# A Lightpath Rerouting Algorithm for WDM Circuit-Switched Networks

N.S.C. Correia*, A. L. Barradas+, M.C.R. Medeiros*, J.J. O'Reilly+

*University of Algarve +University College London

**Abstract:** *In WDM circuit-switched networks rerouting can be used to improve throughput and reduce blocking probability. Here we present a novel rerouting algorithm obtained by representing the network as a directed graph. Our approach allows for a more realistic representation of the network than previously reported algorithms based on graph theory. Specifically it includes the representation of the two unidirectional optical fibres present in a real physical link.*

## 1. Introduction

The large bandwidth of optical fibres and the use of Wavelength Division Multiplexing (WDM) allowing the aggregation of many channels into a single fibre have made fibres attractive for high-speed networks. Although rerouting techniques have mostly been applied to the restoration of lightpaths when failures occur [1,2], it can also be used to improve throughput and reduce blocking probability. Previously reported rerouting algorithms for WDM networks [3,4] based on graph theory consider lightpaths, links and channels all bidirectional. In [3] a two stage algorithm is presented. The controller first tries to find a route without rerouting any existing circuit. If this fails, it tries to route the new circuit after rerouting some existing circuits. In [4] a time optimal algorithm is presented which differs from [3] in three aspects: (i) there are no separate phases for routing and rerouting (weights of edges are such that the algorithm prefers routes requiring no rerouting), (ii) it does not check if a lightpath is retunable whenever routing fails (dynamically updates retunability status of lightpaths) and (iii) it does not construct an auxiliary graph. However, neither approach applies to the unidirectional case. Here we present an algorithm that allows a more realistic representation of the network. Our approach includes the representation of the two unidirectional optical fibres present in a real physical link. We also compare the quality of lightpath routes for the rerouting and no rerouting cases.

## 2. System Configuration

A $G(N,L)$ network is considered where $N$ is the set of routing nodes and $L$ is the set of links. Each physical link is assumed to have 2 unidirectional optical fibres for transmission in both directions. The bandwidth of each optical fibre is wavelength-division demultiplexed into $W$ wavelengths, $\Lambda=\{\lambda_1, \lambda_2, \ldots, \lambda_W\}$. A routing node with $I$ input and $O$ output links, operating $W$ wavelengths, comprises $I$ demultiplexers, $O$ multiplexers and $W$ switches, so that each node is capable of connecting inbound wavelengths to any outbound link. It is assumed that nodes have no wavelength converters, i.e. wavelength continuity constraint is considered in a lightpath. This means that requests may be rejected even if bandwidth is available because of the nonavailability of the same wavelength at all fibre links leading to higher blocking probabilities. This situation can be avoided if rerouting is used where a few existing lightpaths are moved to other wavelengths in order to create a lightpath to satisfy the new request. Connection requests arrive randomly at each node as a Poisson process and hold for an exponentially distributed time. A controller node with network information makes routing decisions and sends control information to nodes to set up the circuit and reroute any circuit if necessary. New requests will not be blocked due to lack of transmitters/receivers.

## 3. Routing Algorithm

### 3.1. Definitions

The definitions follow the notation used in [3].

*Subgraphs*: The network $G(N,L)$ with wavelength set $\Lambda$ give rise to a set of disjoint subgraphs $G(N^\lambda,L^\lambda)$, $\lambda \in \Lambda$, one subgraph for each wavelength, where:

$$N^\lambda=\{i^\lambda: i\in N\},\ \lambda\in \Lambda \qquad [3.1]$$

$$L^\lambda=\{(i^\lambda,j^\lambda): (i,j)\in L\},\ \lambda\in \Lambda \qquad [3.2]$$

*Lightpath retunable cost*: The cost of retuning a lightpath $p$ is given by

$$rtc(p^\lambda) = \begin{cases} h, & \text{if retunable} \\ \infty, & \text{if nonretunable} \end{cases} \qquad [3.3]$$

where $h$ is the number of edges used by the lightpath ($h$+1 nodes need reconfiguration).

*Edge cost*: The cost of an edge connecting nodes $i$ and $j$ is given by

$$c(i^\lambda, j^\lambda) = \begin{cases} \varepsilon, & \text{if idle} \\ \infty, & \text{if busy} \end{cases} \qquad [3.4]$$

where $\varepsilon$ is a tiny value so that the cost of the longest possible path with free edges is less than 1, the smallest possible retuning cost.

*Path cost*: The cost of a route $p$ connecting nodes $u$ and $v$ is the sum of the retunable cost of lightpaths that have an edge in $p$, and $\varepsilon$ times the number of free edges on the path.

### 3.2 Rerouting of Lightpaths

To determine in constant time if a lightpath is retunable, it is necessary to dynamically update retunability information on every lightpath creation and release. For every lightpath $p$ in wavelength $i$ the following information is needed: number of edges the lightpath uses ($e_p$), how many of its edges are free in other subgraphs ($\boldsymbol{v}_p$, vector of size $W$-1) and the number of wavelengths it can be retuned to ($r_p$). If a connection $p$ is being created/released in wavelength $i$, besides filling/deleting this information for the new connection, information on every lightpath (in another wavelength) that shares some edges with $p$ must be updated. When creating $p$, for every edge that $p$ shares with some other lightpath $q$,

1) If $\boldsymbol{v}^i_q = e_q$ then $r_q = r_q - 1$ [3.5]
2) $\boldsymbol{v}^i_q = \boldsymbol{v}^i_q - 1$ [3.6]

When releasing $p$, for every edge that $p$ shares with some other lightpath $q$,

1) $\boldsymbol{v}^i_q = \boldsymbol{v}^i_q + 1$ [3.7]
2) If $\boldsymbol{v}^i_q = e_q$ then $r_q = r_q - 1$ [3.8]

### 3.3 Algorithm Description

The algorithm finds a minimum weighted path in each $W$ subgraphs, one subgraph per wavelength, and then chooses the best one. Initially, the source node $s$ is marked as the "*min*" node with distance value 0. When a node is marked and chosen as "*min*" node, it updates the distance value of its unmarked neighbors connected by free edges using the value $\varepsilon$. If the edge connecting "*min*" node to an unmarked neighbour is already used by a lightpath and that lightpath is retunable, then all edges used by the lightpath that are ahead of "*min*" (we are using a directed graph) and were not considered yet should be updated using the retunable cost. The node with the minimum distance value is chosen as "*min*" node and marked. The algorithm ends when destination node $d$ is chosen as the "*min*" node. After choosing the best route configuration information is sent to all involved routing nodes. The following procedure finds the route from $s$ to $d$ in a subgraph of some wavelength.

```
FOR every node i in G
    Distance(i)=∞; Marked(i)=FALSE; Parent(i)=0; ConnectionId(i)=0;
min=s; Distance(min)=0;
DO
  FOR every unmarked adjacent node y of min
    IF (edge(min,y) is free THEN
      IF Distance(min)+ c(min,y) < Distance(y) THEN
        Distance(y)=Distance(min)+ c(min,y)
        Parent(y)=min
        ConnectionId(y)=0
    ELSE
      p = connection label of edge(min,y)
      IF p is retunable THEN
        FOR every not analysed unmarked edge(i,j) of p ahead of min
          Distance(j)=Distance(i)+ rtc(p)
          Parent(j)=min
          ConnectionId(j)= p
          analysed_edges(i,j) = TRUE
  min = node with smallest label
  Marked(min)=TRUE
WHILE (Distance(min)!= ∞ AND min!=d)
```

## 4. Simulation Modules

To study the performance of the algorithm C++ in combination with a discrete event simulator called OMNet++ was used. The figures show the internal structure of the nodes.

*Switch*: Has information about how wavelengths must be directed from incoming fibres (or node interface) to outgoing fibres. Receives configuration and rerouting information for its wavelength;
*Gen*: Generates connection requests;
*Sink*: Receives traffic directed to the node;
*Node interface*: Retains arriving connection request in queue and asks for a wavelength and route to controller node. When answer arrives the request in queue is sent to the appropriate switch. Receives configuration and rerouting information from controller node and sends it to the appropriate switch.
*Ctrl node interface*: Responsible for making statistics.
*Shortest Path*: Finds a route and wavelength for a connection request using the routing algorithm. Sends configuration and rerouting information to nodes involved through ctrl node interface.

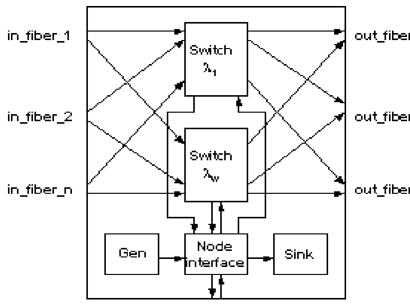


**Fig. 1** – Routing node.

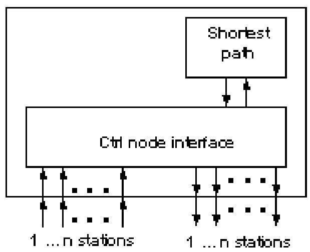


**Fig. 2** – Controller node.

## 5. Performance Analysis

In this section we illustrate the performance of our algorithm when applied to the 19-node and 39-link European Optical Network (EON) [5]. We also use a version of this EON with 24 links only, considering a higher number of wavelengths/link. All simulations produced 1000 messages/node. Blocking probabilities versus arrival rate per node is shown next.

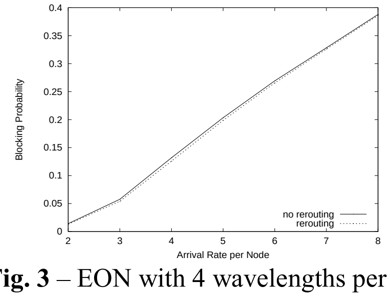


**Fig. 3** – EON with 4 wavelengths per fibre.

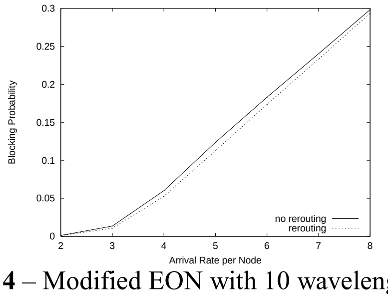


**Fig. 4** – Modified EON with 10 wavelengths per fibre.

The improvement in blocking probability when rerouting is allowed is in general small. This is due to the fact that the algorithm is only considering moving to a wavelength vacant in all edges (one step rerouting). For the modified EON a slightly better improvement was achieved because 10 wavelengths per fibre instead of 4 were considered, which increases the possibility of rerouting. This is confirmed by fig. 5 that shows the percentage of lightpaths requiring rerouting. The higher average number of hops of rerouted lightpaths for the modified EON shown in fig. 6 is also related to the higher number of wavelengths available since a higher number of wavelength allows the rerouting of longer lightpaths.

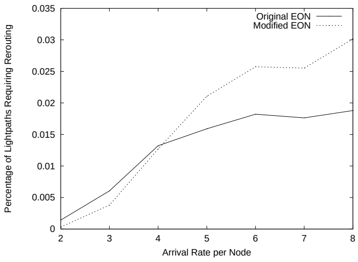


**Fig. 5** – Percentage of lightpaths requiring rerouting.

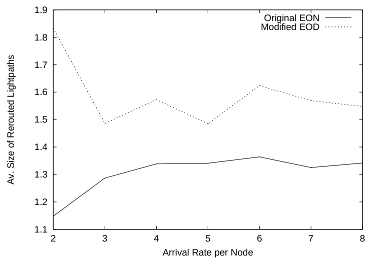


**Fig. 6** – Average size of rerouted lightpaths.

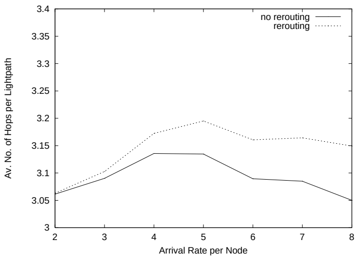


**Fig. 7** – Average size of lightpaths.

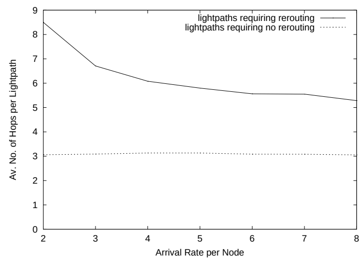


**Fig. 8** – Average size of lightpaths(rerouting allowed).

Fig. 7 shows for the modified EON that the average number of hops of lightpaths is higher when rerouting is allowed. This is because lightpaths requiring rerouting have a higher number of hops then lightpaths requiring no rerouting, as shown in fig. 8 for the allowed rerouting case. That is, while the algorithm reduces blocking probability this leads to a higher number of hops (higher delay) when compared to the no rerouting case.

## 6. Summary

We have presented a rerouting algorithm obtained by representing the network as a directed graph. Preliminary performance assessment of the algorithm was illustrated using the European Optical Network and a modified version of the same network. As ongoing work we are seeking further to develop this algorithm in order to consider the rerouting of interdependent lightpaths. The number of hops of lightpaths requiring rerouting needs also to be reduced.

## Acknowledgments

This work was supported by the Praxis project XXI/2/2.1/TIT/1573/95.

## References

[1] Olivier Crochat and Jean-Yves Le Boudee, "Design Protection for WDM optical Networks", IEEE J. Select. Areas in Communications, Vol.16, Sep.1998

[2] Bart Van Caenegem, Wim Van Parys, Filip De Turek and Piet M. Demeester, "Dimensioning of Survivable WDM Networks", IEEE J. Select. Areas in Communications, Vol.16, Sep.1998

[3] Kuo-Chun Lee and Victor O. K. Li, "A Wavelength Rerouting Algorithm in Wide-Area All-Optical networks", IEEE J. Lightwave Technology, Vol.14, Jun.1996

[4] G. Mohan and C. Siva Ram Murthy, "A Time optimal Wavelength Rerouting Algorithm for Dynamic Traffic in WDM Networks", IEEE J. Lightwave Technology, Vol.17, Mar.1999

[5] Nico Wauters and Piet Demeester, "Design of the Optical Path Layer in Multiwavelength Cross-Connected Networks", IEEE J. Select. Areas in Communications, Vol. 14, Jun. 1996