denotes the set of network nodes and E stands for the set of fiber links, it is assumed that per fiber link contains W wavelengths and the capacity per wavelength can

offer is C . An incoming multicast request R_{MC} is expressed by a three-tuple notation $\{s, D_set, d_size\}$. The first tuple, s , represents the source node of R_{MC} , D_set denotes the membership set, and the last one, d_size , is the size of requested bandwidth. In the remaining of this section, we would briefly explain the preprocessing procedure and then propose the set of ILP formulation we design.

TABLE 3. Notations used in ILP formulation

Notation	Explanation
C	The capacity of a wavelength
W	The number of wavelengths in a fiber link
L	The set of possible light-trails in the network
r	The index of a unicast request
l	The index of a light-trail
B	A very large integer number
λ	The index of a wavelength
d_r	The demand size of a unicast request r
i, j	The index of a network node
LT_l^r	A binary variable representing whether light-trail l can support the unicast request r $LT_l^r = \begin{cases} 1, & \text{if the unicast request } r \text{ can be support by light-trail } l \\ 0, & \text{otherwise} \end{cases}$
LT_l	A binary variable denoting whether light-trail l is carried by the network $LT_l = \begin{cases} 1, & \text{if the light-trail } l \text{ is carried by the network} \\ 0, & \text{otherwise} \end{cases}$
LT_l^λ	A binary variable denoting whether wavelength λ is assigned to light-trail l $LT_l^\lambda = \begin{cases} 1, & \text{if the wavelength } \lambda \text{ is assigned to light-trail } l \\ 0, & \text{otherwise} \end{cases}$
$LT_l^{i,j}$	A binary variable denoting whether light-trail l passes through the link (i, j) $LT_l^{i,j} = \begin{cases} 1, & \text{if the light-trail } l \text{ goes on link } (i, j) \\ 0, & \text{otherwise} \end{cases}$

To find the optimal solution, there are five constraints we should follow. The objective function and the constraints are listed as below.

Objective function:

$$\text{Minimize } \sum_{l=1}^k LT_l \quad (1)$$

Subject to the constraints as follows

- ◆ Request assignment constraint: Each unicast request can be assigned to one and only one light-trail.

$$\sum_l LT_l^r = 1, \forall r \quad (2)$$

- ◆ Capacity constraint: The total amount of unicast requests assigned to a light-trail should not exceed the wavelength capacity C .

$$\sum_r LT_l^r \times d_r \leq C, \forall l \quad (3)$$

- ◆ Light-trail usage constraint: If any unicast request is assigned on light-trail l , LT_l is set to 1; otherwise, $LT_l = 0$.

$$LT_l \leq \sum_r LT_l^r \leq B \times LT_l \quad (4)$$

- ◆ Wavelength continuity constraint: Ensures every light-trail l is given exactly one wavelength.

$$\sum_\lambda LT_l^\lambda - LT_l = 0, \forall l \quad (5)$$

- ◆ Wavelength assignment constraint: Two light-trails pass the same link cannot be assigned the same wavelength.

$$\sum_l LT_l^\lambda - 1 \leq 0, \forall \lambda, \{l: LT_l^{i,j} = 1\} \quad (6)$$

5. Numerical Analysis

In this section, first we derive the optimal light-trail assignment in two network topologies used in Ref. 1. As shown in Fig. 6, the first one is a 6-node topology with 9 links and the other one is a 10-node network with 14 links. The set of ILP formulation is solved using ILOG CPLEX 10.²² The optical signal splitting at each light-trail node brings to power loss. To avoid indistinguishable optical signal due to degradation within light-trail, the hop length of a light-trail is limited in this section.^{4,6} The metric we observe is the number of used light-trails which can support all the given multicast requests. In the following discussion, the optimal answers would demonstrate the number of established light-trails as well as the light-trail assignment of each multicast request in the network. In the other part, we evaluate DLTMR and the method proposed in Ref. 19 (abbreviated as LT_AG here) via simulations in NSFNET, ARPANET and interconnected ring, which are displayed in Fig. 7. The simulation program is developed by SIMPACK library.²³ The major metric we concerned in this simulation is the blocking rate, and the second one is the number of established light-trails.

There are two experiment scenarios in this section: (1) Static scenario: Given a set of multicast requests, compute the optimal solution using ILP and compare the solution with that derived by DLTMR. (2) Dynamic scenario: The arrival information of multicast requests is not known in prior. This scenario is designed to comparing LT_AG with DLTMR on the blocking performance. The capacity of each wavelength is denoted as C . The arrival of multicast requests is a Poisson process, and the holding time of each connection follows negative-exponential distribution with a mean of unity. For a multicast request, the source and destination node are uniformly distributed between all the nodes in network. The cardinality of D_set is uniformly distributed between 1 and $|V|/2$. The demand size of each multicast request is uniformly distributed between 1 and C .

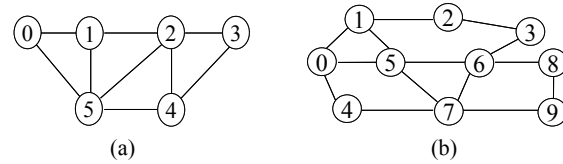


Fig. 6 Network topologies used in the static case (a) 6-node network topology (b) 10-node network topology.

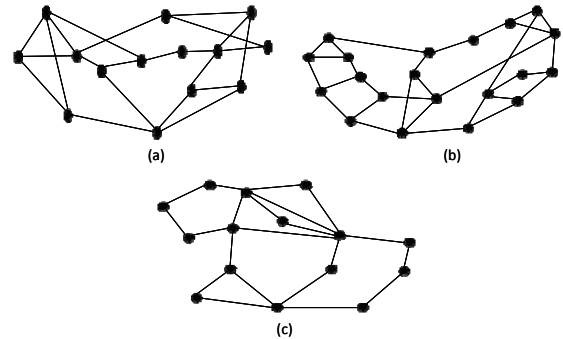


Fig. 7. Network topologies used in the dynamic case (a) NSFNET, 14 nodes and 22 links (b) ARPANET, 20 nodes and 31 links (c) Interconnected ring, 15 nodes and 21 links.²⁴

5.1. Static Case

The first scenario uses the network topology in Fig. 6(a). Each link has two wavelengths, λ_1 and λ_2 . The capacity of each wavelength is 10. Because the network in Fig. 6(a) is a small-scaled one, the hop length limit of each light-trail is within 3. Therefore, the total number of possible light-trails is 120. Three multicast requests are generated randomly and listed in TABLE 4. After

the ILP computation, the minimum number of used light-trail calculated by ILP is 3 and the details of each light-trail are shown in TABLE 5. The illustration of the established light-trails is shown in Fig. 8. The three light-trails can cover all the multicast requests. As to the optimal light-trail assignment, we display it in Fig. 9. With this figure, the multicast route for each multicast request can be observed very clearly. For R_{MC1} , the traffic can be transmitted from source node 0 to destination nodes (node 2, node 3, and node 5) on a single light-trail LT_{66} . The request R_{MC2} can be satisfied by a single light-trail LT_{83} . The multicast route for R_{MC3} is more complex. The traffic is transmitted from source node 5 to destination node 0 by grooming to LT_{82} . In order to reach the destination node 4, the traffic must groom to LT_{83} . And traffic from node 5 to node 3 use LT_{66} for transmitting. In fact, it might have other assignment for the same multicast requests, but the key point at which we aim is the number of established light-trails. With no doubt, three light-trails is the optimal number of used light-trail to support R_{MC1} , R_{MC2} , and R_{MC3} .

TABLE 4. The given multicast requests in light-trail networks for scenario 1

Multicast request	Source node	Destination nodes	Demand size
R_{MC1}	0	2,3,5	2
R_{MC2}	2	1,4	7
R_{MC3}	5	0,3,4	3

TABLE 5. The established light-trails in the optimal solution for scenario 1

Light-trail ID	Light-trail route	Wavelength ID
LT_{66}	0→5→2→3	λ_1
LT_{82}	2→1→5→0	λ_1
LT_{83}	2→1→5→4	λ_2

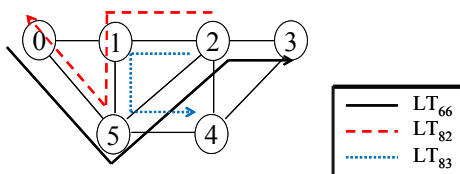


Fig. 8. The illustration of the established light-trails in scenario 1

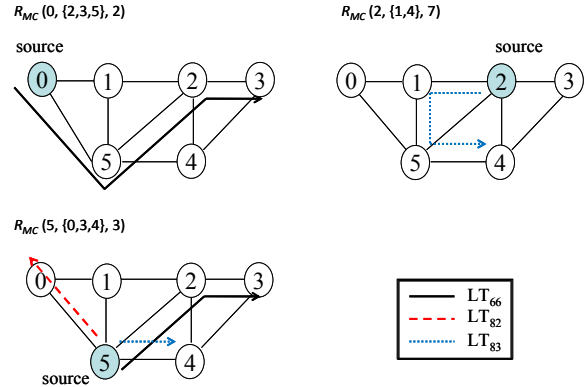


Fig. 9. The illustration of the optimal multicast routes for scenario 1

TABLE 6. The given multicast requests in light-trail networks for scenario 2

Multicast request	Source node	Destination nodes	Demand size
R_{MC1}	5	4,0,8,2	7
R_{MC2}	0	7,9,4,1,2	2
R_{MC3}	4	5,8	5
R_{MC4}	2	3	4
R_{MC5}	6	2,8	5

TABLE 7. The established light-trails in the optimal solution for scenario 2

Light-trail ID	Light-trail route	Used wavelength
LT_{118}	2→3→6→8	λ_1
LT_{140}	5→0→1→2	λ_1
LT_{171}	7→5→6→8	λ_2
LT_{209}	0→4→7→9→8	λ_1
LT_{306}	6→5→0→1→2	λ_2

Next, we use a scenario more complex than the first one to observe the result. The topology we used is the one shown in Fig. 6(b). Each link has two wavelengths, λ_1 and λ_2 . The capacity of each wavelength is 10. The hop length of a single light-trail is 4 at most, thus the total number of possible light-trails in the network is 368. The given multicast requests are listed in TABLE 6. The total splitting unicast requests are 14. The optimal solution is shown in TABLE 7. As we can see, the number of established light-trails is 5 and the number of used wavelengths is 2.

5.2. Dynamic Case

In this case, let us discuss the relationship between blocking performance and traffic load first. Under $W=8$ and $C=8$, DLTMR outperforms LT_AG significantly as shown in Fig. 10. This is because DLTMR differentiates edge weights to reflect resource status more accurately. Differentiated edge weights facilitate least-weighted paths to more thoroughly exploit advantages of light-trails such as grooming as well as multicasting. From Fig. 10, the superiority of DLTMR over LT_AG is more significant in ARPANET and the improvement can reach 58%. The rationale behind this phenomenon is that the number of routes in ARPANET is more than other two topologies owing to its larger average nodal degree.

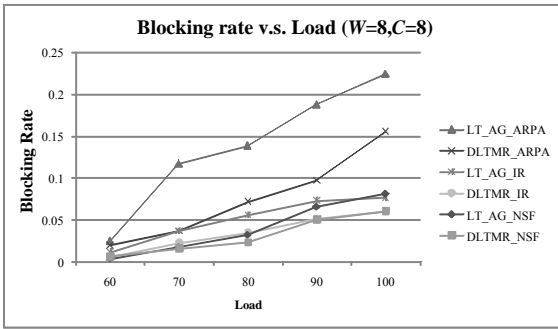


Fig. 10. Blocking rate vs. traffic load ($W=8, C=8$)

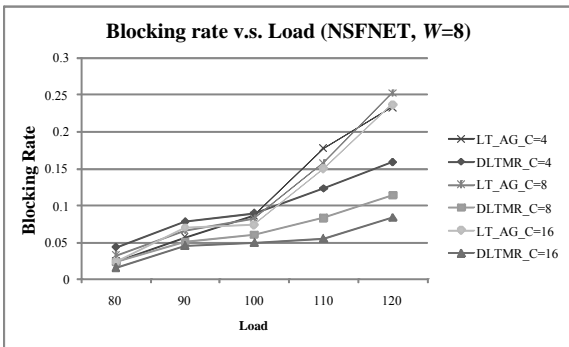


Fig. 11. The effect of traffic granularity on blocking rate (NSFNET)

The effect of traffic granularity on blocking performance in NSFNET is shown in Fig. 11. The larger value of C indicates the smaller traffic granularity because the demand size is uniformly distributed as order of $1/C$. Apparently, the effect of traffic granularity

on blocking performance is not evident to LT_AG scheme. It means that smaller traffic granularity cannot bring benefit to LT_AG. However, on the contrary, the performance gain with larger value of C is significant to DLTMR scheme. Compared with LT_AG, DLTMR is still obviously superior even under heavier load and less capacity ($C=4$). It verifies that the DLTMR can utilize network resource more efficiently than LT_AG does.

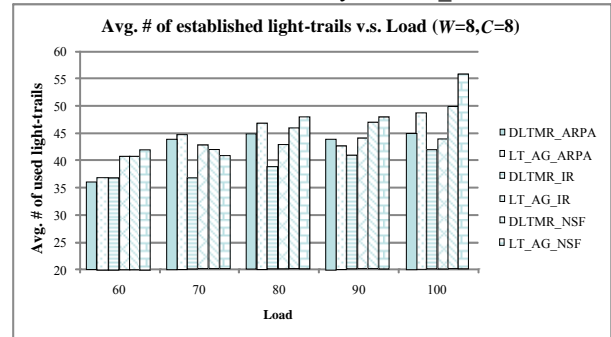


Fig. 12. The average number of established light-trails versus load

In Fig. 12, the simulation result shows that the average number of established light-trails is almost the same between two auxiliary graph models. More accurately, DLTMR merely outperforms LT_AG by 5 at most. However, although the improvement is not apparent, as compared with Fig. 10, we can observe that the blocking performance is significantly improved. It means that DLTMR can utilize network resources more efficiently. The rationale behind this phenomenon lies in that DLTMR always prefers the route containing more existing light-trails with longer hop length and then leaves free wavelengths as many as possible for new multicast requests.

6. Conclusion

In this article, we aim at the multicast routing problem with traffic grooming in light-trail networks. We propose a dynamic multicast routing algorithm based on auxiliary graph model, named DLTMR. In addition, we formulate an ILP to individual unicast request so that the optimal light-trail assignment with minimum number of established light-trails can be obtained. The main concept of DLTMR is to reduce the multicast routing problem to the unicast routing problems owing to the advantages of light-trails such as fast multiple connection provisioning, sub-wavelength

granularity supporting, well optical traffic grooming, and multicast nature. Simulation results show that DLTMR significantly outperforms LT_AG in blocking performance with limited network resources and various traffic granularities. It verifies that the DLTMR can utilize network resource and the characteristics of light-trails more efficiently than LT_AG does. Moreover, numerical data exhibits the optimal solution and the light-trail assignment as well. Compared with similar previous work, our formulation offers an option with lower computational complexity and can be used as a performance bound to corresponding heuristic algorithms.

Acknowledgements

This work was supported by the National Science Council (NSC), Taiwan, under Contract NSC 101-2221-E-006-233-. We would like to express our sincere gratitude to Prof. Ta-Yin Hu, the Chairman of Department of Transportation and Communication Management, National Cheng Kung University, Taiwan, for sharing ILOG CPLEX with us. Without his generosity, we cannot complete numerical data collection for this work.

References

1. A. S. Ayad, K. M. El Sayed, and S. H. Ahmed, Efficient solution of the traffic grooming problem in light-trail optical networks, in *Proc. IEEE ISCC* (2006), pp. 622–627.
2. S. Balasubramanian, A. K. Somani, and A. E. Kamal, Sparsely hubbed grooming light-trail networks, in *Proc. IEEE ICCCN* (2005), pp. 249–254.
3. W. Chen, Y. Fukushima, T. Yokohira, Optimization of token holding times in split light trail networks, in *Proc. IEEE GLOBECOM*, I2011, pp. 1–5.
4. J. Feng, W. He and A. Somani, Optimal light-trail design in WDM optical networks, in *Proc. IEEE ICC* (2004), pp.1699–1703.
5. A. Gumaste, Light-trail and light-frame architectures for optical networks, Ph.D thesis, Fall 2003, EE UT-Dallas.
6. A. Gumaste and I. Chlamtac, Light-trails: an optical solution for IP transport, *OSA J. Opt. Network*, May 2004, pp.864–891.
7. A. Gumaste and I. Chlamtac, Light-trails: a novel conceptual framework for conducting optical communications, in *Proc. IEEE HPSR* (2003), pp.251–256.
8. A. Gumaste and I. Chlamtac, Mesh implementations of light-trails: A solution to IP centric communication in the optical domain, in *Proc. IEEE ICCCN* (2003), pp.178–183.
9. A. Gumaste and S. Q. Zheng, Next generation optical storage area networks: the light-trails approach, *IEEE Commun. Mag.* **43**(3) (2005) 72–19.
10. A. Gumaste, T. Das, A. Mathew, A. Somani, An autonomic virtual topology design and two-stage scheduling algorithm for light-trail wdm networks, *IEEE/OSA J. Opt. Commun. Netw.* **3** (4) (2011) pp. 372–789.
11. X. Luo, B. Wang, On service provisioning using light-trails in wdm optical networks with waveband switching, *Photonic Netw. Commun.* **21** (1) (2011), pp. 97–105.
12. A. K. Somani, A. Gumaste, Light-trails: distributed optical grooming for emerging data-center, cloud computing, and enterprise applications, in *Proc. IEEE OFC/NFOEC*, 2011, pp. 1–3.
13. S. Pal, A. Ranade, Scheduling light-trails on WDM rings, *J. Parallel Distr. Com.* **72** (10) (2012), pp. 1226–1236.
14. W. Zhang, F. Kandah, C. Wang, H. Li, Dynamic light trail routing in WDM optical networks, *Photonic Netw. Commun.* **21** (1) (2011), pp. 78–89.
15. C.-F. Hsu and T.-H. Tang, On multicast routing with traffic grooming in light-trail networks, in *Proc. IEEE AINA* (2012), pp. 566–572.
16. Y. Li, J.-P. Wang, A. Gumaste, Y. Xu, and Y. Xu, Multicast routing in light-trail WDM Networks, in *Proc. IEEE GLOBECOM* (2008), pp. 1–5.
17. A. E. Kamal and R. Ul-Mustafa, Multicast traffic grooming in WDM networks, in *Proc. Opticomm* (Dallas, TX, 2003), pp. 25–36.
18. G. Robins and A. Zelikovsky, Improved steiner tree approximation in graphs, in *Proc. 11th Annual ACM-SIAM Symposium on Discrete Algorithm* (2000), pp. 770–779.
19. Y. Ye, H. Woesner, R. Grasso, T. Chen and I. Chlamtac, Traffic grooming in light trail networks, in *Proc. IEEE GLOBECOM* (2005), pp. 1957–1962.
20. S. Balasubramanian and A. K. Somani, Traffic grooming in statistically shared optical networks, in *Proc. IEEE Local Computer Networks* (2006), pp. 335–342.
21. H. Zhu, H. Zang, and B. Mukherjee, Dynamic traffic grooming in WDM mesh networks using a novel graph model, in *Proc. GLOBECOM* (2002), pp. 2681–2685.
22. ILOG CPLEX 10.1 <http://www.cs.cornell.edu/w8/iisi/ilog/cplex>
23. P. Fishwick, SimPack Toolkit, <http://www.cs.sunysb.edu/~algorithm/implement/simpack/implementation.shtml>
24. R. Ramamurthy and B. Mukherjee, Fixed-alternate routing and wavelength conversion in wavelength-routed optical networks, *IEEE/ACM Trans. on Networking* **10** (3) (2002), pp.351–367.