Waveband Protection Mechanisms in Hierarchical Optical Networks

Aikaterini C. Varsou, Samrat Ganguly and Rauf Izmailov
NEC Laboratories America,
4 Independence Way, Princeton, NJ 08540
{avarsou, samrat, rauf}@nec-labs.com

Abstract—We consider protection and restoration mechanisms for hierarchical optical networks with different granularities of paths (wavelengths and wavebands). We formulate an Integer Linear Programming description for the problem and propose a shared-path waveband protection algorithm. We further demonstrate its efficiency by analyzing its equipment cost and capacity utilization.

I. INTRODUCTION

The continuing increase of data traffic keeps the pressure on the backbone telecommunication networks. In order to satisfy the growing bandwidth demands, more diverse and more intelligent allocation of capacity is required. Optical networking has become a key technology to accommodating the rapidly expanding Internet traffic. New optical networks are expected to support the increasing network load by employing both sophisticated transmission (wavelength division multiplexing (WDM)) and switching (optical switches and cross-connects) technologies [6]. These advanced technologies pose various challenges that need to be addressed.

In optical networks, cost and scalability concerns translate into creation of multiple switching granularities, such as wavelengths and wavebands. The optical paths thus form a hierarchy in which a higher-layer path (waveband) consists of several lower layer paths (wavelengths). The potential cost benefits of wavelength aggregation [1], [2] into wavebands were demonstrated in [3], [4]. A waveband path occupies only two (input and output) ports of an optical switch in a node. The path hierarchy reduces node costs because a waveband can be switched optically as a single unit, thus reducing the number of expensive OEO ports required for processing individual wavelengths.

Cost-efficient implementation of optical hierarchy has to be delivered by appropriately designed routing and scheduling algorithms. Routing and wavelength assignment algorithms were extensively studied in the general context of optical networking. The hierarchy of wavelengths and wavebands can be cast in several models posing new routing and scheduling challenges.

Protection and restoration consist another challenging problem. Wavelength connections can be protected on "end-to-end" and on "span" basis (the latter method provides faster restoration than the former one, but with larger bandwidth waste), while waveband connections require "end-to-end" protection.

If both types of connections are to be supported in hierarchical networks, some type of internetworking will be needed.

In this paper, we develop a shared-path protection/restoration algorithm for wavebands. To our knowledge, this problem has not been addressed before in the literature. The problem is formulated through integer linear programming and a specific algorithm is proposed and analyzed. The main structure of the algorithm that protects the network is a modified spanning tree, called "Waveband/wavelength Protection Tree" (WP-tree).

The rest of the paper is organized in the following manner. In the next section, we present the model of the network and the integer linear programming formulation. Section III provides our proposed protection algorithm followed by simulation results. We conclude in Section V with the scope of our future work.

II. SYSTEM MODEL AND ILP FORMULATION

Before presenting the details of our waveband protection algorithm, we describe the problem and give an integer linear programming formulation. We assume that the network topology is given and expressed as a directed graph. We also know the number of lightpath requests between pairs of nodes and the alternate routing tables at each node and the wavelength assigned to each route. The object is to protect an active waveband path of capacity G (i.e. carrying G wavelengths) that connects a source-destination pair. This can be done in two possible ways. In the first case, a backup waveband path...
of the same capacity needs to be found for the given active path connecting the specific source-destination pair. In the second scenario, the waveband route can be protected by several backup wavelength paths. For each active waveband for source-destination pair \(i\), we need to decide which of the two protection methods will be used. The decision function is the spare capacity that needs to be minimized. If there exist two different protection ways that require the same amount of spare capacity, the decision factor should be the minimum port cost for each case. From now on, the terms “source-destination pair”, “route”, “path”, “flow” will be used interchangeably. We will be using the following notation:

- \(N\): number of nodes in the network (the nodes are numbered from 1 through \(N\) and the flows are numbered from 1 through \(N(N-1)\))
- \(E\): number of links in the network (numbered from 1 through \(E\))
- \(W\): maximum number of wavelengths on a link
- \(R_i^f\): set of alternate waveband paths for path \(i\)
- \(B_i^f\): set of eligible backup wavelength routes for path \(i\), all of them disjoint to the active route
- \(L_i^f\): set of eligible backup wavelength routes for flow \(i\), all of them disjoint to the active route
- \(A_i\): set of active flows using link \(l\)
- \(M = \max M_{si}\), where \(M_i = |R_i^f|\)
- \(s_i^{(1)}, s_i^{(2)}\): spare capacity for link \(l\) under protection scenarios 1 and 2 respectively
- \(w_i\): number of active wavelengths used on link \(l\)
- \(\gamma_{wi}^{r}\): binary variable, equal to 1, if route \(r \in R_i^f\) utilizes wavelength \(w\) before any link failure, or equal to zero otherwise
- \(\delta_{wi}^{r}\): binary variable, equal to 1, if backup waveband route \(r \in B_i^f\) utilizes wavelength \(w\), or equal to zero otherwise
- \(\lambda_w\): binary variable, equal to 1, if wavelength \(w\) is utilized on link \(l\) by some backup waveband route, or equal to zero otherwise
- \(G_w\): binary variable, equal to 1, if waveband path \(r \in B_i^f\) protects path \(i\), or equal to zero otherwise
- \(h_{k}^{(1)}, h_{k}^{(2)}\): length, i.e. number of hops, for waveband (wavelength) route \(r\) respectively
- \(\chi_{ki}^{(1)}, \chi_{ki}^{(2)}\): binary variable, equal to 1, if wavelength path \(k \in L_i^f\) traverses link \(l\), or equal to zero otherwise
- \(\beta_{ki}^{(1)}\): binary variable, equal to 1, if path \(k \in L_i^f\) protects flow \(i\), or equal to zero otherwise
- \(\beta_{ki}^{(2)}\): binary variable, equal to 1, if the active waveband path \(i\) is protected by a backup wavelength path (protection scenario 1), or equal to zero, if the active waveband path is protected by \(G\) backup wavelength paths (protection scenario 2).

In [7], the authors develop an ILP for shared-protection for wavelengths in WDM mesh networks. For the case of a waveband path being protected by another waveband path/route, we can similarly write:

1) The port cost will be equal to \(4G + 2h_{k}^{(1)}\) backup WB route, which can be equivalently written as

\[
C^{(1)} = \sum_{i} \left[ 4G + 2 \sum_{r \in B_i} (h_{k}^{(1)} c_{r}) \right].
\]  
(1)

2) The spare capacity required on link \(l\) is equal to

\[
s_i^{(1)} = \sum_{w=1}^{W} \lambda_w^{i}, \quad \forall l_i.
\]  
(2)

3) and should be upper bounded by

\[
s_i^{(1)} \leq W - \sum_{i \in R_i^f} \sum_{r \in B_i^f} \gamma_{wi}^{r}, \quad \forall l_i.
\]  
(3)

4) There are also constraints indicating whether wavelength \(w\) is utilized for some backup path \(r\) on link \(l\):

\[
\lambda_w^{i} \leq \sum_{r \in R_i^f} \delta_{wi}^{r}, \quad \forall i, w, \quad i.
\]  
(4)

5) Only one primary or backup wavelength can use wavelength \(w\) on link \(l\):

\[
\sum_{r \in R_i^f} \gamma_{wi}^{r} = \sum_{r \in B_i^f} \delta_{wi}^{r} \leq 1, \quad \forall i, w, \quad i.
\]  
(5)

6) For the total number of rerouted lightpaths for flow \(i\), we can write

\[
\sum_{r \in R_i^f} \sum_{w=1}^{W} \gamma_{wi}^{r} + \lambda_w^{i} \leq 1, \quad \forall i.
\]  
(6)

For the case of a waveband path being protected by \(G\) wavelength paths, we can write

1) The port cost for each wavelength path used is equal to \(2h_{k}^{(2)}\) backup WB path, so the total cost for a waveband being protected by \(G\) wavelength paths is

\[
C^{(2)} = 2 \sum_{i} \sum_{k \in L_i^f} (h_{k}^{(2)} \theta_k^{i}),
\]  
(8)

where \(\sum_{k \in L_i^f} \theta_k^{i} = G, \quad \forall i.
\)  
(9)

2) The spare capacity should satisfy

\[
s_i^{(2)} \geq \sum_{i \neq f} \chi_{ki}^{(2)} \beta_{ki}^{(2)}, \quad \forall i, f : i \neq f.
\]  
(10)

3) Since each waveband should be protected by \(G\) wavelength paths, it should also hold

\[
\sum_{k \in L_i^f} \beta_{ki}^{(2)} = G, \quad \forall i.
\]  
(11)

4) and

\[
\beta_{ki}^{(2)} \leq G, \quad \forall i, k : i \leq k.
\]  
(12)

We can therefore see that the shared-path protection can be formulated to an integer linear programming problem. Combining the two different scenarios through usage of the variables \(\epsilon_i\), we can choose the most efficient of the two scenarios and can therefore write the complete ILP for our problem:

\[
\min \sum_{i} \left[ \epsilon_i \sum_{l \in L_i^f} s_i^{(1)} + (1 - \epsilon_i) \sum_{l \in L_i^f} s_i^{(2)} \right].
\]  
(13)
\[\min \sum_i \left\{ \varepsilon_i \left[ 4G + 2 \sum_{r \in B^i} \left( \lambda^{(1)}_{ir} \right) \right] + (1 - \varepsilon_i) \sum_{k \in L^i} (2\lambda^{(2)}_{ik} \delta_k) \right\}\]

subject to

\[s^{(1)}_l = \sum_{w=1}^W \lambda^{(1)}_{lw}, \quad \forall l\]

\[s^{(1)}_l \leq W - \sum_{i \in P} \sum_{r \in B^i} \sum_{w=1}^W \gamma^{lr}_{w}, \quad \forall l\]

\[s^{(2)}_l \geq \sum_{i \in A_T} \sum_{k \in L^i} \left( \lambda^{(1)}_{ik} \beta^{i,k} \right), \quad \forall l, f : i \neq f\]

\[\lambda^{(1)}_{lw} \leq \sum_{i \in B^l} \sum_{r \in E^l} \delta^{(2)}_{lw}, \quad \forall l, w\]

\[\sum_{i \in B^l} \sum_{r \in E^l} \gamma^{lr}_{w} + \lambda^{(1)}_{lw} \leq 1, \quad \forall l\]

\[\sum_{r \in B^l} \sum_{w=1}^W \gamma^{lr}_{w} = \sum_{r \in B^l} \sum_{w=1}^W \delta^{(2)}_{lw}, \quad \forall l, w\]

\[\sum_{k \in L^i} \beta^{i,k} = G_i, \quad \forall i\]

\[\beta^{i,k} \leq G_i, \quad \forall i, k : i \leq k.\]

This is an Integer Linear Programming Problem with two optimization criteria and has been discussed in [10], [11] and their references. There are also a few free software packages solving multiple objective linear programming problems, including ADBASE, PROTASS and NIMBUS. See [9] for more information and links to the specific softwares. We now continue by describing the actual protection algorithm in the next section.

III. WAVEBAND/WAVELENGTH PROTECTION TREES

Our algorithm is based on a modified spanning tree, referred to as "Waveband/Wavelength Protection Tree" (WP-tree). The network is protected by one or multiple WP-trees that are constructed in a way that takes into account the number of paths that can be protected and the number of available wavebands. The following are the main steps for the construction of a WP-tree.

1) First, we check which paths intersect; for each link \(l\) in the network, we keep a list \(P_l\) of the paths that it can protect, as well as a list \(W_l\) of the wavebands that are available to it.

2) The first link to be chosen for the WP-tree \(T\) is the one that can protect the maximum number of paths. At this point, the current WP-tree consists of only one link. As links are being added to the tree, the number of available wavebands for the corresponding link is reduced by one, reflecting the link capacity constraints.

3) For a current WP-tree, we keep the following information: a list of the links \(L_T\) and nodes \(N_T\) it consists of, the disjoint paths \(P_T\) that it can protect either completely or partially and the wavebands \(W_T\) that are available to it.

4) Given a WP-tree, we first find the set of all its adjacent links \(A_T\) (i.e., the links adjacent to the nodes of the tree). Each of these links \(l\) is assigned a path weight \(p_l\) and a waveband weight \(w_l\) that are proportional, respectively, to the number of paths \(|P_l|\) it can protect and to the number of available wavebands \(|W_l|\).

5) The next link \(L\) to be added to the tree should: (i) be adjacent to it, (ii) not be part of a cycle, (iii) protect the maximum number of paths (by having the maximum path weight). If more than one such link exists, the tie is broken with the waveband weight. Once \(L\) is added to the WP-tree, the list of links and nodes the new tree consists of is updated. The paths protected by the new tree are found (i) by intersecting the disjoint paths protected by that the old tree with the disjoint paths protected by \(L\) protects, and (ii) by checking whether each such path is completely protected. The available wavebands for the new WP-tree are also updated: they are the intersection of the wavebands that were available to the old tree with the wavebands that are available for \(L\). If a path is completely protected by the WP-tree, it is removed from the list of paths that a link can protect (we do not want to completely protect the same path twice). Each time a new link is added to the WP-tree, the efficiency score (defined as the number of paths that the tree can protect divided by the number of links in the tree) is updated. The construction algorithm continues until one of the following conditions holds: (i) there is no potential candidate link; (ii) candidate links cannot protect any common paths or use any common bands as the current WP-tree, or (iii) the efficiency score is below some pre-configured value.

WP-trees are being constructed as long as there are still unprotected paths and the current WP-tree protects at least one complete path. When all possible WP-trees are constructed, the algorithm checks the lists of available wavebands for each tree and decides how to assign which band to which tree. If some of the paths are still unprotected at this point, a wavelength protection scheme should be initialized. The same algorithm could be used. Note that a wavelength level protection path could undergo wavelength conversion.

Now let us denote by

- \(e(l)\): the end nodes of a link \(l\)
- \(P\): the set of all paths
Fig. 2. An example of a mesh network and its protecting WP-tree

- CP: the set of all completely protected paths
- scoreT: the score for the WP-tree T, equal to $P_T/|L_T|$, always greater than a preconfigured value $v$

and also

Definition 1. We say that a link $l$ belongs to the set $C_T$ of WP-tree $T$ if adding link $l$ to the tree $T$ would create a cycle.

Then, the algorithm above can be described by the following pseudocode

Algorithm 1. Protection heuristic
1) $i = 0$; //create i-th tree $T_i$
2) do{
   a) at least one link $\leftarrow 0$; //no links in tree yet
   b) $L_T \leftarrow N_T \leftarrow P_T \leftarrow W_T \leftarrow \emptyset$
   c) do{
      - update $P_i, W_i, p_i, w_i$ for each link $i$
      - find $l \in A_T$ st $l \notin C_T$ and $p_l = \max_{k \in A_T}$ $p_k$.
      - If there exist more than one such $l$, choose the one with $\max w_l$.
      - If there does not exist such $l$, break;
      - Add $l$ to tree $T_i$, i.e.
         - $L_T \leftarrow L_T \cup \{l\}$
         - $N_T \leftarrow (N_T \cup e(l)) \cup \{l\} \neq N_T$
         - $P_T \leftarrow P_T \cap P_l$;
         - if $(P_T = \emptyset)$ break;
         - $W_T \leftarrow W_T \cap W_i$;
         - if $(W_T = \emptyset)$ break;
      - $\forall P \in P_T$:
         - if $(P \in CP)$, then $P_l \leftarrow P_l \setminus \forall $ link $l$
         - score $\leftarrow P_T/|L_T|$
         - if (score $< v$) break;
         - at least one link in tree
   }while(true)
   d) $i \leftarrow i + 1$; //new tree
}while($\exists P \in P$ st $P \notin CP$ AND (at least one link))
3) check $W_{Ti}$, $0 \leq k \leq i - 1$ and assign bands/colors
4) while ($\exists P \in P$ st $P \notin CP$) do wavelength protection

Fig. 2 shows an example of a mesh network and the sequence of links added to the WP-tree (for simplicity, disjoint paths are not considered). The first link of the WP-tree is (0,1). Subsequently, the links (1,2), (2,3), (3,7), (7,11), (11,15), (15,14), (14,13), (13,12) are added. At this point, no more links can be added to the current WP-tree and paths 0, 2 and 3 are already completely protected. The paths and available bands sharing among the links is such that this is the only WP-tree created for this network.

IV. RESULTS OF SIMULATIONS

In order to evaluate the performance of our proposed algorithm, we considered a mesh network of size $9 \times 9$, with wavebands consisting of 8 wavelengths each. The active wavebands paths were first computed using the offline waveband routing and aggregation algorithm presented in [1]. According to this algorithm, all wavelength demands are first routed along the shortest available path in the network. In the subsequent steps, the lightpaths sharing the longest common segment (in terms of links) are grouped with $G$ lightpaths in each group, to form a waveband traversing the segment. These steps are carried out until no further grouping is possible.

Then, we computed the protection paths for these active bands using two approaches. In the "Mixed" approach, we run our protection algorithm to find the backup paths; in the "Wavelength" approach, the protection paths are found by...
computing disjoint wavelength paths for the active wavebands. The backup wavelength paths can use OEO ports for wavelength conversion.

Notice that, keeping the same traffic pattern (distribution of traffic on source-destination pair), the load parameter, as used in the simulations, refers to a scalar multiplier to generate different volume of traffic.

Figure 3 shows the performance in terms of spare capacity under different load scenarios, for a score value of 0.01, while Fig. 4 shows the port cost for the same loads. The port cost (required wavelength and waveband ports) is calculated based on hybrid cross-connect architecture [1]. Also, in Fig. 5, we give the number of WP-trees created by our algorithm versus the same loads and for the same score. We notice from this last figure that as the load increases, more WP-trees need to be created in order to protect the paths, with the exception of load equal to 13, where the traffic created is such that only 13 WP-trees are enough to protect the paths. Since the number of trees is increasing, it makes sense to notice an increase in the spare capacity in Fig. 3. Observe that the spare capacity for load = 13 does not follow the increase, since at that load the number of trees created does not follow the tree curve either. In terms of port cost, our proposed scheme always outperforms the “Wavelength” approach.

Furthermore, we plot the number of WP-trees created versus the score, for a fixed load, equal to approximately 10.58. This result is depicted in Fig. 6. Also the spare capacity and the port cost versus the score for the same fixed load are shown in Figs. 7 and 8 respectively. Remember that the score is equal to the ratio of the number of paths the tree protects over the number of links of the tree. So when the score is low, more trees need to be created and hence we would expect more resources (capacity) (Figs. 6 and 7 respectively). On the other hand, the algorithm performs well, since it reduces the port cost as the number of trees created increases. Indeed, this can be verified from Fig. 8, where the cost is low for low score and hence high number of trees. The cost is higher for high score, since in this particular case one huge WP-tree was created.

Finally, since it seems that the score and load are interconnected with the performance results, it would be interesting to check how the spare capacity and cost change for different scores for the same WP-tree. For this, we concentrate on the case where load = 19.5. It turns out that one WP-tree is created for scores ranging from 0.2 to 0.5 but the cost and capacity slightly change, according to Figs. 9 and 10.
V. CONCLUSIONS

We proposed and analyzed a mechanism for protecting waveband paths in hierarchical hybrid optical networks. Compared to wavelength-level protection, our approach can achieve cost reduction for waveband protection. We plan to evaluate our approach on more diverse networks and traffic scenarios.

REFERENCES


Fig. 9. Spare capacity comparison for different scores, for one WP-tree at load = 19.5

Fig. 10. Cost comparison for different scores, for one WP-tree at load = 19.5