MET: an Efficient Static Routing Algorithm for WDM Networks with Full Wavelength Conversion

Bin Wu, *Student Member*, *IEEE*, and Kwan L. Yeung, *Senior Member*, *IEEE*

***Abstract*—Aiming at simultaneously minimizing the network wavelength requirement (NWR) and the network bandwidth requirement (NBR) for a given traffic matrix, we focus on the static routing problem in WDM networks with full wavelength conversion. A new algorithm MET (Most Even Traffic distribution) is proposed. MET consists of two steps, the initial shortest-path-based route assignment, followed by a rerouting process to refine the solution. Unlike existing approaches, MET makes each rerouting decision by judiciously balancing the overall network traffic loads. A uniformity function is defined and embodied in MET to characterize the traffic load distribution in the network. Numerical results show that MET outperforms the best existing algorithm (Min-hops [6]) by requiring less amount (about 10% saving) of network bandwidth (NBR) and less number of wavelengths (NWR).**

***Index Terms*—Network cost, routing and wavelength assignment (RWA), traffic engineering, wavelength conversion.**

I. Introduction

WAVELENGTH division multiplexing (WDM) allows multiple wavelength channels to concurrently share the same fiber for parallel transmissions. It is the most efficient way to fully exploit the fiber capacity while bypassing the speed bottleneck of electronic components. In WDM optical networks, the problem of *routing and wavelength assignment* (RWA) is widely studied [1-6]. In RWA, the path for carrying a call is determined by a routing algorithm, and the specific wavelength channel to be used at each hop along the found path is handled by a wavelength assignment algorithm. There are two major versions of RWA, dynamic and static. A *dynamic* RWA scheme (e.g. [3-4]) operates in real-time with the goal of minimizing the call blocking probability in the network. A lightpath is established for each call upon its arrival and is released upon its departure.

On the other hand, a *static* RWA scheme assumes that the traffic matrix is given. An important objective is to construct a set of lightpaths to carry the given traffic demands with minimum *network cost*. Network cost is *mainly* determined by the *network wavelength requirement* (NWR), which is defined as the maximum number of wavelengths to be supported on any fiber in the network. This is because packing more wavelengths onto a fiber requires more expensive optical components. Therefore, minimizing NWR can minimize the network cost. In addition, we define the *network bandwidth requirement* (NBR) as the total number of hops on all the constructed lightpaths. If NBR can be minimized, then the potential to accommodate future traffic growth can be maximized. This also helps to reduce the network cost.

In contrast to the dynamic RWA, static RWA usually operates offline and the network will be configured to run on a much longer time scale. In a network without wavelength converters, the static RWA problem is subject to the *wavelength continuity constraint* [3], i.e. the same wavelength (channel) must be used on every hop of a lightpath. In a network with *full* wavelength conversion, wavelength continuity constraint no longer exists. This effectively removes the need of wavelength assignment (algorithm). Unfortunately, even the routing problem for minimizing NWR (network wavelength requirement) alone is NP-hard [6]. Heuristic routing algorithms are proposed to solve this problem. The most efficient one is by Nagatsu *et al.* [5]. Their algorithm adopts a priority-based routing followed by a rerouting process to reduce NWR.

Since network cost is also affected by NBR (network bandwidth requirement), Min-hops algorithm [6] is proposed to minimize both NWR and NBR. Specifically, Min-hops initially assigns the traffic demands to lightpaths according to the shortest paths in the network, and then iteratively reroutes the lightpaths on the most heavily loaded links to reduce NWR. Each rerouting is designed such that the increase in NBR is always less than or equal to two hops [6]. It is shown that with Min-hops, less NBR than [5] can be achieved with almost the same value of NWR.

In this paper, we also focus on the static routing problem in WDM networks with full wavelength conversion. Aiming at minimizing both NWR and NBR, a new algorithm called MET (Most Even Traffic distribution) is proposed. MET follows the same two-step approach (initial route assignment followed by rerouting) as [5, 6]. The fundamental difference between our work and Min-hops [6] is that, MET makes each local rerouting decision based on a *global* view of load balancing, whereas only *local* optimization is used in Min-hops algorithm. Although load balancing is a well-known concept, we gracefully embody it into MET by defining a *traffic uniformity function* to characterize the traffic load distribution in the network.

This work is supported by Hong Kong Research Grant Council Earmarked Grant HKU 7032/01E.

Bin Wu and Kwan L. Yeung are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam, Hong Kong (Tel: 852-2857-8493; Fax: 852-2559-8738; e-mail: {binwu, kyeung}@eee.hku.hk).

0-7803-9569-7/06/$20.00 ©2006 IEEE

II. MET Algorithm

Without loss of generality, we focus on bidirectional WDM networks with symmetric fiber connections. The shortest paths between any two nodes (either $i \rightarrow j$ or $j \rightarrow i$) are the same. For a network with N nodes, the traffic demands are represented by an $N \times N$ integer traffic matrix $\boldsymbol{T}=\{t(i, j)\}$, where $t(i, j)$ denotes the number of lightpaths required to accommodate the traffic from node i to node j.

A. MET Overview

MET consists of two steps, initial route assignment and rerouting. In initial route assignment, shortest paths are assigned to carry the traffic demands between all the node pairs. This minimizes NBR (network bandwidth requirement) but not NWR (network wavelength requirement). The set of shortest paths can be determined by Dijkstra's or Floyd-Warshall algorithms [7]. If multiple shortest paths exist between nodes i and j, there are two approaches to assign the traffic demand $t(i, j)$: a) randomly select a shortest path and assign all $t(i, j)$ lightpaths to it, or b) evenly spread out the $t(i, j)$ lightpaths onto the multiple shortest paths in a round robin fashion. The latter approach is adopted by MET since it tends to maximize the effect of load balancing.

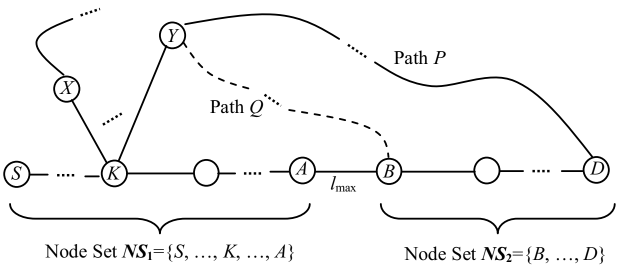
In the rerouting step, we iteratively reroute the lightpaths assigned in the first step to minimize NWR and the potential increase in NBR. Note that in a network with full wavelength conversion, NWR is determined by the number of lightpaths carried by the most heavily loaded link $l_{\max}$. MET first finds link $l_{\max}$. Then all the lightpaths passing through $l_{\max}$ are examined one by one for possible rerouting. The rerouting decision is based on the increase in NBR and the traffic uniformity function defined in Part B of this section. The detailed rerouting process is presented in Part C. When a lightpath is rerouted, the traffic load on $l_{\max}$ can be reduced by one (wavelength). This corresponds to a *possible* decrease of NWR by one. MET terminates when no new rerouting can be found to further reduce NWR.

B. Traffic Uniformity Function U(Φ)

To measure the traffic distribution in a network, we introduce a uniformity function $U(\Phi)$. Here, Φ represents the current state of lightpath assignment in the network (i.e. current *network state*). It changes after each rerouting. Let $w(\Phi, l)$ be the current load on link l, i.e. the number of lightpaths passing through link l under current network state Φ. Further let E be the total number of fiber links in the network. Then the average number of lightpaths per fiber (i.e. average load) can be calculated as $\sum_l w(\Phi, l)/E$. The traffic uniformity function $U(\Phi)$ is defined as the sum of the squared *load fluctuation* on individual links, or

$$U(\Phi)=\sum_l \left[w(\Phi,l)-\frac{1}{E}\sum_l w(\Phi,l) \right]^2 . \qquad (1)$$

From (1), we can see that the traffic uniformity function $U(\Phi)$ gives more weight to larger load fluctuations than smaller ones, and a smaller $U(\Phi)$ corresponds to a better balanced traffic distribution.



X
Y
Path P
Path Q
S
K
A
B
D
$l_{\max}$
Node Set $\boldsymbol{NS}_1$={S, ..., K, ..., A}
Node Set $\boldsymbol{NS}_2$={B, ..., D}

Fig. 1. Rerouting process.

C. Rerouting Process

Fig. 1 shows how the rerouting process works. Assume that the link between nodes A and B is currently the most heavily loaded link $l_{\max}$. Further assume a lightpath $S \rightarrow \ldots \rightarrow K \rightarrow \ldots \rightarrow A \rightarrow B \rightarrow \ldots \rightarrow D$ with source S and destination D is selected for possible rerouting. The nodes on the selected lightpath are separated by link $l_{\max}$ into two sets: $\boldsymbol{NS}_1=\{S, \ldots, K, \ldots, A\}$ and $\boldsymbol{NS}_2=\{B, \ldots, D\}$. To reroute this lightpath away from $l_{\max}$, we first pick up a node from $\boldsymbol{NS}_1$ and check if we can find a substitute path that bifurcates from it to bypass $l_{\max}$.

Suppose node K is selected, as shown in Fig. 1. We *tentatively* reroute $S \rightarrow \ldots \rightarrow K \rightarrow \ldots \rightarrow A \rightarrow B \rightarrow \ldots \rightarrow D$ to $S \rightarrow \ldots \rightarrow K \rightarrow Y \rightarrow$Path $P \rightarrow D$, where Y is a neighbor of K (i.e. Y and K are connected by a direct link) and Path P is a shortest path between Y and D (which bypasses $l_{\max}$). We then check whether the load on each link of $K \rightarrow Y \rightarrow$Path $P \rightarrow D$ is less than (or equal to) that on $l_{\max}$.

If yes, this new *tentative route* is added to a rerouting candidate set $\boldsymbol{C}$ as a *feasible* route. Let Δh denote the difference of the number of hops between the feasible route and the original route. We calculate both Δh and the uniformity value $U(\Phi)$, while assuming this feasible route will be finally chosen for rerouting.

Otherwise, another shortest path between Y and D is considered (if any). When all the shortest paths between Y and D are checked, we proceed to the next neighbor of K, say X, and repeat the same process above for all the shortest paths between X and D. Likewise, this process is performed for each and every node in $\boldsymbol{NS}_1$.

Next we consider the nodes in $\boldsymbol{NS}_2$ in a similar fashion, but this time we use the shortest paths between the source node S and the neighbors of the nodes in $\boldsymbol{NS}_2$.

For each lightpath passing through the most heavily loaded link $l_{\max}$, we carry out the above tentative rerouting process. Finally, we get the rerouting candidate set $\boldsymbol{C}$, which contains all the feasible routes that can reduce the load on $l_{\max}$. Since each feasible route is associated with two values Δh and $U(\Phi)$, the one that satisfies the following condition is selected for rerouting:

$$\min_C \left\{ U(\Phi) \right\} \Big|_{\min_C \{\Delta h\}} . \qquad (2)$$

The idea is to choose the route with the minimum $U(\Phi)$ among the candidates that have the same minimum Δh. This not only minimizes the increase of hop distance in the rerouted

MET Algorithm

Step 1: Initial route assignment:

Determine all-pair shortest paths in the network. Assign the shortest paths to the corresponding lightpaths in a round robin manner. Use a list *LINKS* to record the number of lightpaths passing through each link, and a list *LIGHTPATHS* to record all the lightpaths and their routes.

Step 2: Rerouting:

a) Construct the ROUTES list:

Find the most heavily loaded link $l_{\max}$ by checking the *LINKS* list. Scan the *LIGHTPATHS* list and copy all the lightpaths passing through $l_{\max}$ to a *ROUTES* list. Mark all the lightpaths in *ROUTES* as not-yet-considered by setting an indicator *LP_MARK*=0 for each lightpath. Initialize a rerouting candidate set $\boldsymbol{C}$ to null.

b) Pick up a lightpath for consideration:

Pick up any not-yet-considered lightpath (*LP_MARK*=0) from *ROUTES*. Let $S\to\ldots\to A\to B\to\ldots\to D$ represent this route, where S and D are the source and destination and A, B are the two end nodes of $l_{\max}$. Put all the nodes between S and A to a set $\boldsymbol{NS}_1$, and all the nodes between B and D to another set $\boldsymbol{NS}_2$. Mark all the nodes in $\boldsymbol{NS}_1$ and $\boldsymbol{NS}_2$ as not-yet-considered by setting an indicator *ND_MARK*=0 for each node.

c) Consider the nodes in NS₁:

For each not-yet-considered node K in $\boldsymbol{NS}_1$ (with *ND_MARK*=0), find all its neighbors (except the two on the current lightpath). Let Y represent a neighbor. Tentatively reroute the lightpath to $S\to\ldots\to K\to Y\to$Path $P\to D$, where Path P is a shortest path between nodes Y and D (all the shortest paths between Y and D are sequentially considered). Check if the load on each link of $K\to Y\to$Path $P\to D$ is less than or equal to that on $l_{\max}$. If yes, calculate the increase of hop distance Δh for the tentative route and the corresponding uniformity value $U(\Phi)$ defined in (1). Add this tentative route to $\boldsymbol{C}$. Otherwise, conduct the same tentative rerouting on another neighbor of K until all the neighbors are considered. Set the *ND_MARK* of node K to 1. Repeat this step until all the nodes in $\boldsymbol{NS}_1$ are considered (*ND_MARK*=1).

d) Consider the nodes in NS₂:

For each not-yet-considered node K^* in $\boldsymbol{NS}_2$ (with *ND_MARK*=0), find all its neighbors (except the two on the current lightpath). Let Y^* represent a neighbor. Tentatively reroute the lightpath to $S\to$Path $P^*\to Y^*\to K^*\to\ldots\to D$, where Path P^* is a shortest path between nodes S and Y^* (all the shortest paths between S and Y^* are sequentially considered). Check if the load on each link of $S\to$Path $P^*\to Y^*\to K^*$ is less than or equal to that on $l_{\max}$. If yes, calculate the increase of hop distance Δh^* for the tentative route and the corresponding uniformity value $U(\Phi^*)$ defined in (1). Add this tentative route to $\boldsymbol{C}$. Otherwise, conduct the same tentative rerouting on another neighbor of K^* until all the neighbors are considered. Set the *ND_MARK* of node K^* to 1. Repeat this step until all the nodes in $\boldsymbol{NS}_2$ are considered (*ND_MARK*=1).

e) Loop for lightpath analysis:

Set the *LP_MARK* of the selected lightpath to 1. Loop to *b)* until all the lightpaths in the *ROUTES* list are considered (*LP_MARK*=1).

f) Rerouting decision and network state update:

Scan the rerouting candidate set $\boldsymbol{C}$ and find all the tentative routes with minimum Δh. Among them, pick up the one with minimum $U(\Phi)$ for rerouting. Randomly pick up one if multiple choices are available. Update *LINKS* and *LIGHTPATHS* lists according to the rerouting result.

g) Loop for the next rerouting:

Loop to *a)* until no further rerouting can be made for all the most-heavily-loaded links. The final solution is in *LIGHTPATHS*.

lightpath, but also ensures that the traffic is evenly distributed in the network.

After each successful rerouting, MET updates the current network state Φ and enters the next rerouting iteration. MET terminates if no additional rerouting can be made to further reduce NWR.

D. Time Complexity

Consider a network with N nodes and E links. Assume that $W_{\max}$ is the initial NWR after Step 1 (initial route assignment). Let $h_{\max}$ be the hop distance of the longest lightpath, and $D_{\max}$ be the maximum number of neighbors a node can have. The time complexity of Step 1 is dominated by finding the all-pair shortest paths in the network, which is $O(N^3)$ with Floyd-Warshall algorithm [7]. In Step 2 of the algorithm, we use the shortest paths (determined in Step 1) between the neighbors and the end nodes of the selected lightpath to construct the tentative routes. Let $S_{\max}$ be the maximum number of shortest paths between any node pair. To check a selected lightpath for possible rerouting, $O(h_{\max}D_{\max}S_{\max})$ tentative routes are to be examined and $O(h^2_{\max}D_{\max}S_{\max})$ comparisons are needed (to compare the traffic loads on the links of the tentative route with that on $l_{\max}$). The tentative rerouting is performed over all the lightpaths passing through each of the most heavily loaded links (i.e. $l_{\max}$) in order to reduce NWR by one, resulting in a time complexity of at most $O(h^2_{\max}D_{\max}S_{\max}W_{\max}E)$. In total, the time complexity of Step 2 is $O(h^2_{\max}D_{\max}S_{\max}W^2_{\max}E)$ and the overall complexity of MET is $O(S_{\max}N^3+h^2_{\max}D_{\max}S_{\max}W^2_{\max}E)$. It is *roughly* $O(S_{\max})$ times higher than Min-hops.

Since MET is designed for solving static routing, e.g. network planning before lightpaths are actually provisioned, its time complexity is acceptable. In fact, for the NSFNET in Fig. 2 with the traffic matrix in Fig. 3, MET needs only 10 seconds to get the final solution on a standard Pentium-IV 2.2G PC. For the PEON (Pan-European Optical Network) in Fig. 4 with the traffic matrix in Fig. 5, about 16 minutes are required.

III. Performance Evaluation

To compare the performance of MET with Min-hops [6], we apply both algorithms to the NSFNET in Fig. 2 and the PEON in Fig. 4, with the traffic matrices shown in Fig. 3 (taken from [8]) and Fig. 5 (taken from [9]) respectively. The results returned by both algorithms are summarized in TABLE I, where NBR is measured by the total number of hops on all the lightpaths constructed, and the last column is the percentage saving in NBR by MET. From TABLE I, we can see that for both NSFNET and PEON, MET outperforms Min-hops by requiring less or same number of wavelengths (NWR), while providing 8% to 12% saving in bandwidth (NBR). [1]

We also randomly generate network topologies and traffic demands to compare the performance of MET and Min-hops. In our experiments, topologies and traffic demands are generated as follows:

- *Random topology construction.* We first treat the graph as a directed one, and each node in the network emanates 2 directed edges to any other nodes with the same probability. Then, each of the directed edges is replaced

[1] A careful study shows that the NBR values listed in TABLE I of [6] is incorrect. For example, there are 180 lightpaths required in the traffic matrix shown in Fig. 3. However, TABLE I in [6] gives an NBR value less than 180 for NSFNET (Case 2 in [6]), which is impossible because each lightpath should take at least one hop.

by a bidirectional link. This resulting topology is also checked to ensure that it is connected.

- *Traffic matrix generation*. We first randomly generate a traffic matrix $\boldsymbol{T}=\{t(i,j)\}$ with $t(i,j)$ uniformly distributed in the range of 0—7. Then we adjust $\boldsymbol{T}$ to characterize some possible "hot" paths in the network as follows. If $t(i,j)=6$, we reset $t(i,j)$ to 0; if $t(i,j)=7$, we replace it by another random number in the range of 0—15; otherwise $t(i,j)$ keeps unchanged.

We vary the network size from N=10 to 30 in a step increase of 5. For each network size, statistics from 100 random experiments are collected and averaged. The results are given in TABLE II, where the last column indicates the number of times (out of 100) that Min-hops outperforms MET. Here, we have applied a very strict metrics to count this number. That is, the solution returned by Min-hops is regarded as better than MET if either its NWR or NBR is less than that of MET. The small values in the last column of TABLE II indicate that Min-hops can seldom outperform MET. In our experiments, we found that when this happens, the solutions returned by the two algorithms usually have no big difference in both NWR and NBR. For larger network sizes (such as N=25 or 30 in TABLE II), the chance that Min-hops outperforms MET seems to be even smaller.

TABLE II also confirms that an average saving of about 10% in NBR can be achieved by MET. Comparing the average NWR of the two algorithms, we can see that as the network size increases, MET tends to save more wavelengths (NWR).

IV. Discussion

Like all the existing algorithms, MET is a heuristic and it

TABLE I
Performance Comparison of Min-hops and Met

Network	Min-hops		MET		% saving in NBR by MET
	NWR	NBR	NWR	NBR	
NSFNET	25	462	24	422	8.66%
PEON	67	1897	67	1669	12.02%

TABLE II
Results Averaged over 100 Random Experiments

N	Min-hops		MET		% saving in NBR by MET	Times that Min-hops outperforms MET
	NWR	NBR	NWR	NBR		
10	16.13	216.86	15.92	196.13	9.56%	2
15	28.34	614.96	27.54	551.26	10.36%	1
20	38.50	1177.22	37.58	1056.46	10.26%	1
25	51.67	2013.24	50.43	1817.90	9.70%	0
30	62.35	3111.14	60.44	2800.96	9.97%	0

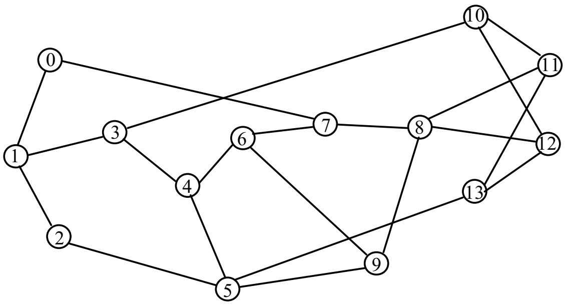


Fig. 2. NSFNET topology.

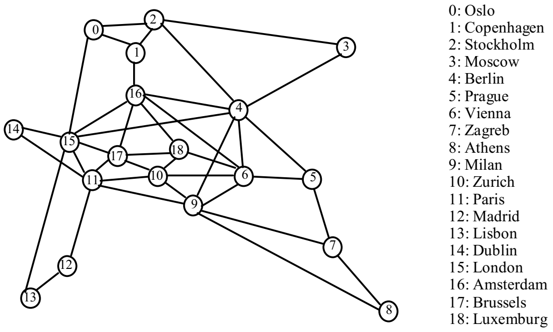


Fig. 4. PEON topology.

$$\boldsymbol{T}=\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
 & 1 & 0 & 7 & 1 & 2 & 6 & 2 & 4 & 0 & 4 & 0 & 3 \\
 & & 1 & 2 & 0 & 10 & 21 & 4 & 1 & 0 & 0 & 0 & 1 \\
 & & & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\
 & & & & 0 & 2 & 6 & 2 & 2 & 0 & 8 & 1 & 4 \\
 & & & & & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\
 & & & & & & 11 & 3 & 2 & 0 & 7 & 0 & 3 \\
 & & & & & & & 6 & 6 & 1 & 5 & 1 & 3 \\
 & & & & & & & & 4 & 1 & 13 & 2 & 0 \\
 & & & & & & & & & 1 & 2 & 1 & 0 \\
 & & & & & & & & & & 1 & 0 & 0 \\
 & & & & & & & & & & & 0 & 0 \\
 & & & & & & & & & & & & 1
\end{bmatrix}$$

Fig. 3. Traffic matrix $\boldsymbol{T}$ for the NSFNET topology.

$$\boldsymbol{T}=\begin{bmatrix}
1 & 3 & 1 & 5 & 2 & 1 & 1 & 6 & 1 & 16 & 7 & 1 & 3 & 1 & 2 & 9 & 15 & 1 \\
 & 1 & 4 & 1 & 6 & 1 & 1 & 1 & 10 & 12 & 7 & 1 & 1 & 1 & 9 & 1 & 11 & 5 \\
 & & 1 & 1 & 1 & 1 & 3 & 5 & 16 & 13 & 2 & 4 & 1 & 1 & 1 & 5 & 1 & 1 \\
 & & & 1 & 1 & 9 & 3 & 1 & 5 & 1 & 11 & 1 & 16 & 1 & 12 & 8 & 1 & 1 \\
 & & & & 1 & 4 & 1 & 1 & 4 & 4 & 16 & 1 & 14 & 2 & 4 & 1 & 1 & 1 \\
 & & & & & 9 & 1 & 3 & 1 & 4 & 1 & 10 & 1 & 5 & 1 & 16 & 1 & 1 \\
 & & & & & & 1 & 1 & 1 & 1 & 1 & 1 & 3 & 1 & 5 & 1 & 7 & 1 \\
 & & & & & & & 1 & 9 & 1 & 11 & 1 & 16 & 1 & 8 & 4 & 1 & 14 \\
 & & & & & & & & 1 & 2 & 1 & 5 & 1 & 7 & 1 & 3 & 1 & 6 \\
 & & & & & & & & & 4 & 4 & 1 & 2 & 1 & 16 & 13 & 11 & 1 \\
 & & & & & & & & & & 7 & 3 & 8 & 1 & 3 & 3 & 3 & 3 \\
 & & & & & & & & & & & 1 & 1 & 11 & 4 & 16 & 1 & 3 \\
 & & & & & & & & & & & & 1 & 3 & 1 & 1 & 7 & 9 \\
 & & & & & & & & & & & & & 4 & 5 & 1 & 1 & 1 \\
 & & & & & & & & & & & & & & 1 & 10 & 2 & 1 \\
 & & & & & & & & & & & & & & & 9 & 1 & 1 \\
 & & & & & & & & & & & & & & & & 3 & 1 \\
 & & & & & & & & & & & & & & & & & 1
\end{bmatrix}$$

Fig. 5. Traffic matrix $\boldsymbol{T}$ for the PEON topology.

does not guarantee optimal solutions. In MET, the rerouting decision is made by intelligently checking only a subset of the total solution space: those routes containing the shortest paths between the neighbors and the source or destination node of the selected lightpath.

MET outperforms Min-hops due to the following four reasons. First, in each rerouting of MET, only those candidates with minimum hop distance increase (Δh) are considered, whereas Min-hops does not ensure that the rerouted lightpath has minimum Δh. Secondly, among those candidates in $\boldsymbol{C}$, the one that makes the network traffic load most evenly distributed is chosen by MET; similar feature is missing from Min-hops. Thirdly, in each rerouting, MET directly reroutes from a neighbor to the source/destination using the shortest path; Min-hops only bypasses the most heavily loaded link (detailed in the following paragraphs). Fourthly, MET searches a proper rerouting in a solution space that is slightly larger than Min-hops. The first three reasons essentially differentiate MET from Min-hops by always taking a global view in making each rerouting decision. The fourth reason also explains why MET has a slightly higher time complexity than Min-hops.

Min-hops increases the hop distance of a lightpath by at most two in each rerouting. This is highlighted as an important feature/reason for the good performance of Min-hops (see formula (4) in [6]). However, our rerouting process in MET might be even better. In order to bypass the most heavily loaded link $l_{\max}$, Min-hops reroutes a lightpath from a neighbor of the selected lightpath to one of the end nodes of $l_{\max}$, but MET directly reroutes the lightpath to the source or destination of the lightpath. Consider the scenario shown in Fig. 1. Assume that both Paths P & Q are the shortest paths between the corresponding nodes. If lightpath $S\to\ldots\to K\to\ldots\to A\to B\to\ldots\to D$ is selected for rerouting, Min-hops reroutes it to $S\to\ldots\to K\to Y\to$Path $Q\to B\to\ldots\to D$, instead of $S\to\ldots\to K\to Y\to$Path $P\to D$ by MET. Let h_p represent the number of hops of an arbitrary path p, because

$$h_{Y\to\text{path}\quad Q\to B\to\cdots\to D} \geq h_{Y\to\text{path}\quad P\to D}\,, \tag{3}$$

we have

$$h_{S\to\cdots\to K\to Y\to\text{path}\quad Q\to B\to\cdots\to D} \geq h_{S\to\cdots\to K\to Y\to\text{path}\quad P\to D}\,, \tag{4}$$

As a result, Δh introduced by MET is no more than that introduced by Min-hops (assume that the current network state is the same for both).

Besides, if link $l_{\max}$ in the network carries the heaviest load, most likely other nearby links of $l_{\max}$ are also heavily loaded. If path $S\to\ldots\to K\to Y\to$Path $Q\to B\to\ldots\to D$ is used for rerouting (as by Min-hops), it may not be efficient to reduce the traffic load in the surrounding area of $l_{\max}$. In MET, Path P is adopted for rerouting, which directly goes to the destination of the lightpath as shown in Fig. 1. This route tends to bypass the "hot area", such that the traffic load in the network can be better balanced.

Although the network cost is mainly determined by NWR, in certain cases, we may hope to achieve a given NWR while judiciously save some bandwidth for future traffic growth. Then, NWR becomes a constraint rather than an objective parameter. In this case, two strategies may be taken: 1) still use MET to minimize the number of wavelengths required (which must be smaller than the given NWR) at the possible expense of a larger NBR; or 2) slightly modify MET to stop the rerouting when the constraint (i.e. the given NWR) is met. In the second strategy, the number of wavelengths required may not be minimized, but the solution requires a smaller NBR.

For a network without full wavelength conversion, MET can be combined with a wavelength assignment scheme (such as the one proposed in [6]) to reduce the number of wavelength converters required.

V. Conclusion

For WDM networks with full wavelength conversion, we proposed a new static routing algorithm MET (Most Even Traffic distribution) to simultaneously minimize the network wavelength requirement (NWR) and the network bandwidth requirement (NBR) for a given traffic matrix. MET consists of two steps, initial route assignment and rerouting. Initial route assignment is to construct lightpaths based on the shortest paths. In the rerouting step, we iteratively reroute the lightpaths to minimize NWR while keeping the increase in NBR minimum. Each rerouting decision is made by judiciously balancing the traffic load in the network based on a traffic uniformity function defined in the paper. Compared with the best-known Min-hops algorithm [6], MET achieves an average NBR saving of about10% with a smaller NWR.

References

[1] Y. Zhang, O. Yang and H. Liu, "A Lagrangean relaxation and subgradient framework for the routing and wavelength assignment problem in WDM networks", *Selected Areas in Communications*, *IEEE Journal on*, vol. 22, issue 9, pp. 1752-1765, Nov. 2004.

[2] J. Kuri, N. Puech, M. Gagnaire, E. Dotaro and R. Douville, "Routing and wavelength assignment of scheduled lightpath demands", *Selected Areas in Communications*, *IEEE Journal on*, vol. 21, issue 8, pp. 1231-1240, Oct. 2003.

[3] G. Mohan and C.S.R. Murthy, "A time optimal wavelength rerouting algorithm for dynamic traffic in WDM networks", *Lightwave Technology*, *Journal of*, vol. 17, issue 3, pp. 406-417, Mar. 1999.

[4] Chien Chen and S. Banerjee, "A new model for optimal routing and wavelength assignment in wavelength division multiplexed optical networks", *INFOCOM '96*, *Proceedings IEEE*, vol. 1, pp. 164-171, Mar. 1996.

[5] N. Nagatsu, Y. Hamazumi and K. I. Sato, "Number of wavelengths required for constructing large-scale optical path networks", *Electron. Commun. Jpn.*, *Part I-Commun.*, vol. 78, no. 9, 1995.

[6] H. V. Madhyastha and N. Balakrishnan, "An efficient algorithm for virtual-wavelength-path routing minimizing average number of hops", *Selected Areas in Communications*, *IEEE Journal on*, vol. 21, issue 9, pp. 1433-1440, Nov. 2003.

[7] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Upper Saddle River, NJ: Prentice-Hall, 1993.

[8] Y. Miyao and H. Saito, "Optimal design and evaluation of survivable WDM transport networks", *Selected Areas in Communications*, *IEEE Journal on*, vol. 16, issue 7, pp. 1190-1198, Sept. 1998.

[9] A. Fumagalli, I. Cerutti, M. Tacca, F. Masetti, R. Jagannathan, and S. Alagar, "Survivable networks based on optimal routing and WDM self-healing rings", *Proc. IEEE INFOCOM'99*, vol. 2, pp.726-733, Mar. 1999.