Photon Netw Commun
DOI 10.1007/s11107-007-0080-3

Enhanced Optimal and Heuristic Solutions of the Routing Problem in Light-Trail Networks

Amr S. Ayad · Khaled M. F. Elsayed · Sanaa H. Ahmed

Received: 3 January 2007 / Accepted: 23 July 2007

Abstract Light-trail is an efficient and feasible technology for IP transport over all-optical networks. The proposition of light-trails for all-optical networks has demonstrated a number of advantages over other paradigms such as Wavelength Routing (WR), Optical Burst Switching (OBS) and Optical Packet Switching (OPS). This article tackles the routing problem of light-trails with the solution objective of minimizing the number of needed light-trails to accommodate an offered traffic matrix. We present two enhancements to the Integer Linear Programming (ILP) formulation of the routing problem. We also propose a computationally efficient routing heuristic for use with static and incremental traffic models. The heuristic is based on routing flows one-by-one. This is done by assigning a set of attributes to each flow and to each network path. The flow attributes are used to determine the order in which flows are presented to the routing algorithm. The path attributes are used to determine which path is selected to route the flow at hand. The efficiency of the proposed heuristic is confirmed using example problems of different network topologies.

Keywords All-optical networks · IP over optical · Light-trails · Traffic grooming · ILP · Heuristic solution

A. S. Ayad · S. H. Ahmed
Faculty of Computers & Information Technology, Cairo University, 12613 Giza, Egypt

K. M. F. Elsayed (✉)
Faculty of Engineering, Department of Electronics and Communications Engineering, Cairo University, 12613 Giza, Egypt
e-mail: khaled@ieee.org

1 Introduction

Present DWDM transport networks are circuit-based backbones carrying TDM, ATM, Ethernet and IP services as overlay networks. This architecture served well as a multi-service transport technology with mature standards in place.

The emergence of converged IP-based services such as triple-play and IP Multimedia Subsystem (IMS) [1] has urged carriers to exploit the benefits of a consolidated IP/MPLS backbone to bring down both capital and operational expenses.

A number of frameworks have been proposed targeting a transport technology that is packet-based and yet provides for efficient use of the inherent circuit-switching nature of wavelength channels offered by DWDM. These are: wavelength routing networks with electronic grooming (lightpaths) [3], Optical Burst Switching (OBS) [4] and Optical Packet Switching (OPS) [5].

In [6], the framework of light-trails was shown to be superior to lightpaths, OBS and OPS in terms of being technologically feasible, allowing for faster service provisioning time, grooming of sub-wavelength demands and adapting to the bursty nature of IP traffic. A brief explanation of the light-trail technology is given below. More details are available in [2, 6, 14].

Light-trails are based on the use of a drop-and-continue sharing scheme of a wavelength channel. Upstream nodes can transmit to downstream nodes but not vice versa. The following description is per wavelength; the mentioned components are replicated for each wavelength carried on a physical network link.

Using a drop coupler, a node couples a portion of the optical energy into its own local receiver. Using an add coupler, any node on the light-trail can transmit traffic to downstream neighbours; only the interested downstream node shall

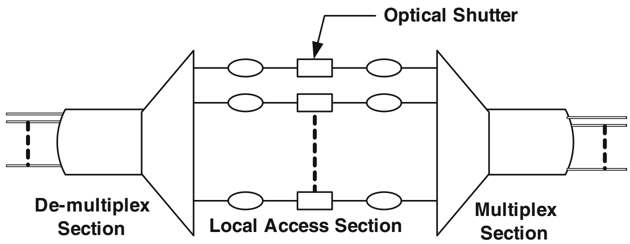

Fig. 1 Architecture of a light-trail node

process the incoming traffic. If more than one node transmits on the light-trail, a mechanism should be used to arbitrate the access (see discussion below on bandwidth arbitration).

Each node is equipped with an optical shutter that can be either switched on or off to either allow the optical signal to further propagate downstream or to be blocked at the given node. The nodes at the two ends of a light-trail (termed the convener and end nodes) configure their shutters to the OFF state for the whole lifetime of a light-trail such that the signals transmitted by light-trail nodes remain confined to it and subsequently allowing for spatial re-use of the wavelength by other light-trails. Figures 1 and 2 illustrate the concept [2].

While the node architecture in Figs. 1 and 2 is simple enough to illustrate the concept; it is only applicable for nodes with a connectivity degree of two and is thus only usable in line or ring topologies. Two other architectures of a light-trail node for use in mesh networks with higher connectivity degree are given in [10] using wavelength-selective power blockers, and in [11] using optical space switches.

Bandwidth arbitration on the light-trail can be either static TDM (round robin or weighted round robin) [12] or stochastic using a MAC protocol [13]. In this article, we assume static TDM arbitration with the transmission time allocated to each node on the light-trail proportional to its demand. When routing several flows on a certain light-trail, we put a constraint that the sum of their traffic demands does not exceed the capacity of the light-trail wavelength channel.

We address the problem of routing flows on a light-trail network in a way that maximizes the grooming of sub-wavelength demands by minimizing the number of signalled light-trails. Better grooming in light-trail networks not only implies more efficient use of network resources such as wavelengths and transceivers, but also provides for considerably faster service provisioning time compared to lightpath or OBS schemes. Routing a new flow on an existing light-trail does not involve any optical switching but merely requires the exchange of out-of-band (OOB) control packets over the Optical Supervisory Channel (OSC) among the participant nodes of the already established light-trail to accommodate the traffic of the new flow.

In [11], electronic grooming of flows is allowed; that is a flow can traverse more than one light-trail with electronic grooming performed at a downstream node that is common to two intersecting light-trails. We do not consider the electronic grooming as it implies additional cost and precludes rate and protocol transparency. We only consider the grooming in the sense of routing flows in a way that packs as many flows as possible on a light-trail for better utilization of its wavelength.

Routing with a static traffic model is a problem in which all offered flows are given before a solution technique starts routing them. Routing with an incremental traffic model is a problem in which no knowledge of future demands is assumed when making the routing decision for the newly requested flow. We assume for the incremental problem that the lifetime of accepted flows is very long such that no flows terminate before routing the whole set of given flows.

Throughout the article, we will refer to the routing problem with static traffic model as the *static problem* and will refer to the routing problem with incremental traffic model as the *incremental problem*.

The rest of this article is organized as follows: in Sect. 2, a simple example is used to present the problem terminology used throughout the article. In Sect. 3, we outline the general approach used to solving the light-trail routing problem. Section 3 also includes the ILP formulation of the static problem as derived in [6] and presents two proposed enhancements to the problem formulation. This is followed by a presentation of the proposed heuristic as applied to the static and incremental problems. Section 4 includes the results of applying our heuristics to sample problems and compare their quality with ILP solution. Section 5 discusses future work and concludes the article.

Fig. 2 Three nodes in a light-trail configuration

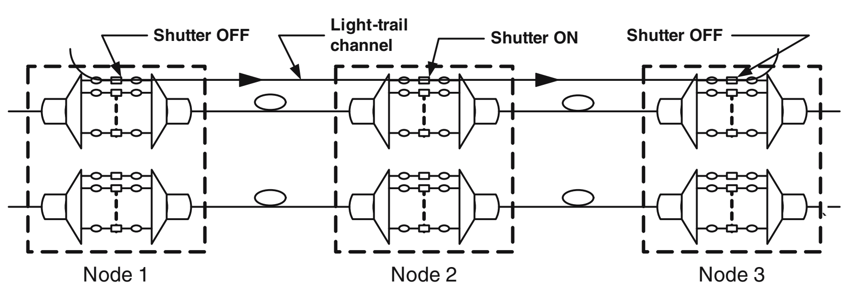

Table 1 An example demand matrix

From/To	1	2	3	4
1	0	5	2	7
2	10	0	17	8
3	30	20	0	35
4	0	5	11	0

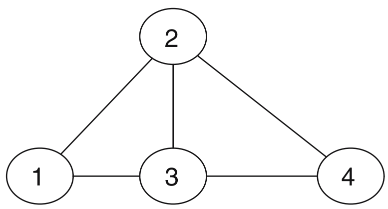

Fig. 3 An example network topology

2 Problem terminology

We use the topology in Fig. 3 to describe the used terminology throughout the article. A sample demand matrix is given by Table 1.

The capacity of a light-trail is equal to the bit rate of its wavelength channel. As in [6], we assume the unit of traffic demand to be equivalent to the speed of an OC-1 signal (51.84 Mbps) and we also assume all wavelength channels to be having the same speed of OC-48 signal (2.48 Gbps). Therefore, the capacity of each wavelength channel is equal to 48 traffic demand units.

We define the set of *eligible paths for a flow* as the set of paths in which every member path includes both the source and destination nodes of a flow and in which the source node is upstream with respect to the destination node. For example, the eligible paths for the flow ($4 \to 1$) are ($4 \to 2 \to 1$), ($4 \to 3 \to 1$), ($4 \to 2 \to 3 \to 1$), ($4 \to 3 \to 2 \to 1$), ($3 \to 4 \to 2 \to 1$) and ($2 \to 4 \to 3 \to 1$). The set of eligible paths depends on the allowed maximum hop count of a path; if we restrict the hop count of used paths to two hops, the set of eligible paths for the flow ($4 \to 1$) becomes only ($4 \to 2 \to 1$) and ($4 \to 3 \to 1$).

Similarly, we define the set of *eligible flows for a path* as the set of flows with both of their source and destination nodes belonging to the path with the source node upstream with respect to the destination node. For example, the eligible flows for the path ($2 \to 4 \to 3 \to 1$) are ($2 \to 4$), ($2 \to 3$), ($2 \to 1$), ($4 \to 3$), ($4 \to 1$) and ($3 \to 1$).

We define a path to be *saturable* if the sum of traffic demands of its eligible flows exceeds its capacity. The eligible flows of path ($3 \to 2 \to 4$) are ($3 \to 2$), ($3 \to 4$) and ($2 \to 4$) with a total traffic demand of 63 units (>48). Therefore, the path ($3 \to 2 \to 4$) is said to be saturable; while the flow ($1 \to 3$) is the only eligible flow for the path ($1 \to 3$) with a traffic demand of 2 (<48). Therefore, the path ($1 \to 3$) is not saturable.

Different light-trails passing by the same network link must use different wavelengths. Each light-trail occupies the same single wavelength on all links between its start and end nodes. Therefore, the number of occupied wavelength links by a light-trail is equal to its hop count. The total number of wavelength links occupied by a certain routing solution is equal to the sum of hop counts of the signalled light-trails.

Finally, we note that maximum grooming is achieved if every occupied wavelength channel is used to its full capacity. A simple, yet potentially loose, lower bound (MinNumLTs) for the number of light-trails that need to be established to carry the offered demand matrix can then be computed as:

$$\text{MinNumLTs} = \sum_{f \in F} \frac{D_f}{C} \tag{1}$$

where C is the capacity of a wavelength channel in traffic demand units, D_f is the traffic demand of flow f, and F is the set of all offered flows.

3 Solution approach

Figure 4 outlines the overall approach that we follow for solving the light-trail routing problem. As proposed in [6], we use Depth First Search (DFS) to enumerate all possible network paths up to a certain hop limit. The eligibility relations among enumerated paths and flows are then determined to generate a set of eligible paths for each flow and a set of eligible flows for each path.

For the static problem, the used routing algorithm can be based on either solving an ILP as in [2, 6, 15] to obtain optimal results or using some heuristic that provides near-optimal results but at a much less computational cost. Our contribution includes two enhancements to the ILP formulation proposed in [6] and a routing heuristic that gives a near-optimal solution.

For the incremental problem, it is computationally expensive to use ILP techniques as an ILP needs to be solved for each incoming flow. Therefore, we only use the proposed heuristic when solving the incremental problem.

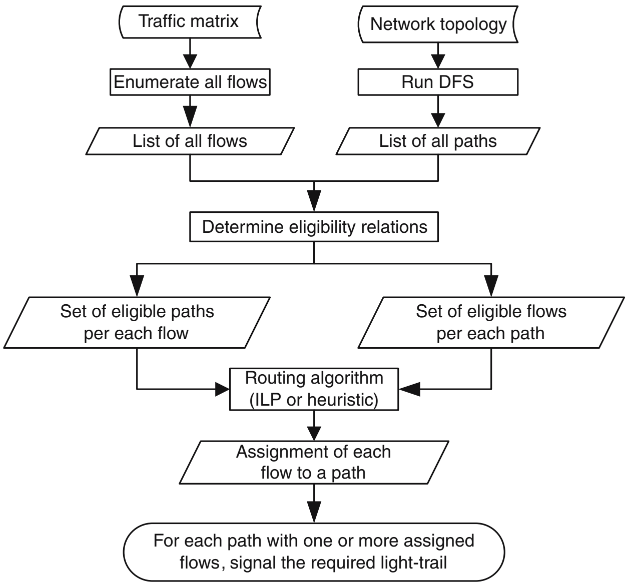

Fig. 4 Overall solution approach

3.1 ILP solution

The problem formulation in [6] is given as follows:

- Parameters:
  For the given directed graph $G(V, E)$, $N = |V|$, let P be the set of all paths discovered by DFS within a hop limit of H, P_f to be the set of eligible paths for flow f, and F_p to be the set of eligible flows for path p. We assume $D_f \leq C$ for all flows as we are only considering demands of fractional wavelength capacity to assess the grooming efficiency of light-trail networks.
- Variables:
  - μ_f^p: binary variable, route indicator, takes a value of 1 if flow f takes path p; zero otherwise.
  - δ^p: binary variable, path usage indicator, takes a value of 1 if path p is used by any flow; zero otherwise.
- ILP Formulation:
  Objective:

$$\min \sum_{p \in P} \delta^p \tag{2}$$

Assignment constraints: Each flow is assigned to one and only one path.

$$\sum_{p \in P_f} \mu_f^p = 1 \quad \forall f, D_f > 0 \tag{3}$$

Capacity constraints: The total amount of traffic routed on a light-trail should not exceed its wavelength capacity.

$$\sum_{F_p} \mu_f^p D_f \leq C \tag{4}$$

Usage constraints: If any flow is assigned on light-trail p, δ^p is set to 1; otherwise, if none of the flows picked light-trail p, $\delta^p = 0$. Recall that δ^p is a binary variable.

$$\delta^p \geq \mu_f^p \quad \forall f \in F, \quad p \in P \tag{5}$$

$$\delta^p \in \{0, 1\}, \quad p \in P \tag{6}$$

In Sects. 3.1.1 and 3.1.2, we propose two enhancements to the above problem formulation.

3.1.1 Removing redundant constraints

We propose an enhancement to the ILP formulation in [6] by making use of the saturable path definition given in Sect. 2. If a path is not saturable, then there is no need to include a capacity constraint for it in the ILP formulation. This can greatly reduce the number of capacity constraints and, hence, the overall problem size. Using the modified ILP formulation, we were able to reduce the solution time of the routing problem in [6] from 2,146 to 1,117 s (about 48% less). The solution was obtained using the open source GLPK library [7] on a Pentium M 2 GHz processor with 1 GB of RAM.

3.1.2 Narrowing the search space

Considering the lower bound in (1), we reduce the solution space by adding the below constraint on the objective value to the problem formulation:

$$\sum_{p \in P} \delta^p \geq \text{MinNumLTs} \tag{7}$$

Upon adding the constraint (7); a large number of sub-problems were readily fathomed and it took relatively less running time for GLPK to find the integer optimal solution.

3.2 Proposed heuristic

The objective of the proposed heuristic is to route the offered traffic using the least number of light-trails. The proposed heuristic operates on two steps. On the first step, the list of offered flows is sorted based on the defined flow attributes. The first step is only usable with the static problem. On the second step, the offered flows are routed one-by-one by the order determined in step 1. On routing a flow, the set of eligible paths with enough spare capacity is determined and then sorted based on the defined path attributes to determine the light-trail over which the flow is routed. Figure 5 shows the flowchart of the proposed heuristic.

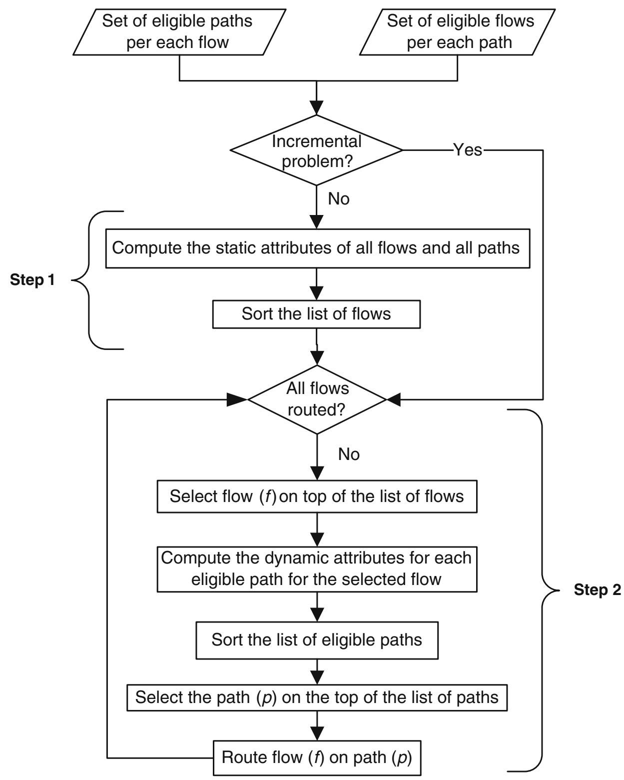

Fig. 5 Flowchart of the proposed heuristic

3.2.1 Problem attributes

The algorithm makes use of a number of *static* and *dynamic* attributes. We define *static* attributes as the attributes whose values are known prior to running the routing algorithm. These values do not change as flows are being routed. In contrast, dynamic attributes are continually updated during the course of the algorithm upon the routing of each flow.

Following is a list of used attributes:

Static path attributes:
EF_p : Number of eligible flows for path p.

$$\mathrm{EF}_p = \left|F_p\right| \tag{8}$$

ED_p: Eligible demand for path p; this is the sum of traffic demands of the eligible flows for path p.

$$\mathrm{ED}_p = \sum_{f \in F_p} D_f \tag{9}$$

Static flow attributes:
EP_f: Number of eligible paths for flow f.

$$\mathrm{EP}_f = \left|P_f\right| \tag{10}$$

D_f: Demand of flow f.

Dynamic path attributes:
RF_p: Number of flows routed on path p.
RD_p: Routed demand for path p, this is the sum of traffic demands of the flows routed on path p.

For the static problem; the algorithm makes use of all the above attributes to route each flow on one of its eligible paths. When solving the incremental problem, the algorithm can only make use of dynamic attributes.

3.2.2 Step 1. Ordering of offered flows

Given the list of flows to be routed, one can arbitrarily select the next flow to be routed. This arbitrary selection is the only option when solving the incremental problem as there are no dynamic flow attributes. However, when solving the static problem, our results show that taking the above flow attributes into account has a considerable effect on bringing the routing results closer to the optimal ILP solution.

Therefore, only in the case of the static routing problem, our heuristic computes the values of the flow attributes for each flow on the flow list and then sorts the list of flows in a way that is computationally efficient while boosting the likelihood of sharing common light-trails among routed flows.

Considering the number of eligible paths (EP_f), we note that this attribute represents the allowed degree of freedom when making the routing decision for a flow. It is thus preferable to begin routing the flows with a limited degree of freedom. This would later allow flows with more eligible paths to prefer the already occupied paths for better sharing of light-trails. Therefore, we sort the list of flows in the ascending order of EP_f and then select the flow on top of the list.

Considering the demand (D_f), if flows with lower values of demand were routed first; then it is more likely for flows with higher demand to be routed later on separate light-trails rather than sharing the already occupied light-trails. The reason is that already occupied light-trails are less likely to have enough capacity to accommodate such flows with high demand value. Therefore, we start routing the flows with higher demand first by sorting the list of flows in the descending order of D_f and then selecting the flow on top of the list.

To avoid sorting the list of flows with multiple attributes; we reduce the computational complexity of the sorting function by combining both flow attributes, using a weighted sum, in one *flow-preference* attribute (Q_f) as follows:

$$Q_f = W_D \times D_f - W_{\mathrm{EP}} \times \mathrm{EP}_f \tag{11}$$

where $W_D, W_{\mathrm{EP}} > 0$

A higher value of Q_f implies a higher value of D_f and a lower value of EP_f. Therefore, a single sorting operation of the flow list in the descending order of Q_f achieves similar effect as sequential sorting based on the two flow attributes.

Setting the values of weights W_D and W_{EP} allows the algorithm to determine which attribute (D_f or EP_f) to be used as the primary sorting attribute. Deciding which attribute to be the primary sorting key is critical to the quality of the obtained routing solution. We propose a simple rule for making this decision by first computing:

$$\mathrm{EP}_{\min} = \min_{f \in F} \mathrm{EP}_f \tag{12}$$

We choose EP_f to be the primary sorting key of the flow list if $\mathrm{EP}_{\min}$ is less than MinNumLTs given by (1). The reasoning is that if $\mathrm{EP}_{\min}$ is higher than the minimum number of routes used to carry the offered flows; then most flows should have a sufficient number of eligible paths to be routed on. The effect of the EP_f attribute is thus less significant than the D_f attribute which should then be used as the primary sorting key.

Having an attribute as the primary sorting key requires its weight to exceed the maximum difference between any two values of other attributes with their weights set to unity.

If D_f is to be used as the primary sorting key (that is $\mathrm{EP}_{\min} > \mathrm{MinNumLTs}$), the weights are calculated as:

$$W_{\mathrm{EP}} = 1,\ \ W_D = \Delta\mathrm{EP}_{\max} + 1 \tag{13}$$

where $\Delta\mathrm{EP}_{\max} = \mathrm{EP}_{\max} - \mathrm{EP}_{\min},\ \mathrm{EP}_{\max} = \max_{f \in F} \mathrm{EP}_f$

If EP_f is to be used as the primary sorting key (that is $\mathrm{EP}_{\min} < \mathrm{MinNumLTs}$), the weights are calculated as:

$$W_D = 1,\ \ W_{\mathrm{EP}} = \Delta D_{\max} + 1 \tag{14}$$

where $\Delta D_{\max} = D_{\max} - D_{\min},\ D_{\max} = \max_{f \in F} D_f,\ D_{\min} = \min_{f \in F} D_f$.

3.2.3 Step 2. Routing of ordered flows

Having selected the flow (f) to be routed next; the algorithm proceeds to select an eligible path (p) with enough spare capacity. As with the flow ordering process, the list of eligible paths is sorted, using the path attributes, in a way that is both computationally efficient and is favourable of routing more flows on highly shared paths rather than using separate underutilized paths.

Considering a path p, a higher value of ED_p and EF_p indicates that the path is more likely to be selected for future flows implying a better opportunity of capacity sharing. A higher value of RD_p and RF_p indicates that the path has been already used to route more past flows; again implying more preference for selecting it to boost capacity sharing.

To avoid sorting with multiple attributes, we combine the path attributes using a weighted sum in one path-preference Q_p attribute as follows:

For the static problem:

$$Q_p = W_{\mathrm{RD}} \times \mathrm{RD}_p + W_{\mathrm{RF}} \times \mathrm{RF}_p + W_{\mathrm{ED}} \times \mathrm{ED}_p + W_{\mathrm{EF}} \times \mathrm{EF}_p \tag{15}$$

For the incremental problem:

$$Q_p = W_{\mathrm{RD}} \times \mathrm{RD}_p + W_{\mathrm{RF}} \times \mathrm{RF}_p \tag{16}$$

where $W_{\mathrm{RD}}, W_{\mathrm{RF}}, W_{\mathrm{ED}}, W_{\mathrm{EF}} > 0$

A higher value of Q_p implies a higher value for all path attributes. Therefore, sorting the eligible path list in the descending order of Q_p achieves the preference objectives of path selection.

Setting the values of weights W_{RD}, W_{RF}, W_{ED} and W_{EF} allows the algorithm to tune which attribute is most prevalent in making the routing decision. Our results have shown no significant effect of the attributes that are based on the number of flows (EF_p and RF_p). Using a value of unity for the W_{EF} and W_{RF} had no effect on the obtained results. This is consistent with the fact the actual network capacity is limited in terms of the channel bit rate capacity rather than the number of flows per channel. We are thus left with the attributes RD_p and ED_p.

Intuitively, the algorithm should always prefer paths with more used capacity when routing a new flow. However, when routing the first few flows, the used capacity (RD_p) is still zero for most paths. Using random tie breaks for those initial routing decisions greatly degrades the quality of the final routing results as the algorithm tends to pack all remaining flows on the initial randomly selected routes. It is therefore critical to make use of the amount of eligible traffic attribute (ED_p) as a guiding parameter for the initial routing decisions.

Our algorithm adopts to the former consideration by setting the weights (W_{ED} and W_{RD}) in a way that allows the ED_p attribute to be the primary sorting key for initial flows and then switches to using the RD_p attribute as the primary sorting key once it begins to have non-zero values. Therefore, the weights are calculated as:

$$W_{\mathrm{RF}} = W_{\mathrm{EF}} = W_{\mathrm{ED}} = 1 \tag{17}$$

$$W_{\mathrm{RD}} = \Delta\mathrm{ED}_{\max} + 1 \tag{18}$$

where $\Delta\mathrm{ED}_{\max} = \mathrm{ED}_{\max} - \mathrm{ED}_{\min}$, and $\mathrm{ED}_{\max} = \max_{p \in P} \mathrm{ED}_p$, $\mathrm{ED}_{\min} = \min_{p \in P} \mathrm{ED}_p$.

It is worth noting that the value of Q_p has the potential to be the same for multiple paths when selecting the path to route the current flow. This is especially more probable for the initial flows when solving the incremental problem, where Q_p is typically zero in (16). In such cases, the heuristic arbitrarily breaks the tie to select a path leading to results that are further from optimal as compared to the results of the static problem.

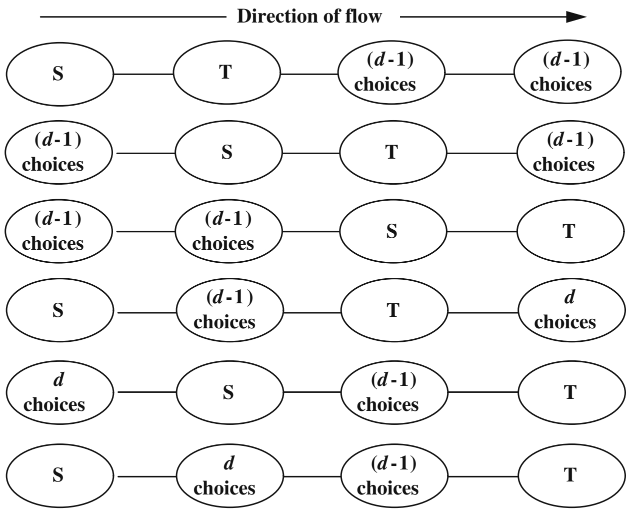

Fig. 6 Computing the number of eligible paths per flow

3.2.4 Complexity analysis

Proposition *The running time complexity of the algorithm described in Fig. 5 is $O(N^2 d^{H-1} log\ d)$, where d is the maximum connectivity degree of a network node and H is the limit on the hop count of DFS discovered paths.*

Proof In step 1, the list of flows is sorted according to the calculated flow-preference (Q_f) given by (11). The Maximum number of unidirectional flows in any problem is $N(N-1)$; that is $O(N^2)$. Thus, the first sorting operation is $O(N^2 log\ N)$.

In step 2, the algorithm loops over all flows to route them one-by-one; so this loop is executed $O(N^2)$ times. In each iteration of this loop; the algorithm sorts the eligible paths per flow according to the calculated path-preference (Q_p) given by (15) (or (16) for the incremental traffic case). This sorting operation have a running time of $O(\text{EP}_f \log \text{EP}_f)$. Therefore, the running time of the two steps is:

$$O(N^2 \log N) + O(N^2\, \text{EP}_f \log \text{EP}_f) \tag{19}$$

On solving the incremental problem, only step 2 is performed and only the second term of (19) applies then. We need to derive $O(\text{EP}_f)$ to substitute in (19). We prove that EP_f is $O(H^3 d^{H-1})$.

Let d be the maximum node degree over all network nodes; let h be the hop count of a path. The derivation follows by considering Fig. 6 which shows the number of eligible paths passing by the source (S) and destination (T) nodes of a flow for the case of $h = 3$.

We see that the number of eligible paths for a flow from node S to node T is equal to $3(d-1)^2 + 3d(d-1)$ which is $O(6d^2)$. We generalize this to the case of any value of h by considering that each row in Fig. 6 corresponds to a different position of the nodes S and T. The number of such rows for a path of h hops is computed as follows: Among $(h+1)$ nodes; node S can take one of h positions ranging from top upstream to penultimate downstream position. When S is the top upstream node; there can be h possible positions for T and when S is the next to top upstream node; there can be $h-1$ positions for T and so on until S is in the penultimate downstream position where T can only take the single position of the most downstream node. Thus, similar to Fig. 6, the number of rows for h hops:

$$h + (h-1) + \cdots + 2 + 1 \Rightarrow (h+1)h/2 \tag{20}$$

For each specific placement of S and T nodes, the remaining number of nodes is $h-1$ nodes. Each node can be selected in $O(d)$ times; therefore there is $O(d^{h-1})$ paths for each specific placement of S and T nodes, which corresponds to one row in Fig. 6. Given (20), we conclude that the number of eligible paths of a hop count of h for a flow f is $O(h^2 d^{h-1})$. Therefore, for a hop limit of H:

$$O(\text{EP}_f) \equiv O\left(\sum_{h=1}^{H} h^2 d^{h-1}\right) \equiv O(H^3 d^{H-1}) \tag{21}$$

Substituting (21) in (19), the worst-case running time of the proposed heuristic for the static problem is:

$$O(N^2 \log N) + O(N^2 H^3 d^{H-1} \log(H^3 d^{H-1})) \tag{22}$$

Hop count (H) is typically a limited constant; the second term of (22) can be reduced as:

$$O(N^2 H^3 d^{H-1} \log(H^3 d^{H-1})) \equiv O(N^2 d^{H-1} \log d) \tag{23}$$

Substituting (23) in (22); the worst-case running time for the static problem is:

$$O(N^2 \log N) + O(N^2 d^{H-1} \log d) \tag{24}$$

As $(d^{H-1} \log d > \log N)$ for typical network topologies, then only the second term of (24) dominates. Furthermore, the complexity of the incremental problem only involves the second term of (24) which completes the proof. □

4 Results

In this section, we solve three problems using both ILP and heuristic techniques. As discussed in Sect. 3.2.3, the results of both incremental and static problems is generally dependent on the initial ordering of offered flows and discovered network paths due to the possibility of making some routing decisions based on random tie breaks. This fact is more pronounced when solving the incremental problem in which only the dynamic attributes counts to the value of Q_p.

To assess that our results are not specific to a particular initial ordering of those lists, we solve each problem 10 times with a different randomized order for each run. We report the result parameters in terms of their average values and the percentage ratio of standard deviation to the average value.

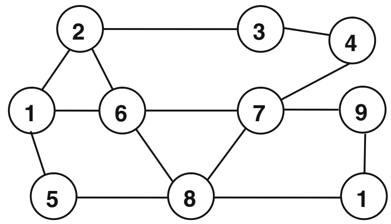

Fig. 7 A 10-node mesh network from [6]

We report the following parameters for a routing solution:

1. Number of signalled light-trails to carry the offered traffic,
2. Number of signalled light-trails carrying 95% of the amount of offered traffic, and
3. The total number of wavelength links used by the signalled light-trails.

4.1 Problem 1. A 10-node mesh network

Figure 7 and Table 2 give the network topology and traffic matrix used in [6], respectively. We assume a hop limit of four hops when discovering network paths using DFS.

The optimal solution routes the offered traffic over 13 light-trails [6]. Using the proposed heuristic, the total traffic was routed over 20 light-trails for the static traffic case. However, more than 94% of the offered traffic was routed over the same optimal number of 13 light-trails. Table 3 gives the routing result of the static problem.

Manual inspection of the routing results reveals that the routing decision for some flows did not serve the purpose of the algorithm. For example, flows (3,1) and (4,2) could have been routed on a single light-trail (4,3,2,1). However, this is an expected "glitch" of the algorithm due to the initial routing decisions attempting to increase the likelihood of more flows sharing the same light-trails. For initial flows, the decision is only based on the amount of eligible demand (ED_p) as

Table 2 Traffic matrix of the network in Fig. 7

From/To	1	2	3	4	5	6	7	8	9	10
1	0	5	11	10	4	5	4	6	6	10
2	8	0	5	5	1	3	1	11	7	2
3	3	0	0	3	0	2	9	4	10	9
4	8	2	11	0	11	6	11	6	9	4
5	11	7	7	6	0	11	3	2	9	9
6	8	9	7	5	4	0	11	8	10	9
7	9	7	11	9	1	10	0	4	11	2
8	6	0	10	4	2	4	4	0	2	9
9	2	9	10	2	6	9	9	8	0	9
10	11	0	10	0	8	10	8	11	4	0

Table 3 Static traffic routing result using the proposed heuristic on the network in Fig. 7

No.	Path	Carried Flows	Load
1	(5,1,6,7,9)	(1,6)(1,7)(5,6)(5,7)(6,9)(1,9)(5,9)	48
2	(5,1,6,8,10)	(5,1)(8,10)(6,10)(1,10)(5,10)	48
3	(9,7,6,2,3)	(7,2)(2,3)(6,3)(9,6)(9,2)(9,3)	47
4	(9,10,8,5,1)	(8,5)(9,8)(9,10)(10,1)(9,1)(10,5)(9,5)	46
5	(9,10,8,6,7)	(8,6)(8,7)(10,8)(9,7)(10,7)(10,6)	46
6	(10,8,7,4,3)	(4,3)(7,3)(8,4)(8,3)(10,3)	46
7	(3,4,7,9,10)	(7,10)(3,4)(3,7)(4,10)(4,9)(3,9)(3,10)	46
8	(2,6,8,7,9)	(6,7)(2,6)(7,9)(2,8)(2,7)(8,9)(2,9)	46
9	(3,4,7,6,8)	(6,8)(7,6)(7,8)(4,7)(3,6)(4,8)(3,8)	45
10	(4,7,6,1,5)	(1,5)(7,1)(6,5)(7,5)(4,6)(4,1)(4,5)	43
11	(5,1,6,2,3)	(6,2)(1,2)(5,2)(1,3)(5,3)	39
12	(5,1,6,7,4)	(7,4)(6,4)(1,4)(5,4)	30
13	(2,1,6,8,10)	(1,8)(2,1)(2,10)	16
14	(2,6,8,5,1)	(6,1)(8,1)(2,5)	15
15	(2,3,4)	(2,4)	5
16	(10,9)	(10,9)	4
17	(3,4,7,6,1)	(3,1)	3
18	(4,7,6,2)	(4,2)	2
19	(5,8)	(5,8)	2
20	(9,10,8,7,4)	(9,4)	2

Table 4 Parameters (Avg., %SD) of routing results of the network in Fig. 7

Problem	No. of used LTs	No. of LTs carrying 95% of traffic	No. of used wavelength links
Optimal	13	13	52
Static (Table 3)	20, 0%	14, 0%	71, 0%
Incremental	26.9, 6.8%	21.3, 7%	91.7, 4.2%

Table 5 Traffic matrix M1 of NSFNET topology

From/To	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	4	0	25	1	42	1	2	0	1	1	0	0	2
2	1	0	0	1	2	28	0	9	1	0	39	0	0	1
3	34	0	0	0	2	2	6	15	27	2	2	0	0	1
4	1	2	1	0	2	18	0	1	0	44	1	2	1	2
5	1	0	2	40	0	0	40	1	0	1	1	1	2	1
6	0	2	2	2	1	0	0	2	1	2	1	13	40	31
7	1	0	0	2	36	0	0	19	20	1	1	10	1	1
8	18	2	0	2	0	2	0	0	29	1	2	1	37	1
9	0	23	1	24	0	1	31	3	0	28	0	1	2	0
10	25	1	2	1	34	0	1	3	2	0	2	0	2	0
11	1	1	45	0	0	0	1	38	0	0	0	1	0	1
12	0	0	46	1	2	1	1	1	2	1	2	0	0	39
13	2	1	0	1	2	2	2	2	10	0	36	37	0	2
14	1	46	1	0	1	0	1	0	1	2	0	26	0	0

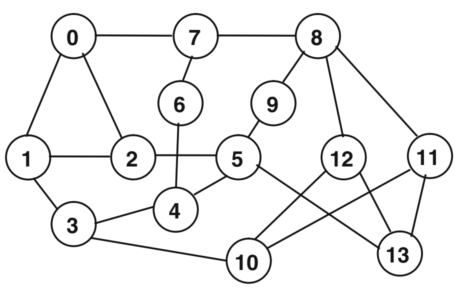

Fig. 8 NSFNET topology

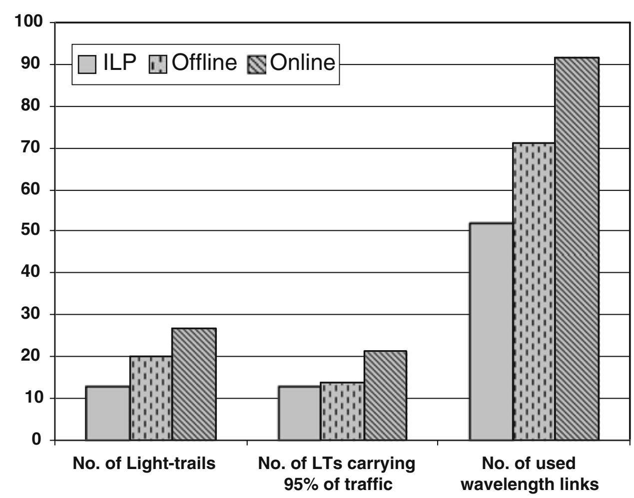

Fig. 9 Results for 10-node mesh network in Fig. 7

the amount of routed traffic on all eligible paths is still zero. Table 4 and Fig. 9 show the routing results.

4.2 Problem 2. NSFNET model

Figure 8 shows the 14-node NSFNET network given in [8]. We assume a hop limit of four hops.

We solve this problem two times with two different traffic matrices with varying demand intensities derived from [8]. On solving the ILP problem for both cases; GLPK could not find the optimal integer solution in a reasonable running time. Therefore, we had to use the achieved feasible solution after a sufficiently high number of iterations (20 millions).

We use a similar idea as in [8] to generate the first traffic matrix, Table 5, which captures a situation where most of the network traffic is concentrated among 42 pairs, with little traffic among the remaining ones. The achieved ILP solution used 30 light-trails. The static traffic heuristic was able to use an average of 44.7 light-trails over 10 runs. Table 6 and Fig. 10 show the routing results.

The second traffic matrix, Table 7, corresponds to a measured traffic distribution taken from [9] with traffic distributed more evenly over a large number of source-destination pairs. The achieved ILP solution used 33 light-trails. Solving the static problem using the proposed heuristic; we were able to route the offered traffic over an average of 40.7 light-trails. Table 8 and Fig. 11 show the routing results.

Table 6 Parameters (Avg., %SD) of routing results of the NSFNET topology using traffic matrix M1

Problem	No. of used LTs	No. of LTs carrying 95% of traffic	No. of used wavelength links
ILP (non-optimal)	30	28	117
Static	44.7, 2.1%	28.8, 1.5%	161.6, 1.6%
Incremental	49.6, 6.5%	31.3, 2.6%	166.1, 4.7%

Table 7 Traffic matrix M2 of NSFNET topology

From/To	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	2	2	0	0	0	0	2	2	0	0	2	0	0
2	6	0	6	2	6	4	2	8	2	8	2	18	4	6
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	2	8	0	2	2	2	2	2	2	2	4	0	2
5	0	32	10	2	0	2	6	30	12	10	0	38	6	4
6	0	4	2	4	2	0	2	2	2	2	0	2	0	2
7	2	12	48	2	12	4	0	48	10	12	2	34	0	16
8	2	30	48	6	12	2	46	0	22	16	6	24	4	8
9	4	10	18	4	12	4	12	28	0	18	8	48	12	2
10	0	16	6	2	12	6	4	28	4	0	4	10	4	2
11	0	2	2	2	0	0	2	2	2	4	0	2	0	2
12	2	12	2	4	12	2	4	20	14	12	6	0	2	2
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	2	14	8	6	18	6	4	14	4	0	2	0	6	0

Table 8 Parameters (Avg., %SD) of routing results of the NSFNET topology using traffic matrix M2

Problem	No. of used LTs	No. of LTs carrying 95% of traffic	No. of used wavelength links
ILP (non-optimal)	33	30	117
Static	40.7, 1.2%	28.7, 1.7%	141.3, 1.3%
Incremental	50.1, 6.3%	35.7, 6.3%	160, 6.2%

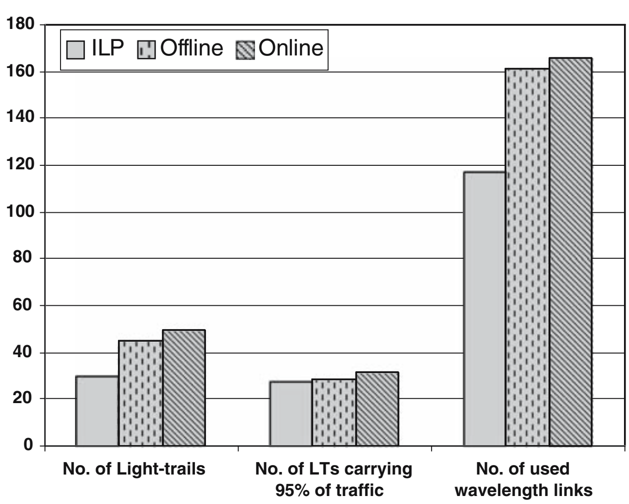

Fig. 10 Results of NSFNET with traffic matrix M1

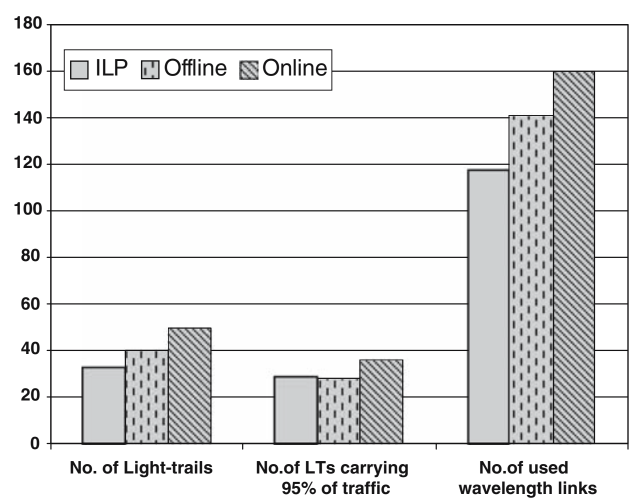

Fig. 11 Results of NSFNET with traffic matrix M2

5 Conclusion and future work

Light-trails provide a technologically feasible alternative for the efficient use of DWDM channels when transporting IP traffic. Proper routing of traffic flows is essential to the establishment of light-trails carrying more flows per DWDM channel and thus achieving higher utilization.

We have proposed two enhancements for reducing the number of constraints and narrowing the search space of the ILP formulation of the light-trail routing problem. We have also proposed a heuristic that is based on multi-attribute sorting of both the set of offered flows and the set of available network paths. The worst-case running time complexity of the proposed heuristic was derived.

Our heuristic gives near-optimal results in terms of the number of light-trails used to carry the offered traffic. It has also been able to pack more 95% of the offered traffic within a number of light-trails that is very close to the optimal minimum. We have also applied our heuristic to the case of incremental problem where we were able to only make use of the dynamic path attributes when selecting the best path for the new flow. As expected, incremental traffic solutions were further from optimal comparing to the static traffic results.

Our future work involves porting the proposed heuristic to the survivable routing problem where each flow is assigned to two link-disjoint paths for backup purposes against link failures.

References

[1] Poikselka, M., Niemi, A., Khartabil, H., Mayer, G.: The IMS: IP Multimedia Concepts and Services. Wiley, New York (2006), ISBN 0-470-01906-9

[2] Gumaste, A.: Light-trails and light-frame architectures for optical Networks. Ph.D. thesis, University of Texas, Dallas, TX, USA (2003)

[3] Zhang, Z., Fu, J., Guo, D., Zhang, L.: Lightpath routing for intelligent optical networks. IEEE Netw. **15**(4), 28–35 (2001)

[4] Qiao, C., Yoo, M.: Optical Burst Switching (OBS)—A new paradigm for an optical Internet. J. High Speed Netw. (Special Issue on Optical Networks) **8**(1), 69–84 (1999)

[5] Mahony, M., Simeonidou, D., Hunter, D., Tzanakaki, A.: The application of optical packet switching in future communication networks. IEEE Commun. Mag. **39**(3), 128–135 (2001)

[6] Fang, J., He, W., Somani, A.K.: Optimal light-trail design in WDM optical networks. Proc. of IEEE ICC'04 (Paris, France, June 2004), vol. 3, pp. 1699–1703

[7] Makhorin, A.: GNU Linear Programming Kit (GLPK). Department for Applied Informatics, Moscow Aviation Institute, Moscow, Russia

[8] Ramaswami, R., Sivarajan, K.N.: Design of logical topologies for wavelength-routed optical networks. IEEE J. Select. Areas Commun. **14**(5), 840–851 (1996)

[9] Mukherjee, B., Banerjee, D., Ramamurthy, S., Mukhrejee, A.: Some principles for designing a wide-area optical network. Proc. of IEEE INFOCPM'94 (Toronto, Canada, June 1994), pp. 110–119

[10] Gumaste, A., Chlamtac, I.: Mesh implementation of light-trails: a solution to IP centric communication. Proc. of IEEE 12th International Conference on Communication (ICCCN) (Dallas, USA, October 2003), pp. 178–183

[11] Ye, Y., Woesner, H., Grasso, R., Chen, T., Chlamtac, I.: Traffic grooming in light trail networks. Proc. of IEEE GLOBECOM'05 (St. Louis, MO, USA, Nov–Dec 2005), vol. 4, pp. 1957–1962

[12] Gumaste, A., Palacharla, P., Kinoshita, S.: Light-trail based shared wavelength optical network. Proc. of 10th Opto-Electronics Communication Conference (OECC) (Seoul, Korea, July 2005)

[13] Balasubramanian, S., Somani, A.K.: Traffic grooming in statistically shared optical networks. Proc. of 31st IEEE conference on Local Computer Networks (LCN) (Tampa, FL, USA, Nov 2006), pp. 335–342

[14] Gumaste, A., Palacharla, P., Naito, T.: Experimental demonstration for static and dynamic control for light-trail networks. Proc. of 11th Opto-Electronic Communications Conference (OECC) (Kaohsiung, Taiwan, July 2006)

[15] Balasubramanian, S., Kamal, A.E., Somani, A.K.: Network design for IP-centric light-trail networks. Proc of the 2nd international conference on Broadband Networks (Broadnets) (Boston, MA, USA, Oct 2005), vol. 1, pp. 41–50

Authors Biographies

Amr S. Ayad received his B.Sc. (honors) and M.Sc. in electrical engineering from Cairo University in 1996 and 2001, respectively. He is currently working towards his Ph.D. in the faculty of computing, Cairo University. His research interest is in the area of design of all-optical network and in particular light-trails based networks.

Prof. Dr. Khaled M. F. Elsayed received his B.Sc. (honors) in electrical engineering and M.Sc. in engineering mathematics from Cairo University in 1987 and 1990, respectively. He received his Ph.D. in computer sciences and computer engineering from North Carolina State University in 1995. He is now a full Professor in Cairo University Egypt and a technical director at SysdSoft Inc. in Cairo. Between 1995 and 1997, he was a member of scientific staff with Nortel Wireless Systems Engineering in Richardson, TX.

Dr. Elsayed was the editor for the Internet technology series of the IEEE Communications Magazine from 1998 until 2002. He has served in technical programme committees for several IEEE, IFIP and ITC conferences. He was the technical co-chair for IFIP MWCN 2003 conference in Singapore. He also served as an expert evaluator for the European Commission FP5 and FP6 programmes. His research interest is in the area of performance evaluation of communications networks including IP, wireless and optical networks.

Prof. Dr. Sanaa El-Ola Hanafy Ahmed received her B.Sc. and M.Sc (honors) in electrical engineering from Cairo University. She received her Ph.D. in computer networks, Toulouse, France. For the period of 1998–2006, she has been the head of the information technology department at the school of computers and information at Cairo University. Her current research interests include optical networks, wireless networks and distributed systems. She has served on the editorial of the International Journal of Intelligent Computing and Information Sciences. She is currently the vice dean for educational and student affairs at the faculty of computers and information, Cairo University.