EIDGENÖSSISCHE TECHNISCHE HOCHSCHULE LAUSANNE
POLITECNICO FEDERALE DI LOSANNA
SWISS FEDERAL INSTITUTE OF TECHNOLOGY LAUSANNE

LABORATOIRE DE RÉSEAUX DE COMMUNICATION
Département d'Informatique

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Design of a Survivable WDM Photonic Network *

J. Armitage — BT Laboratories, UK - Suffolk IP5 7RE

O. Crochat, J.-Y. Le Boudec — LRC - EPFL, CH - 1015 Lausanne

Technical Report No 96/198

June 1996

Abstract

We consider schemes for protecting a network using a Wavelength Division Multiplexing (WDM) infrastructure against component or link failures. First, we explain how protection can be achieved by hardware redundancy. Then, we consider that with WDM networks, the failure of a single link or component may cause the simultaneous failure of several optical channels, potentially making impossible the restoration by rerouting in higher layers (SDH, ATM, IP). To address this, we introduce the concept of *Design Protection*, which aims at making such failure propagations impossible. We present the Disjoint Alternate Path (DAP) algorithm which places optical channels in order to maximise design protection. We show the result on the example of the ARPA-2 network.

Keywords: WDM, photonic, resilience, routing, tabu search.

To obtain a copy of this report, please contact Mrs Y. Dubuis, secretary, tel. +41 21 693 52 02

LRC
IN-Ecublens
1015 Lausanne - Switzerland

crochat@di.epfl.ch
tel. +41 21 693 6601

Contents

Design of a Survivable WDM Photonic Network*

J. Armitage — BT Laboratories, Martlesham Heath - Ipswich, UK - Suffolk IP5 7RE, Tel: +44 1473 645364, Fax: +44 1473 644171, *jon.armitage@bt-sys.bt.co.uk*

O. Crochat† — Laboratoire de Reseaux de Communication (LRC), EPFL, CH - 1015 Lausanne, Tel: +41 21 693 6601, Fax: +41 21 693 6610, *crochat@lrc.epfl.ch*

J.-Y. Le Boudec — *leboudec@di.epfl.ch*

Abstract

We consider schemes for protecting a network using a Wavelength Division Multiplexing (WDM) infrastructure against component or link failures. First, we explain how protection can be achieved by hardware redundancy. Then, we consider that with WDM networks, the failure of a single link or component may cause the simultaneous failure of several optical channels, potentially making impossible the restoration by rerouting in higher layers (SDH, ATM, IP). To address this, we introduce the concept of *Design Protection*, which aims at making such failure propagations impossible. We present the Disjoint Alternate Path (DAP) algorithm which places optical channels in order to maximise design protection. We show the result on the example of the ARPA-2 network.

Keywords: WDM, photonic, resilience, routing, tabu search.

1 Introduction

The large bandwidth of optical fibres have made them attractive for high-speed networks. The use of Wavelength Division Multiplexing (WDM) allows aggregation of many channels onto a single fibre [1, 2, 3, 4, 5]. It is the simplicity of this multiplexing scheme which permits a relatively easy access to the fibre's bandwidth, which is the underlying reason why WDM is about to play a major role in the expansion of photonic networks.

But the existence of many independent data channels over the fibre infrastructure could lead to problems in case of failure, as the amount of bandwidth lost by a component or cable failure is now much larger than what would have been lost in a traditional network. This is why an extra effort has to be spent

*This work is supported in part by the ACTS project COBNET and the Swiss government (OFES) and represents the view of the authors

†Corresponding author

in analyzing optical WDM networks, and finding ways to protect them against failures.

1.1 Definitions

We consider a backbone WDM photonic network consisting of *optical cross-connect nodes* (OXNs, or simply *nodes*) interconnected by optical bidirectional links. Each link actually consists of a pair of unidirectional fibres. We assume that a node is capable of routing any incoming wavelength channel on any of the incoming fibres to any wavelength channel on any of the outgoing fibres. The node may be performing all cross-connecting functions purely in the optical domain, or, like in the COBNET project [6], use a hybrid electrical-optical system (we do not make any distinction between these cases in this paper).

- A *Network Port* is a point on a node which gives access to the photonic network. Connected to a network port, one might have an SDH cross-connect, an ATM node, or (for example) an IP router (called *higher layer* switches). In the last two cases, the ATM or IP layer uses a physical channel such as SDH or Fibre Channel to carry the cells or packets.

- An *Optical Link* is a bidirectionnal physical connection between two nodes.

- A *Clear-Channel* (CC) is a bidirectionnal communication channel between two ports (like $c6$ between ports $B2$ and $C3$ in Figure 1). The *Route* of a clear-channel is the list of optical links used by the clear-channel (in Figure 1, the route of clear-channel $c6$ is $(l4, l5)$.

- A *Path* between two ports is a concatenation of clear-channels that are interconnected in the higher layer (SDH, ATM or IP). Figure 1 shows how paths, clear-channels and optical links are related. A path can be *Single-hop*, if only one clear-channel is used (path $p1 = (c6)$ between B and C in Figure 1) or *Multihop*, if more than one is used (for example path $p2 = (c4, c5, c3)$ between B and C).

Note that, with these definitions, hops refer to higher layer switching; a single-hop path will usually traverse more than one OXN. A multi-hop path is not a clear-channel, and is protocol-specific (as all the clear-channels used must conform to the same protocol).

In the following, we define two levels of network topology and two types of network protection.

The *Virtual Topology* consists of a graph representing all the clear-channels that are present in the network. It is the only view of the network available to the higher layer switches. The *Physical Topology* is the real network, composed of optical links and photonic nodes. The mapping between these two topologies (i.e. the assignment of the clear-channels within this network) is performed by the *design algorithm.*

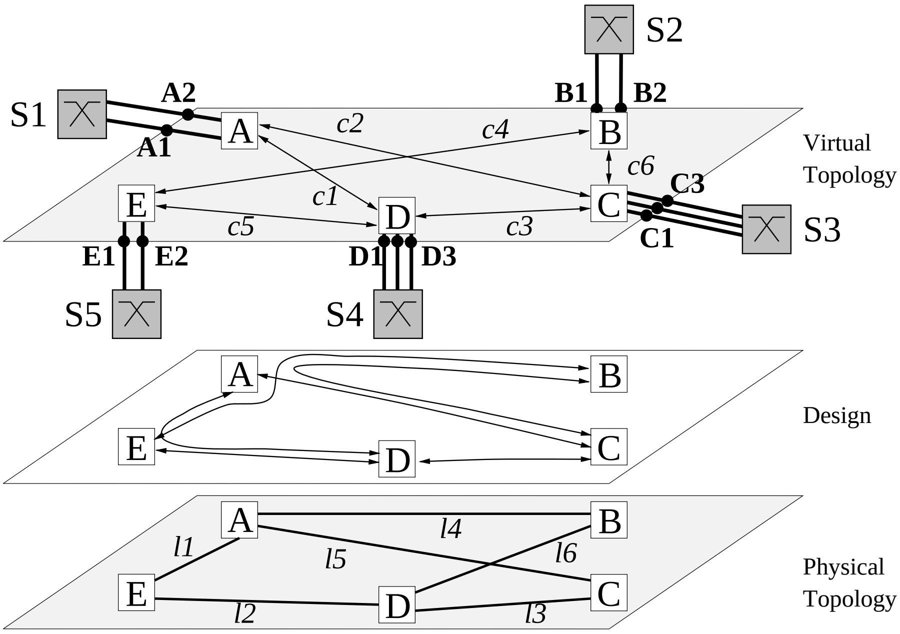

Figure 1: The two levels and a design example. Clear-channel $c6$ uses optical links $l4$ and $l5$. There is a single-hop path $p1 = (c6)$ between B and C, and (for example) an alternate, multi-hop path $p2 = (c4, c5, c3)$

1.2 Protection Mechanisms

The *Physical Protection* has the objective of ensuring that a bidirectional communication channel between two network ports will remain operational in the event of a component or an optical link failure. This protection is performed at the physical level. It can react very rapidly to such failure, but requires hardware redundancy as is explained in section 2. In general, physical protection also includes the ability to re-route clear-channels after a failure, transparently to the higher level switches, although we do not consider such methods within this paper.

There is a price to pay for Physical Protection, namely that of optical hardware duplication. In many situations it may not be cost-effective to provide such redundancy, and thus in real networks, we may only find physical protection implemented on business-critical routes. For other routes in such networks, clear-channel failures will occur. This will usually not result in dramatic losses of service because the higher layers (for example IP routers) have their own protection mechanisms. However, this may not always be true. Consider again Figure 1. The virtual topology seen by the higher layer seems to offer adequate redundancy between the higher layer switches (or routers). Assuming that clear-channel $c6$ fails, the higher layer will be able, thanks to its own routing algorithm, to reroute traffic between B and C through the alternate path $p2 = (c4, c5, c3)$. In reality however, clear-channels $c6$ and $c4$ share a common optical link $l4$; if $l4$ fails, there is a simultaneous failure of $c4$ and $c6$ in the virtual network.

The higher layers have a general view of the network where clear-channels are independent. Their own protection mechanisms (like rerouting algorithms or spare capacity allocation on the clear-channels) are based on this independence assumption. The previous example shows that this is not always true if WDM is used in the lowest layer. The purpose of *Design Protection* is to avoid such situations, by a proper placement of clear-channels onto optical links. In other words, the objective is to produce virtual networks for which the assumption of clear-channels independence is true. Of course it is still necessary for the higher layers to plan spare capacities inside the clear-channels in order to make service restoration possible.

Section 2 describes the components of a realistic photonic network, explains which kind of failures can occur and how full protection can be implemented by means of hardware redundancy. Section 3 introduces the concept of design protection, and presents the Disjoint Alternate Path (DAP) algorithm, an algorithm using tabu search for placing clear-channels in order to maximize design protection. In this version, the DAP algorithm does not consider limits on the number of clear-channels that can be supported per optical link. However, we show on the example of the ARPA-2 network that design protection does not necessarily require more wavelengths per fibre than the standard shortest-path algorithm. Extension of the DAP algorithm to incorporate constraints on the number of wavelengths per fibre is the object of ongoing research.

Section 4 concludes the article with a discussion of ongoing research.

2 Physical Protection

In this section, we discuss, in general terms, the problems of protecting an optical cross-connect network against both failures in the fibre cable linking together the optical nodes and also in the failure of the optical components and subsystems within the photonic network nodes.

For the purposes of the discussion we will consider a rather general WDM network architecture consisting of a number optical cross-connect nodes interconnected via a mesh of fibre pairs (see Figure 2). Each fibre in this photonic network is carrying multiple independent WDM channels. At each node in the network, there may be pairs of ports to and from the photonic network.

Many papers have been published on such meshed networks of optical cross-connect nodes, with many different technological solutions proposed for the cross-connect function [7, 8, 9, 10]. To be able to cross-connect between any incoming wavelength channel and any outgoing wavelength channel requires wavelength conversion to take place within the optical node. At present there are no commercially available components to perform this function, although

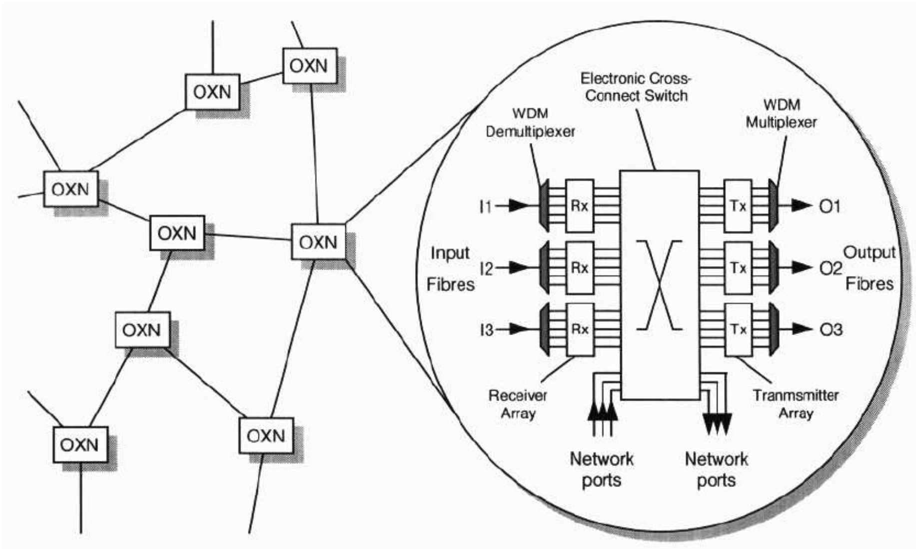

Figure 2: A generalised WDM cross-connect network. Each bidirectionnal optical link between two OXN consists of a pair of unidirectionnal fibres and each node can route any channel of any incoming fibre to any channel of any outgoing one. Communications using this optical network can be set by a connection to a pair of ports (In/Out).

there is much research progressing around the world towards the necessary components (such as tunable sources/receivers, wavelength translation devices, etc.) to construct an all-optical cross-connect node. One possible implementation for the cross-connect node using existing technologies is shown schematically in Figure 2. The M incoming wavelength channels on each of the N input fibres are optically demultiplexed and are converted to electrical data signals. These electrical data channels are then cross-connected electrically to the M outgoing wavelengths on the N output fibres or are dropped down to the local ports. The use of an optoelectronic cross-connect node, rather than a all-optical node, should not be viewed as a second-rate option. Indeed, with the use of variable bitrate retiming functions within an opto-electronic node, such a node would still provide all the network functionality as an all-optical node, and in addition would allow pulse retiming and reshaping, thereby overcoming link dispersion and WDM crosstalk problems.

This optical cross-connect node could just provide a static, but software reconfigurable, connectivity between the input and output WDM channels and the local add/drop channels to the ports, and this is the cross-connect function that is the focus for this paper. The ability to reconfigure the optical network under software control is for future work.

The optical cross-connect node as shown in Figure 2 is susceptible to component failures. Also without specific precautions the generalised network shown in the same figure is susceptible to breakages in the fibre cables linking the nodes. Protection of the network against both these types of eventuality can be made by duplications of optical hardware. Figure 3 shows a possible implementation for this network which allows for single component failures and optical link failures, and yet always maintains a connection between the two ports of a clear-channel.

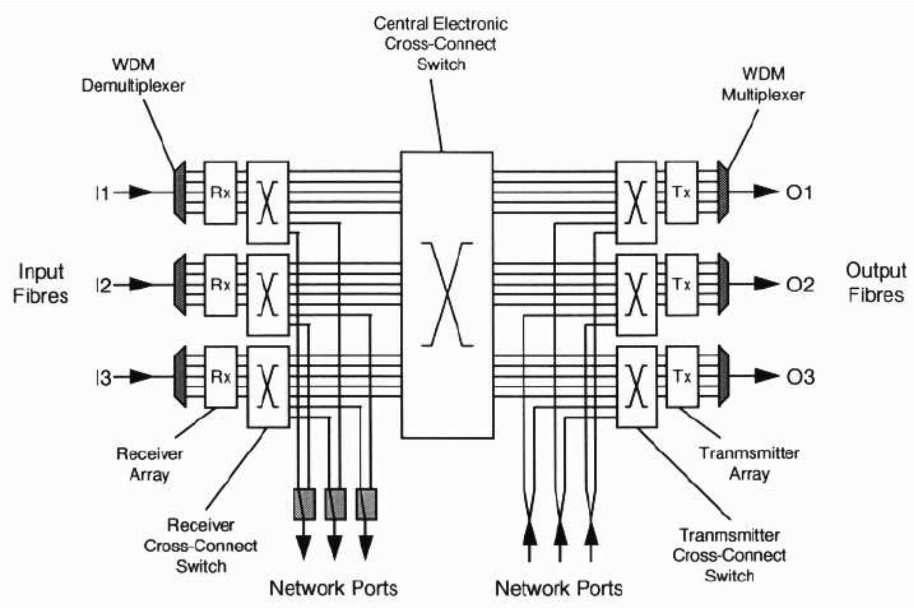

Figure 3: A possible implementation of a photonic node protected against component or link failures. The data from a port is sent over the network on two separate fibres, and at the receiving port a 1x2 switch chooses one of the two possible reception channels. Any single failure (link or component) only causes the failure of one of the two clear-channels between this pair of network ports.

For a network implemented with the nodes shown in Figure 3, the aim is that there are always two different physical routes between a pair of ports. At the input port, the electrical data delivered to the photonic network from the higher network layer is sent out onto the photonic network via two different transmitters which are located on two separate transmitter cards. These transmitters launch the data onto two different fibres leaving the node on separate physical routes. Resilient data channels are thus provided at the Transmit Node. At the receiving port, the two independent data channels arrive at the optical node on two separate fibres and pass through two separate receiver boards. The local manager then chooses to use the data from one of these two paths. Again there exist dual data paths within the Receive Node. For data passing through an intermediate node in the path, the Central Electronic Cross-Connect Switch does represent a single point of failure. This is not a problem as long as only one cross-connect switch in the whole network fails at any one time.

The specific details of the hardware implementation are not particularly important for the purposes of this paper. What should be noted is that if the network designer is worried about component or link failures and wishes to provide protection at the physical layer, the design of the photonic cross-connect network would have to include duplicated optical hardware, and would suffer the consequent cost implications. Nevertheless with the decreasing cost of optical technologies with their increasing worldwide usage, we may expect significant decrease in the cost of photonic nodes. Positive features of this protection strategy are its simplicity, and the speed at which the backup data stream can be attached to the network port. If any link or subsystem in this network were to fail, it is solely the responsibility of the receiver at the terminating port to make a decision about which of the two possible clear-channels it should utilise to extract the real data. This decision can be made locally – the local management function does not need to know about the global state of the network. Of course, the speed at which the higher level switch can then make use of the traffic from this backup data stream is dependent of the transport protocols used by those higher level switches.

3 Design Protection

The cost of physical protection and the guarantee of connectivity between network ports may be prohibitively high for its implementation in many situations. In this section, we consider a weaker form of network protection, called Design Protection, which aims at avoiding simultaneous failures in the virtual network.

3.1 The Principle

We have explained in Section 1.2 how hidden dependences between clear-channels could lead to the impossibility for the higher layer to reroute around a failure, in spite of apparent redundancy in the virtual network. The aim of *Design Protection* is to find, from a given virtual connectivity, a route for each of the clear-channels in the photonic network such that a single optical link failure leaves the virtual network connected. For example, a placement of clear-channels that satisfies this requirement is shown in Figure 4.

The effects of correlated failures of many clear-channels sharing each physical link can be eliminated – or at least minimized – by using the Disjoint Alternate Path (DAP) algorithm that we develop in this section. The DAP algorithm maps the clear-channels onto the physical network in such a way that, for each of them, there exists an alternate path with same end-nodes, but sharing no optical link with the clear-channel to which it is associated.

The algorithm is explained in this section, and simulations on the ARPA2 network shows the strong enhancement achieved in protection against the effects

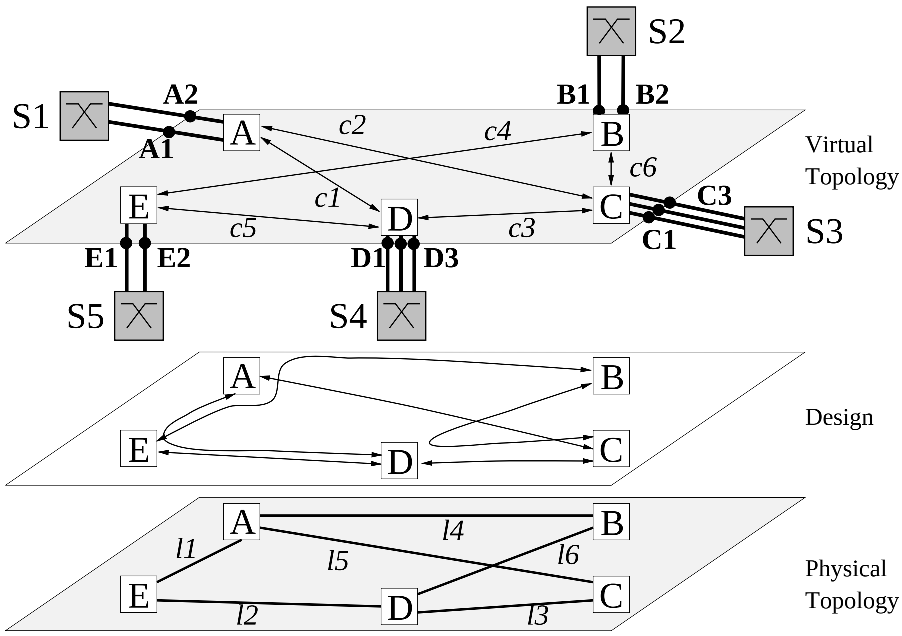

Figure 4: Same physical and virtual topologies as in Figure 1 but with a slightly different design (the route of $c6$ is now $(l6, l3)$), allowing survivability to any one-link failure in the physical topology

of optical link failures by the DAP algorithm compared to the one achieved by simple design strategies, such as a Shortest-Path Routing Algorithm.

Design Protection is performed by the configuration management at the setup time, through the design algorithm which maps the clear-channels described by the virtual topology onto the physical topology. It should be noted that this design protection is a static protection i.e. we are not considering a dynamic reconfiguration of the photonic nodes in the network following a node or link failure.

One should notice that this discussion about resilience against optical link failures can easily be adapted to node failures. The difference is that the routes of a clear-channel and its associated alternate path must be *node*-disjoint (instead of link-disjoint for link failure protection). But, in the case of node failures, it is not possible to protect all the clear-channels that had a port in the broken node, as all the end-users connected to this node are now unreachable. The only achievable protection is against failure propagation for the remaining clear-channels.

3.2 Formulation of the problem

The physical topology is represented by an undirected graph, G_p, where each node corresponds to a node in the network, and each edge to an optical link. The virtual topology is represented by an undirected graph, G_v, where each node corresponds to a node in the network, and each edge to a clear-channel.

3.2.1 Given

- Let N be the number of nodes in the network.
- Let $L = (l_{mn})$ be the $N \times N$ matrix representing the physical topology, i.e.,

$$l_{mn} = l_{nm} = \begin{cases} 1 & \text{if there is an optical link between } m \text{ and } n \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

- Let $C = (c_{sd})$ be the $N \times N$ matrix representing the virtual topology, i.e.,

$$c_{sd} = c_{ds} = \begin{cases} 1 & \text{if there is a clear-channel between } s \text{ and } d \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

3.2.2 Variables

A solution to the problem consists of a set of values for the primary variable of the system.

Primary variable

- Clear-Channel Design:
  The route $\mathcal{R}_{sd}$ used by the clear-channel c_{sd} on the physical level is

$$\mathcal{R}_{sd} = \begin{cases} 0 & \text{if clear-channel } c_{sd} = 0 \\ (n_0, n_1, \ldots, n_k) & \text{otherwise, where } n_0 = s \text{ and } n_k = d \end{cases} \tag{3}$$

Auxiliary variables Those variables are obtained from the primary variable and the given constants. They help in formulating the optimization criteria.

- $\overline{C}_{mn} = (\overline{c}_{ij})$ is the $N \times N$ matrix representing the virtual topology after having removed all the clear-channels c_{ij} that have l_{mn} or l_{nm} as part of their route $\mathcal{R}_{ij}$,

$$\overline{c}_{ij} = \begin{cases} 0 & \text{if } (c_{ij} = 0) \cup ((n_n, n_m) \in \mathcal{R}_{ij}) \cup ((n_m, n_n) \in \mathcal{R}_{ij}) \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

- b_{mn}^{sd} equals 1 if clear-channel c_{sd} has no alternate path when link l_{mn} is broken,

$$b_{mn}^{sd} = \begin{cases} 1 & \text{if } (c_{sd} = 1) \cap (l_{mn} = 1) \cap (s \text{ and } d \text{ unconnected in } \overline{C}_{mn}) \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

- A *broken pair* is a couple $\{c_{sd}, l_{mn}\}$ for which $b_{mn}^{sd} = 1$.

Constraint

- $\mathcal{R}_{sd}$ is a sequence of nodes such that each pair $(n_0, n_1), \ldots, (n_i, n_j), \ldots,$ (n_{k-1}, n_k) are edges of L ($l_{n_i n_j} = 1$) and where no node is repeated.

Optimality Criterion The aim is to minimize the number of broken pairs.

Let us define:

$$n(l_{mn}) = \sum_{s,d} b_{mn}^{sd} \tag{6}$$

$n(l_{mn})$ is equal to the number of clear-channels having no disjoint alternate path in case of failure of the physical link l_{mn}. $n(l_{mn})$ represents the number of broken pairs that have l_{mn} as component.

A link l_{mn} is considered to be *critical* if $n(l_{mn}) \geq 1$. A clear-channel c_{sd} is *protected* if and only if any single link failure that causes c_{sd} to fail leaves an alternate path to c_{sd} not failed. A clear-channel c_{sd} is protected iff $\sum_{m,n} b_{mn}^{sd} = 0$ (from eq. (5)).

The function to minimize is then

$$\mathcal{F}(R) = \sum_{m,n} n(l_{mn}) \tag{7}$$

where

$$R = (\mathcal{R}_{ij}) \tag{8}$$

is the $N \times N$ network design matrix containing the routes $\mathcal{R}_{ij}$ of all the clear-channel, as shown in (3).

$\mathcal{F}(R)$ represents the number of broken pairs of the actual network design.

3.3 The Algorithm

The search for the best solution, R* which minimizes $\mathcal{F}$, requires the evaluation of $\mathcal{F}$ for all the possible values of R (eq. (8)). This means calculating, for each clear-channel, all the possible elementary routes between its two end-nodes in the physical network. Then the value of $\mathcal{F}$ must be computed for all the possible combinations of elementary routes for each clear-channel.

Let us define X as the number of clear-channels in the virtual topology and Y as the average number of possibilities to route a clear-channel in the physical topology. There are Y^X possible configurations for which $\mathcal{F}$ has to be calculated in order to find its minimum value. As in real cases both X and Y grows non-linearly with the number of nodes, the optimum search of such problems soon becomes untractable. We have not tried to build an algorithm to find the optimum solution, but a heuristic based on Tabu Search.

First, a small summary of the Tabu Search and how it applies to our routing problem is presented. Then the Disjoint Alternate Path (DAP) algorithm is described and finally, simulations show the results that we obtained, compared to the ones of a 'plain' shortest-path routing algorithm.

3.3.1 Tabu Search

Tabu Search can be defined as a general heuristic in which a local search procedure is applied at each step of the general iterative process. It could be superimposed on other heuristics to prevent them being trapped in a local optima. Tabu Search has been proposed almost at the same time by Glover [11] and Hansen [12] independently, and, for interested readers, a recent paper [13] gives a detailed description of the method.

To see how Tabu Search works, let us take an optimization problem, where a function $\mathcal{F}$ has to be minimized. Let S be the solution space containing all the particular solutions s, and let $\mathcal{F}$ be a function assigning to each element s of S a real value $\mathcal{F}(s)$. Then the aim is to find the solution s^* from S so that $\mathcal{F}(s^*)$ is minimum.

Tabu Search starts from an initial solution s_0 (given by a heuristic, or chosen arbitrarily in S) and explores S in an iterative way by moving from a solution to an other.

It is usual, as the search space can be very large, to limit the allowed modification to a class $\mathcal{M}$ of simple modifications m. Then, at a given step l in the iteration process, with a solution s_l, one can reach in one step from s_l all the neighbours (denoted $N(s_l)$) that are at a 'distance' m of s_l, that is $s_{l+1} \in N(s_l)$ or $s_{l+1} = s_l + m$.

The chosen s_{l+1} is the one that most reduces $\mathcal{F}$, and, if it cannot be reduced, the one that least augments it. By doing so, the algorithm is able to climb out a local minima.

The main idea of Tabu Search is that, in order to avoid cycles trapping the algorithm in a local minima, a *Tabu List* is created, which keeps track of the most recently visited solutions. Before moving to a new solution, the algorithm scans this list, and does not do the move if the new solution is in the list. So, a 'path' out of a local minima can be found. The use of such a list also help to diversify and intensify the search in the medium and long term.

The Tabu Search process is stopped when a given criterion is satisfied. It can be reaching the global minimum (if it is known) or, as happen most of the time, when a certain number of iterations without improving the best found solution until now has been reached.

In 3.3.2 we show how the Tabu Search method is applied to the resolution of our network design problem.

3.3.2 The DAP Algorithm

The aim of this algorithm is to find a network design matrix R (eq. (8)) which minimizes the function $\mathcal{F}$ (eq. (7)).

The algorithm consists of routing the various clear-channels in a way such that none of them (if it is possible) or, at least, the minimum number of them, are broken in case of any single optical link failure.

The algorithm implements the following principle:

```
(1) It starts from an arbitrary solution.
(2) It tries to modify the routes of clear-channels
      that have no disjoint alternate path,
      in a way to so as to avoid the link which causes a problem.
(4) The solution which has the minimal F is kept.
(5) The new list of problematic links and clear-channel
      is calculated based on the new matrix R.
(6) The next step is started from (2)
```

The algorithm implements the Proximate Optimality Principle [14], which stipulates that a good solution at one level is likely to be found close to a good solution at one adjacent level. In our case, it means that the starting solution for a network with n clear-channels routed will be the best solution for a network with $n-1$ clear-channels routed.

The algorithm starts with no clear-channel being routed and runs a *global* iteration until all are routed, having as starting solution at each step the best solution from the previous one plus a new clear-channel added using the Shortest Path Routing (SPR) algorithm.

SPRA means that the shortest route is always used to route a clear-channel on the physical network [15]. In our case, the *length* of a route is the number of optical links it uses.

At each step of the global iteration, a Tabu Search is performed, starting from the initial solution of this step. To reduce the number of possible modifications, let us restrict a *move* to a rerouting without using l_{mn} of only one clear-channel c_{sd} which was unprotected due to l_{mn} ($b_{mn}^{sd} = 1$, as described in eq. (5)). So, only the route of clear-channels having no disjoint alternate path can be modified. Each modification leads to a new R matrix and its associated $\mathcal{F}(R)$ is computed. The move chosen for this step is the one that most reduces $\mathcal{F}$.

In order to enlarge the visited solution space, and to avoid cycles, as described in 3.3.2, the clear-channels whose routes have been recently modified are placed in a Tabu List, which means that it is forbidden to try another route for them during a certain number of iterations. A stopping criteria, defined as the maximum number of iterations allowed with no decrease of $\mathcal{F}$ over the best found solution until now, is also needed, as we do not know the theoretical minimum of $\mathcal{F}$.

The appropriate size of the Tabu List, and the maximum number of iterations without improvment before stopping (to reach a good solution without spending too much time doing iterations) are normally found by empirical testing for a class of problem [16]. In our case, we have chosen values that seemed appropriate, due to the kind of problem we have [12], but further empirical work could be performed to refine these values.

The size of the Tabu List, ts, is equal to $\lceil \frac{1}{2} \cdot$number of clear-channels routed$\rceil$.

The maximum number of iteration, c, is equal to $\lceil \frac{3}{2} \cdot$ number of clear-channels routed$\rceil$.

The pseudo-code of the DAP algorithm is shown in Figure 5. One can well see the two nested **WHILE** loops: (a) The global one to give, at each step, a starting solution based on the best solution found at the previous step; (b) The local one, with a condition to stop the Tabu Search if no improvement has been performed after a certain number of iterations.

```
NotRoutedCC = {all CC}
WHILE  NotRoutedCC not empty  DO
   FOR  each CC of NotRoutedCC DO
      route it on the Shortest Path;
      compute F for this configuration;
      remove it;
   include in R the (not yet routed) CC that most reduced F;
   remove this CC from NotRoutedCC;
   compute Fmin based on R;
   c = 3/2 * number of CC routed; nbit = 0;
   ts = 1/2 * number of CC routed; tabulist = {};
   WHILE  nbit < c  DO
      compute the matrix CritMat containing, for each link,
         the CCs that are blocked by it;
      FOR  each CC of CritMat not in tabulist DO
         reroute it avoiding the blocking link;
         compute F for this configuration;
         restore R as before;
      include in R the new route of the CC that most reduced F;
      include this CC in tabulist;
      IF size(tabulist) > ts THEN
         remove the oldest CC from tabulist;
      IF  F < Fmin THEN   Fmin = F; nbit = 0; Rmin = R;
      ELSE  nbit ++;
   END WHILE;
END WHILE;
```

Figure 5: Pseudo-code of the DAP Algorithm. The global WHILE-loop gives, at each step, a starting solution based on the best solution found at the previous step and the local WHILE-loop performs the Tabu Search.

3.4 Simulation Results

The DAP Algorithm has been implemented in Mathematica 2.2 for Solaris on a Sparc 20 workstation.

The physical topology used for the tests was the ARPA2 network, as shown in Figure 6. The virtual topology has been defined by randomly generating clear-channels to obtain a (at least) biconnected network with a connectivity of 0.2 (20% of a fully-meshed network).

The DAP Algorithm has been tested with six different virtual topologies on the ARPA2 physical network, and the values of $\mathcal{F}$ for the design it proposes are

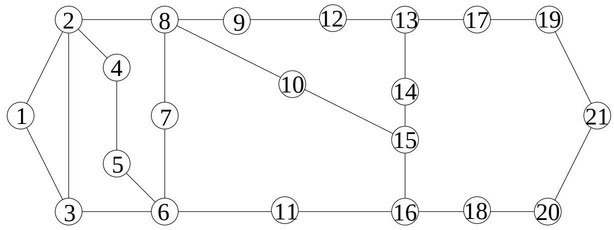

Figure 6: The 21-nodes 26-links ARPA2 physical network

compared with the ones obtained by a plain Shortest Path Routing Algorithm. The use of SPRA to perform the network design means that each clear-channel is placed on the physical topology using the shortest route (in number of used links), without any optimization towards resilience. The results are shown in Table 1.

Test nr	$\mathcal{F}_{SPR}$	$\mathcal{F}_{DAP}$
1	14	0
2	20	0
3	3	0
4	17	0
5	37	0
6	23	0

Table 1: The results of the tests on ARPA2 physical network: For each test, 42 clear-channels have been randomly generated. The network design has been performed using the SPR and DAP algorithm, and this table shows the number of broken pairs for each test.

As an example, the virtual topology of Test1 is shown in Figure 7 and Table 2 shows, for all the clear-channels CCs of the virtual topology, the routes proposed by the Shortest Path Routing algorithm (SPR Routes) and the ones proposed by our Disjoint Alternate Path algorithm (DAP Routes). The broken pairs for this design are listed in Table 3.

One can note from Table 3 that all the clear-channels having node 18 as one end-node are blocked – for the SPR design – if the optical link (16,18) is broken. One can see in Table 2 that when using the SPR design, clear-channels (5,18), (8,18), (11,18) and (16,18) have the optical link (16,18) as part of their route. This of course lead to a problem in case of link (16,18) failure, as it is the only one used to reach node 18, even if another one is available, the link (18,20). The design proposed by the DAP algorithm avoids this problem by choosing another route for the clear-channel (11,18), longer but not using the link (16,18) (see Table 2). This allows to have an alternate path not using link (16,18) for the three clear-channels that uses it – (5,18), (8,18) and (16,18) – and one not using link (18,20) for the clear-channel (11,18).

A limitation of the DAP algorithm is the fact that there is no constraint on the maximum number of clear-channels allowed per optical link. But it should be noted that this number stays low (11 for the most used link (6,11) in test1) and that DAP, by trying to avoid a clear-channel and its alternate path to share the same optical link, has a tendance to automatically spread the routes over all the links. A firm constraint on the maximal capacity of optical links is part of our future work.

Note that if $\mathcal{F}(\text{DAP}) \neq 0$, it does not mean that there is no solution, but that our heuristic has not found one. As a search through the whole solution space has not been perfomed, it is impossible to be sure that no solution exists.

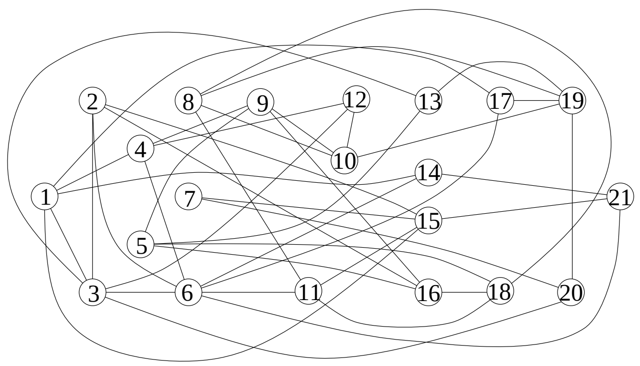

Figure 7: The 42 clear-channels virtual topology of Test 1

4 Conclusion

WDM optical networks allow an optimal use of the fibres' capacity by offering a wide bandwidth, necessary to respond to the needs of actual applications. But we are also witnessing a rapidly increasing concern about the reliability of the communications, as the consequences of a failure in high-speed networks are proportional to the amount of information flowing through the network.

In this article, we have studied the various components of a proposed architecture for a WDM photonic network and we have defined two possible levels where protection could be implemented.

Physical Protection provides a reliable transport capability to the higher level networks. It is expensive to implement due to the need to duplicate all the optical hardware, but it extremely simple to control. For Physical Protection, it is the photonic network that is responsible for performing the necessary reconfiguration following a failure, and will do so on a very fast timescale.

Design Protection is a new kind of protection well adapted to WDM networks, where many channels share the same fibre. It should be noted that the concept applies to cases where the virtual network is SDH, ATM or even directly a router (IP) network. Design Protection does not guarantee to carry all the traffic that was carried prior to the component of optical link failure. Instead, it aims to maintain connectivity between all pairs of network ports following a single failure, and lets the higher level network reconfigure itself so as to carry only the high priority traffic from the failed virtual channels. The fraction of the disrupted traffic that can be accommodated on the Alternative Path will be dependent on the loading of the higher level network.

The independence between the virtual and physical topologies offers a lot of possibilities to design clear-channels. We have shown the problems that failure propagation causes if the network design is realized without taking into account the interraction between the two levels. As the resolution of this problem is rather complex, a heuristic, the DAP algorithm, has been developed to find rapidly a good solution to this problem.

Simulations have shown that by using this DAP algorithm to place the clear-channels at the design's time, the network is much more resilient to single optical link failures than when clear-channels are placed using SPRA.

Our future work will be oriented towards the introduction of maximal capacity for the optical links and nodes (i.e. having a maximum number of channels per fibre, or of receiver/transmitter per node) and also to adapt the algorithm to have the possibility to associate failure probabilities to the links.

Lastly, it should be noted that Design Protection applies to other settings like (1) an SDH network where the "physical topology" is made of fibres or clear-channels and the "virtual topology" is made of SDH end-to-end paths or (2) an ATM network where the "physical topology" is made of SDH paths and the "virtual topology" is made of ATM Virtual Paths connections.

References

[1] P. E. Green, "The Future of Fiber-Optic Computer Networks," *IEEE Computer*, Sept. 1991.

[2] "Special issue on dense Wavelength Division Multiplexing techniques for high capacity and multiple access communications systems," *Journal of Selected Areas in Communication*, vol. 8, Aug. 1990.

[3] B. Mukherjee, "WDM-based local lightwave networks - part I: Multihop systems," *IEEE Networks*, pp. 12–27, May 1992.

[4] B. Mukherjee, "WDM-based local lightwave networks - part II: Multihop systems," *IEEE Networks*, pp. 20–32, July 1992.

[5] R. Ramaswami, "Multiwavelength Lightwave Network for Computer Communication," *IEEE Communications Magazine*, pp. 78–88, Feb. 1993.

[6] "Physical and networking layer issues of COBNET," Deliverable AC069-/COB/WP1/DS/R/111/b1, COBNET Consortium, Mar. 1996.

[7] C. Brackett *et al.*, "A Scalable Multiwavelength Multihop Optical Network: A Proposal for Research on All-Optical Networks," *Journal of Lightwave Technology*, vol. 11, pp. 736–753, May 1993.

[8] G. Hill *et al.*, "A transport network layer based on optical network elements," *Journal of Lightwave Technology*, vol. 11, pp. 667–679, May 1993.

[9] H. S. Hinton, "Photonic Switching Fabrics," *IEEE Communications Magazine*, Apr. 1990.

[10] S. Johansson *et al.*, "Optical cross-connect system in guarantee-band networks: System concept and demonstrator description," *Journal of Lightwave Technology*, vol. 11, pp. 688–694, May 1993.

[11] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers and Operations Research*, no. 13, pp. 533–549, 1986.

[12] P. Hansen, "The steepest ascent mildest descent heuristic for combinatorial programming," in *Proc. Congr. on Numerical Method in Combinatorial Programming*, Academic Press, 1986.

[13] F. Glover, E. Taillard, and D. de Werra, "A user's guide for tabu search," *Annals of Operations Research*, no. 41, pp. 3–28, 1993.

[14] A. Lokketangen, "Tabu search - using the search experience to guide the search process. an introduction with examples," in *AICOM*, vol. 8, pp. 78–85, June 1995.

[15] B. Mukherjee *et al.*, "Some principles for designing a wide-area optical network," in *Proc. of INFOCOM*, vol. 1, pp. 110–119, 1994.

[16] F. Glover and M. Laguna, "Tabu search. modern heuristic techniques in combinatorial problems," pp. 70–150, Blackwell, 1993.

clear-channel	*SPR route*	*DAP route*
(1,3)	(1,3)	(1,3)
(1,4)	(1,2,4)	(1,2,4)
(1,14)	(1,2,8,10,15,14)	(1,2,8,9,12,13,14)
(1,15)	(1,2,8,10,15)	(1,2,8,10,15)
(1,17)	(1,2,8,9,12,13,17)	(1,2,8,9,12,13,17)
(2,3)	(2,3)	(2,1,3)
(2,6)	(2,3,6)	(2,3,6)
(2,15)	(2,8,10,15)	(2,8,10,15)
(2,16)	(2,8,10,15)	(2,3,6,11,16)
(3,6)	(3,6)	(3,6)
(3,12)	(3,2,8,9,12)	(3,6,11,16,15,14,13,12)
(3,20)	(3,6,11,16,18,20)	(3,6,11,16,18,20)
(4,6)	(4,5,6)	(4,5,6)
(4,9)	(4,2,8,9)	(4,2,8,9)
(4,12)	(4,2,8,9,12)	(4,2,8,9,12)
(5,9)	(5,4,2,8,9)	(5,4,2,8,9)
(5,13)	(5,6,11,14,13)	(5,6,11,14,13)
(5,16)	(5,6,11,16)	(5,6,11,16)
(5,18)	(5,6,11,16,18)	(5,6,11,16,18)
(6,11)	(6,11)	(6,11)
(6,14)	(6,11,16,15,14)	(6,11,16,15,14)
(6,17)	(6,7,8,9,12,13,17)	(6,7,8,9,12,13,17)
(6,21)	(6,11,16,18,20,21)	(6,11,16,18,20,21)
(7,15)	(7,8,10,15)	(7,8,10,15)
(7,20)	(7,6,11,16,18,20)	(7,6,11,16,18,20)
(8,10)	(8,10)	(8,10)
(8,11)	(8,7,6,11)	(8,7,6,11)
(8,18)	(8,10,15,16,18)	(8,10,15,16,18)
(8,19)	(8,9,12,13,17,19)	(8,9,12,13,17,19)
(9,10)	(9,8,10)	(9,12,13,14,15,10)
(9,16)	(9,8,10,15,16)	(9,8,10,15,16)
(10,12)	(10,8,9,12)	(10,8,9,12)
(10,19)	(10,15,14,13,17,19)	(10,15,14,13,17,19)
(11,15)	(11,16,15)	(11,16,15)
(11,18)	(11,16,18)	(11,16,15,14,13,17,19,21,20,18)
(13,19)	(13,17,19)	(13,17,19)
(14,21)	(14,13,17,19,21)	(14,13,17,19,21)
(15,21)	(15,16,18,20,21)	(15,16,18,20,21)
(16,18)	(16,18)	(16,18)
(17,19)	(17,19)	(17,19)
(19,20)	(19,21,20)	(19,21,20)

Table 2: The network design proposed by the SPR and DAP algorithms for the Test 1 virtual topology. For each clear-channel CC, the route chosen by the two algorithms are shown.

Link	Non-protected clear-channels for SPRA
(8,9)	(3,12), (4,9), (4,12), (5,9), (9,10), (9,16), (10,12)
(9,12)	(3,12), (4,12), (10,12)
(16,18)	(5,18), (8,18), (11,18), (16,18)

Table 3: The broken pairs of the SPR design shown in Table 2. NB In the DAP design all the clear-channels are protected. For example, (3,12) was not having any alternate path in case of links (8,9) or (9,12) failure when routed by the SPR algorithm but is protected for any link failure when routed by the DAP.