Double-link failure protection algorithm for shared sub-path in survivable WDM mesh networks

Lei Guo (郭 磊), Hongfang Yu (虞红芳), and Lemin Li (李乐民)

Key Lab of Broadband Optical Fiber Transmission and Communication Networks,
University of Electronics Science and Technology of China, Chengdu 610054

Received April 12, 2004

We propose a novel shared sub-path protection (SSPP) algorithm to protect the double-link failures in wavelength division multiplexing (WDM) mesh networks. SSPP segments the primary path into several equal-length sub-paths and searches two link-disjoint backup paths for each sub-path. When computing the paths, SSPP considers the load balance and the resource sharing degree, so that the blocking ratio can be effectively reduced. The simulation results show that SSPP not only can completely protect the double-link failures but also can make the tradeoffs between the resource utilization ratio (or blocking ratio) and the protection-switching time.

OCIS codes: 060.4250, 060.4510.

In survivable wavelength division multiplexing (WDM) networks, a wavelength channel has the transmission rate of over gigabits per second (e.g., OC-48, OC-192 or OC-768, etc.)[1]. If the fiber links fail, then a lot of connection streams[2] are dropped. Thus, the protection designs are important in WDM optical networks. The conventional protection designs mostly considered the single-link failure[3]. Lots of users increasing heavily lead to the size of networks keeping enlarging, and so many heterogeneous networks interconnecting lead to more and more complicated structure of networks. Then, the probability of the failure risks become much higher, and the double-link failures must be considered for protection designs in WDM optical networks.

An algorithm, called shared-path protection (SPP), has been proposed for protecting the double-link failures[4]. SPP searches two link-disjoint backup paths for the primary path of each connection request. However, when searching the paths, SPP does not consider the load balance and the resource sharing degree, and the paths are not minimum cost but minimum hop, thus the reserved resources are not shared effectively and the resource utilization ratio is not optimal[5]. SPP also does not consider the protection-switching time, so that the failure recovery time is not the best[6].

In order to effectively improve the resource utilization ratio and neatly make the tradeoffs between the protection-switching time and the resource utilization ratio, we propose a novel algorithm, called shared sub-path protection (SSPP), for protecting the double-link failures.

Before describing the proposed algorithm, the following notations are introduced.

$G(N, L, W)$ is a network topology for a given WDM mesh network, where N is the set of nodes, L is the set of fiber links, and W is the set of available wavelengths per fiber link. $r(s,d,b)$ is a connection request, s, $d \in N$ denote the source node and the destination node respectively, and b specifies the bandwidth requirement of the request.

l ($\in L$) is a bi-directional fiber link in G, and the total capacity is C. c_l is the basic cost of the link l, and it is determined by many factors, such as the physical length of the fiber link, the installation cost of the fiber link, and so on. c_l' is the cost of link l and it is determined by the current state of the network. a_l is the total bandwidths consumed on the link l. r_l is the total residual bandwidths on the link l, and $a_l + r_l = C$ should be satisfied. w_l is the total bandwidths already consumed by the primary paths of the connection requests. p_{1l} and p_{2l} are the reserved bandwidths on the link l, and $p_{1l} \geq p_{2l}$. $p_{1l} + p_{2l} + w_l = a_l$ should be satisfied. tp_{1l} and tp_{2l} are the reserved bandwidths of temporary records on the link l, and $tp_{1l} \geq tp_{2l}$ should be satisfied.

sp_n is a sub-path, where n is the sub-path identification Id, which can be distributed by arriving the orders of the connection request and the segmenting orders of the sub-paths. bp_{1n} and bp_{2n} are the first and the second backup paths for sp_n, respectively. DP_{1n} and DP_{2n} are the sets of sub-path Ids that affect p_{1l} and p_{2l}, respectively. tDP_{1n} and tDP_{2n} are the sets of sub-path Ids that affect tp_{1l} and tp_{2l}, respectively.

$|S|$ is the number of elements in the set S.

The path searching algorithm, Dijkstra, is applied to compute the routes. In our study, we allow wavelength conversion. Now, we describe the SSPP algorithm.

Step 1: Wait for a connection request arrival. If a connection request arrives, go to Step 2. Otherwise, update the state of the network and go back to Step 1.

Step 2: Before computing the primary path, c_l' is calculated in

$$c_l' = \begin{cases} +\infty & r_l < b \\ -\alpha r_l + c_l & \text{otherwise} \end{cases}, \tag{1}$$

where α is a constant that considers the load balance, and $\alpha r_l + c_l > 0$ should be satisfied. After adjusting the link-cost according to Eq. (1), SSPP computes a minimum cost primary path, and $0 < C_w < +\infty$ should be satisfied, where C_w is the total costs of all links of the primary path. If fail to find, abandon this connection request and go to Step 6. Otherwise, segment the primary path into several equal-length sub-paths, and go to Step 3.

Step 3: The sub-path Ids that have yet held on the network before n compose the set B. We assume $Y = \{j;\ j \in B \cap (l \in bp_{1j} \cup l \in bp_{2j})\}+\{n\}$ and $U = \{f;\ f \in sp_n\}$ for arbitrary link l. We firstly find the sub-paths (in Y) that traverse the link k ($\in U$) and record their corresponding sub-path Ids. Then, compute the sum of the corresponding request bandwidths. Lastly, let p_1 and p_2 be equal to the first larger and the second lager of these sums, respectively. Assume $N = \{p_{1l}, p_{2l}, p_1, p_2\}$, then tp_{1l} is equal to the maximum(N), and the corresponding sub-path Ids compose the set tDP_{1l}. Let $N = N - \{\text{maximum}(N)\}$ and delete the repeated sub-path Ids in the sets, where the sets correspond to the elements in N respectively, and have repeated sub-path Ids with tDP_{1l}. Then, tp_{2l} is equal to the maximum(N), and the corresponding sub-path Ids compose the set tDP_{2l}. c'_l is calculated in

$$c'_l = \begin{cases} +\infty & (l|U \neq \emptyset) \cup (r_l + p_{1l} + p_{2l} \\ & \qquad < tp_{1l} + tp_{2l}) \\ -\eta C + c_l & p_{1l} + p_{2l} \geq tp_{1l} + tp_{2l} \\ -\eta r_l + c_l & \text{otherwise} \end{cases}, \tag{2}$$

where η is a constant, which considers the resource sharing degree, and $\eta r_l + c_l > 0$ should be satisfied. After adjusting the link-cost according to Eq. (2), SSPP computes the first link-disjoint and minimum cost backup path (bp_{1n}), and $0 < C_b < +\infty$ should be satisfied, where C_b is the total costs of all links of bp_{1n}. If fail to find, abandon this request and go to Step 6. Otherwise, go to Step 4.

Step 4: Assume $Q = \{f;\ f \in bp_{1n}\} + U$, and c'_l is calculated in

$$c'_l = \begin{cases} +\infty & (l|Q \neq \emptyset) \cup (r_l + p_{1l} + p_{2l} \\ & \qquad < tp_{1l} + tp_{2l}) \\ -\eta C + c_l & p_{1l} + p_{2l} \geq tp_{1l} + tp_{2l} \\ -\eta r_l + c_l & \text{otherwise} \end{cases}. \tag{3}$$

After adjusting the link-cost according to Eq. (3), SSPP computes the second link-disjoint and minimum cost backup path (bp_{2n}), and $0 < C_b < +\infty$ should be satisfied. If fail to find, abandon this request and go to Step 6. Otherwise, go to Step 5.

Step 5: If all sub-paths have been assigned the backup paths, then memorize the finding primary and backup paths and the reserved resources and update the temporary records, and go to Step 6. Otherwise, choose a sub-path that has not been assigned the backup paths and go back to Step 3.

Step 6: Check the connections that leave the network and update the state of the network. Go back to Step 1.

The protection-switching time for a sub-path can be defined as the period from the links failing to a backup path starting working, if the sub-path traverses the failed links. The following notations are introduced according to Ref. [3].

D, the message processing time at a node, is 10 μs;

P, the propagation delay on each link, is 400 μs;

X, the time to configure an optical cross connection (OXC), is 10 μs;

F, the time to detect the link failures, is 10 μs. We assume the double-link failures are detected simultaneously;

l, k are the two failed links.

If l is traversed by the sub-path r while k is traversed by the first backup path, the number of hops from the l source to the sub-path source node is n_1, the number of hops from the k source to the sub-path source node is n_2, and the number of hops of the second backup path is h. The protection-switching time t_r for the sub-path r is calculated in

$$\begin{aligned} t_r &= F + n_1 \times P + (n_1 + 1) \times D + 2 \times n_2 \times P \\ &\quad + 2 \times (n_2 + 1) \times X + 2 \times (n_2 + 1) \times D + 2 \times h \times P \\ &\quad + 2 \times (h + 1) \times D + (h + 1) \times X. \end{aligned} \tag{4}$$

If the first backup path does not traverse k, the number of hops of the first backup path is h, and the protection-switching time t_r is calculated in

$$\begin{aligned} t_r &= F + n_1 \times P + (n_1 + 1) \times D + 2 \times h \times P \\ &\quad + 2 \times (h + 1) \times D + (h + 1) \times X. \end{aligned} \tag{5}$$

Now, we introduce some performance parameters.

Balance degree (BD) is calculated in

$$\text{BD} = |L| \max_{k \in L}\{a_k\} / \sum_{k \in L} a_k. \tag{6}$$

BD is close to 1 that means the algorithm is more favorable for the load balance.

Resource utilization ratio (RUR) is calculated in

$$\text{RUR} = \sum_{k \in L} p_{1k} + p_{2k} / \sum_{k \in L} w_k. \tag{7}$$

It is obvious that a smaller value of RUR means a higher resource utilization ratio.

Blocking ratio (BR) and dropping ratio (DR) can be defined according to Eqs. (6) and (3), respectively.

Average protection-switching time (APST) is calculated in

$$\text{APST} = \sum_{k \in M} t_k / |M|, \tag{8}$$

where M is the set of the sub-paths of the connections that hold on the network, where the sub-paths traverse the failed links.

We simulate a dynamic network environment with the assumptions that connection requests arrival according to an independent Poisson process with arrival rate β, and the connections holding time is negative exponentially distributed $1/\mu$. We assume $\mu = 1$ and each request bandwidth is a wavelength. We assume that there are no waiting queues in the network, so if a connection request is blocked, it does be abandoned immediately. The test network is shown in Fig. 1[6], each node-pair is interconnected by a bi-directional fiber link that has ten wavelengths. We assume the basic link-cost is 100, that is, the length of the sub-path can be equal to 1, 2, 3, and ML respectively, where ML is the full length of the primary path. In fact, SPP[4] is a special example for SSPP (when m = ML, $\alpha = 0$, and $\eta = 0$). All results are averaged via simulating 10^6 connections.

In Fig. 2 we can observe that BD is big for $\alpha = 0$ and reduces when α increases under the same m. The reason is that there is no consideration about the load

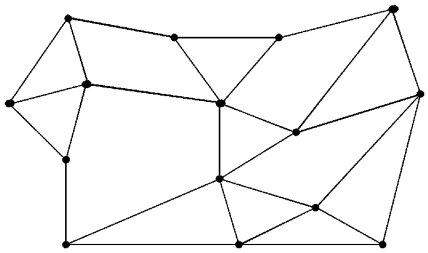

Fig. 1. Network topology of USA.

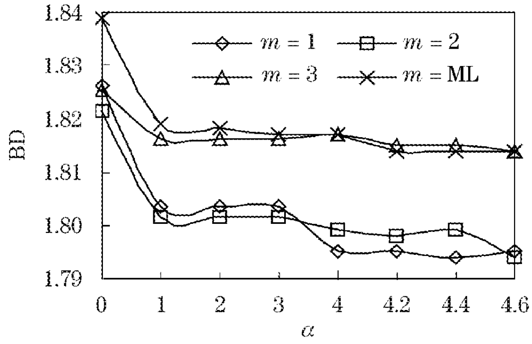

Fig. 2. BD versus α (load=10).

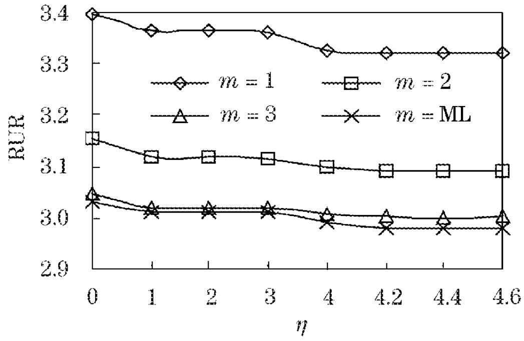

Fig. 3. RUR versus η (load=10).

balance according to Eq. (1) when $\alpha = 0$. When α increases, the links, which have more residual resources, have less link-cost. Then, the primary paths are favorable for traversing these links, and the consumed resources are more uniformly distributed to the network. Thus, the load is more balance.

Figure 3 shows that the RUR reduces when η increases under the same m. Because when η increases, Eqs. (2) and (3) can adjust the link-cost, and the links, which already have enough reserved resources (namely $p_{1l} + p_{2l} \geq tp_{1l} + tp_{2l}$), have less link-cost. Then, the backup paths are favorable for traversing these links, namely they are favorable for selecting the least additional backup bandwidths reserved for paths. Thus, the resource utilization ratio is improved.

According to above analysis, SSPP can improve the performances of BD and BUR with a proper (α, η). Through testing the actual network (Fig. 1), we find that $(\alpha, \eta) = (3, 1)$, in comparison with $(\alpha, \eta) = (0, 0)$, can produce a better performance.

By simulating two random failed links, we can observe that the DRs are always equal to zero with the same load in Fig. 4, and this means that SSPP can completely protect the double-link failures.

Figure 5 shows that the performance of RUR of SSPP $(m, 3, 1)$ is better than SSPP $(m, 0, 0)$ with the same load. Under the same (α, η), the RUR reduces with m increasing.

Figure 6 shows that the performance of BR of SSPP $(m, 3, 1)$ is better than SSPP $(m, 0, 0)$ with the same load. Because the performance of RUR of SSPP $(m, 3, 1)$ is better, more spare resources can be used by the following connections, and this leads to lower BR. Under the same (α, η), the BR reduces with m increasing, and the reason is that RUR reduces with m increasing.

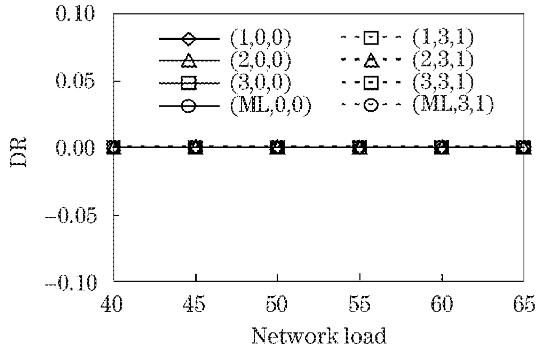

Fig. 4. DR versus network load.

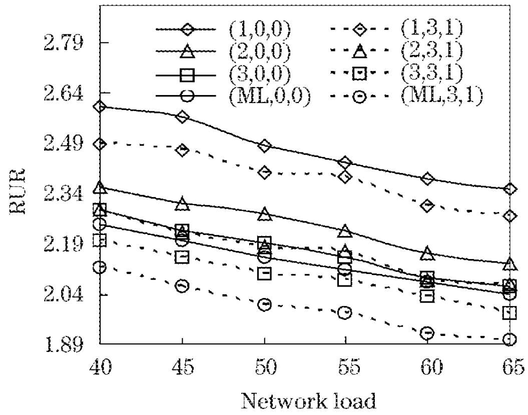

Fig. 5. RUR versus network load.

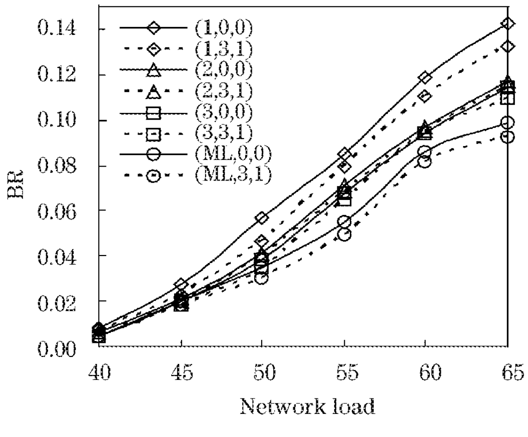

Fig. 6. BR versus network load.

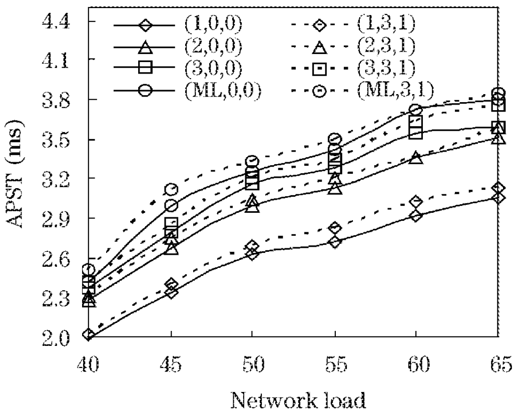

Fig. 7. APST versus network load.

Figure 7 shows that SSPP $(m,0,0)$ has smaller APST than SSPP $(m,3,1)$ with the same load. Because the paths of SSPP $(m,0,0)$ are minimum hop while the paths of SSPP $(m,3,1)$ are minimum cost, and generally the hops of paths with minimum cost are longer, and this leads n_1, n_2, and h in Eqs. (4) and (5) to increase, and then APST becomes long. Under the same (α, η), APST increases with m increasing. Because the length of backup path increases with m increasing, and then n_1, n_2, and h in Eqs. (4) and (5) increase, and this leads APST to become big.

In conclusion, SSPP has a better performance than SPP[4] (a special example for SSPP (ML, 0, 0)) via configuring the different (m, α, η). SSPP not only can completely protect the double-link failures but also can make the tradeoffs between the resource utilization ratio (or blocking ratio) and the protection-switching time.

This work was supported by the National Natural Science Foundation of China under Grant No. 60302010. L. Guo's email is haveball@263.net.

References

1. H. B. Wen, L. M. Li, R. X. He, H. F. Yu, S. Wang, and N. Song, Photon. Network. Commun. **6,** 253 (2003).
2. F. Q. Liu, Q. J. Zeng, X. Zhu, and S. L. Xiao, Chin. Opt. Lett. **1,** 259 (2003).
3. S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, J. Lightwave Technol. **21,** 870 (2003).
4. B. G. Jozsa, D. Orincsay, and A. Kern, in *Proceedings of IEEE ISCC'2003* 1333 (2003).
5. Y. Wang, L. M. Li, and S. Wang, Acta Electron. Sin. **29,** 1319 (2001).
6. L. Guo, H. F. Yu, and L. M. Li, Opt. Express **12,** 2327 (2004).