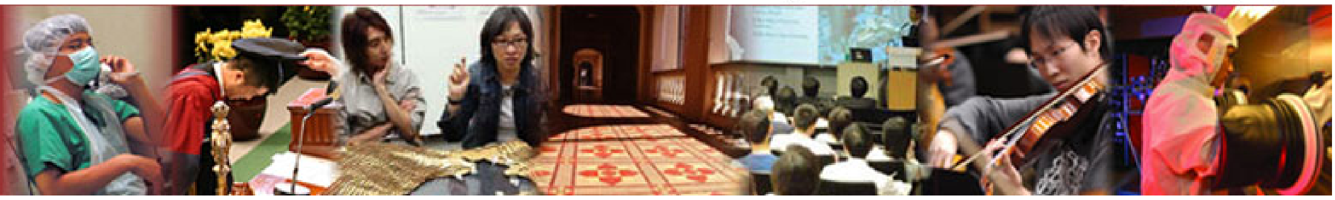


Title	Super monitor design for fast link failure localization in all-optical networks
Author(s)	Mao, M; Yeung, KL
Citation	The 2011 IEEE International Conference on Communications (ICC 2011), Kyoto, Japan, 5-9 June 2011. In Proceedings of the IEEE ICC, 2011, p. 1-5
Issued Date	2011
URL	http://hdl.handle.net/10722/140251
Rights	Creative Commons: Attribution 3.0 Hong Kong License

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2011 proceedings

Super Monitor Design for Fast Link Failure Localization in All-Optical Networks

Minjing Mao and Kwan L. Yeung
Department of Electrical and Electronic Engineering
The University of Hong Kong
Pokfulam, Hong Kong
{mjmao, kyeung}@eee.hku.hk

Abstract— **An m-cycle is an optical loop-back pre-cross-connection of a supervisory wavelength. In a cycle-based link failure detection scheme, a monitor transmits supervisory signals onto the m-cycle, receives them back, and compares with the one sent for fault detection. In this paper, we propose the notion of super monitor for cutting down the hardware cost of monitors. Instead of having a dedicated monitor for each m-cycle, a super monitor is placed at the junction of a set of m-cycles. Supervisory signals from a single laser source are split simultaneously onto multiple m-cycles using an optical splitter. As the cost of a monitor is usually dominated by the laser, co-locating conventional monitors to form a super monitor for cost reduction makes sense. To this end, we formulate the problem of determining the optimal number of super monitors as well as their locations as an add-on feature of any existing cycle-based link failure detection scheme. We call it monitor placement problem. We follow a two-step approach for its solution, where in the first step, we enumerate each candidate cycle-set (i.e., a set of m-cycles where the super monitor can be placed); and in the second step, a simple integer linear programming (ILP) is constructed for placing super monitors at some candidate cycle-sets. Numerical results show that by properly placing super monitors, considerable amount of monitoring cost can be saved.**

Keywords- link failure detection and localization; monitoring cycle (m-cycle); super monitor; all-optical networks.

I. INTRODUCTION

Wavelength Division Multiplexing (WDM) technology enables cost-efficient data communications in all-optical networks, by multiplexing hundreds of high-speed wavelength channels onto a single fibre for parallel transmission. Nevertheless, this makes the network more vulnerable to component failures, especially link failures. Since the per-channel data rate on a fibre can reach up to 40 Gbps, a single link failure (e.g. due to a fiber cut) can result in enormous amount of data loss. To minimize the data loss, a fast link failure detection scheme is essential in shortening the amount of time required to recover from the failure.

Various fast link failure detection schemes have been proposed [1]-[10]. They can be implemented at different protocol layers independently, or co-operatively in a cross-layer manner [1]. Compared with the optical/physical layer schemes, those at the upper layers (such as the fault management mechanisms in some routing protocols like IS-IS and OSPF [2]) need more signaling and thus require a relatively longer detection time. As a result, optical/physical layer schemes are preferred in designing survivable all-optical networks.

Fault detection at optical layer usually involves optical power measurement, optical spectral analysis, pilot tones and/or optical time domain reflectometry (OTDR) [11]–[12]. They are carried out by a device called *monitor* [12], which monitors the health of a certain part of the network. When a fault occurs, the associated monitors will be triggered. The generated alarms are then collected and analyzed by a traffic management center. Once a fault is detected and localized, the recovery mechanism will re-route the affected traffic to bypass the fault.

There are different types of optical layer link failure detection schemes. A link-based scheme requires one monitor for each link. When a fault occurs, only one monitor will be triggered. But the number of monitors required is equal to the number of links in the network. To reduce the number of monitors, the concept of *monitoring cycle* (m-cycle) was introduced in [3]-[4]. An m-cycle is an optical loop-back pre-cross-connection of a supervisory wavelength with a dedicated monitor. The length of an m-cycle is the number of links it traverses. The m-cycle in Fig. 1 has a length of 4. A monitor can be placed at any node on an m-cycle. It consists of a pair of optical transmitter and receiver (i.e., a laser diode and a photodiode) and an electrical controller. When the received optical power is below a pre-defined threshold, a link failure (somewhere along the cycle) is detected and the monitor sends an alarm to the traffic management center (typically via a separated control network).

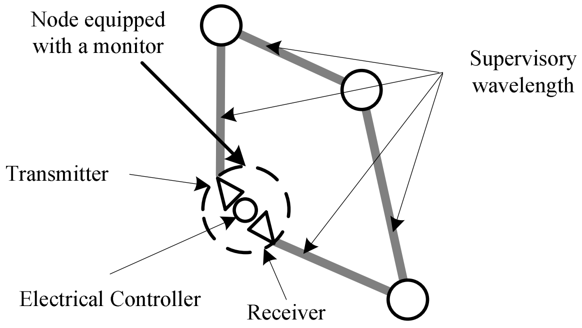
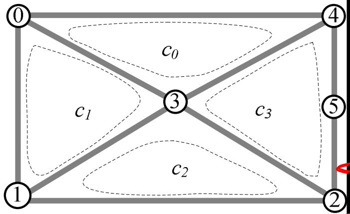


Figure 1. An example of m-cycle with a simple monitor.

978-1-61284-233-2/11/$26.00 ©2011 IEEE

In a cycle-based link failure detection scheme, each link in the network must be covered by at least one m-cycle. Once a link failure occurs, all m-cycles/monitors that traverse the failed link will alarm. Assume no two or more link failures occur at the same time, i.e. the single link failure model. Each link failure corresponds to an alarm code, with the format of $[\alpha_{n-1},...,\alpha_1,\alpha_0]$, where $\alpha_j = 1$ means that the monitor of m-cycle c$_j$ alarms and $\alpha_j = 0$ otherwise. From Fig. 2, we can see that the alarm code for the failure at link (0, 1) is [0, 0, 1, 0]. Notably, a cycle-based scheme cannot distinguish the failures at the links that form a "chain", e.g. (4, 5) and (2, 5) in Fig. 2(a). This is because the same set of m-cycles traverse through both links and will produce the same alarm code of [1, 0, 0, 0]. In such a case, additional link-based monitors must be used for uniquely identify each link failure.



Link	c_3	c_2	c_1	c_0	Decimal
(0, 1)	0	0	1	0	2
(0, 3)	0	0	1	1	3
(0, 4)	0	0	0	1	1
(1, 2)	0	1	0	0	4
(1, 3)	0	1	1	0	6
(2, 3)	1	1	0	0	12
(2, 5)	1	0	0	0	8
(3, 4)	1	0	0	1	9
(4, 5)	1	0	0	0	8

(a) Generated m-cycles (b) Alarm code set

Figure 2. An m-cycle solution/example from [6].

A cycle-based detection scheme [3]-[7] aims at minimizing the monitoring cost while ensuring 100% single link failure detection. Monitoring cost usually includes three parts: hardware cost of the monitor, management/signaling cost for error reporting, and bandwidth cost for carrying the supervisory signals. In order to minimize the hardware cost, the conventional approach is to minimize the number of monitors required. Since each m-cycle has a dedicated monitor, minimizing hardware cost becomes minimizing the number of m-cycles. When the number of monitors is reduced, the associated error reporting effort, i.e. the management cost, is also reduced. Nevertheless, the bandwidth cost will be increased because length of each m-cycle as well as the total length of all m-cycles will be increased. In [7], efficient ILPs have been formulated for finding a set of m-cycles with minimum monitoring cost.

In this paper, we propose the notion of *super monitor* for further cutting down the monitoring cost. Instead of having a dedicated monitor for each m-cycle, we can place a super monitor at a junction of a set of m-cycles. Consider two m-cycles c_0 and c_2 in Fig. 2(a). We can place a super monitor at node 3. A super monitor only requires a single laser for simultaneously transmitting supervisory signals onto multiple m-cycles. This is achieved by splitting the power from a single laser using an inexpensive optical splitter, as shown in Fig. 3. Note that we still require a dedicated receiver for each m-cycle, but it is less expensive than a laser. Alternatively, we can have a single receiver detecting the signals from all m-cycles in a time-multiplexed fashion. But this will slightly increase the fault detection time.

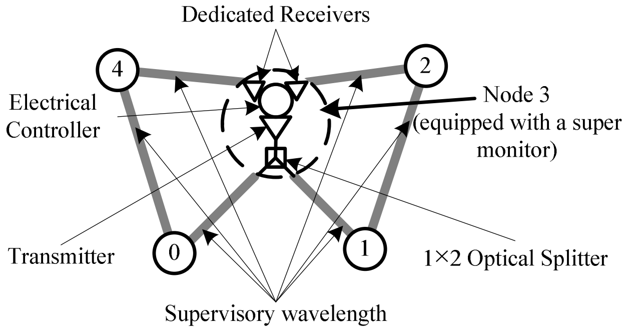


Figure 3. An example super monitor at the junction of two m-cycles.

Based on the set of m-cycles found using a cycle-based scheme, we need to decide how many super monitors are required and where to place them. We call it monitor placement problem. Finding the optimal solution involves integer and combinatorial optimization. In Section II, we formally formulate the monitor placement problem and propose a two-step approach for its solution. In Section III, the effectiveness of using super monitors is studied. We conclude the paper in Section IV.

II. MONITOR PLACEMENT PROBLEM

A. *Hardware cost of monitors*

A super monitor consists of a laser/transmitter, an optical splitter, n optical receivers, and an electrical controller. As the cost of optical splitter and electrical controller is far less than laser and receiver, the cost of a super monitor can be represented by $T+nR$, where T and R are the costs of a laser and a receiver, respectively. Then a conventional monitor has a cost of $T+R$.

If a super monitor is not used, then n conventional monitors are required. The cost reduction is $(n-1)T$. Based on the market price of optical devices, a laser diode is generally twice of a photodiode. Therefore, the notion of super monitor can give a significant saving on the hardware cost. As we have pointed out before, there is also a big saving on the management cost because management cost is proportional to the number of monitors, no matter super or not.

B. *Super Monitor Location*

A set of m-cycles can share a super monitor only if they have at least one common node but no common link. Common link should be prohibited because the optical signals must be split (again) at the end of a common link – this requires not only an additional splitter, but also extra co-ordination effort between the monitor and the additional splitter.

We define the set of m-cycles that satisfy the above requirement as a candidate cycle-set (CCS), such as the set of m-cycles c_0 and c_2 in Fig. 2(a). A super monitor for this CCS can be placed at the common node 3.

C. *Two-step Approach*

For a given set of m-cycles found using an existing cycle-based scheme, the monitor placement problem is to find the number of super monitors required as well as their best locations such that the total hardware cost is minimized. We focus on hardware cost because the bandwidth cost is fixed once the set of m-cycles are given. Besides, the management cost is reduced with the number (as well as the size) of super monitors.

We follow a two-step approach to solve the monitor placement problem. Without loss of generality, we denote the given set of m-cycles by M. Accordingly, any candidate cycle-set (CCS) for a super monitor must be a subset of M.

1) Candidate Cycle-Sets Enumeration:

Assume the size of M is m. Then the size of each candidate cycle-set cannot be larger than m. Our first step is to find out all the possible candidate cycle-sets. The enumeration algorithm used is summarized by the pseudo code below.

```
Initialize the queue of candidate cycle-sets (CCS_SET) to be empty. The size
of a CCS is initialized to be two, i.e., size = 2.
while(size ≤ m)
    for(any set of size m-cycles)
        if(the structure of a specific set of size m-cycles satisfies a CCS)
            Mark these size m-cycles as a qualified CCS and insert it
            into CCS_SET.
        end if
    end for
    if(no set of size m-cycles is marked as a CCS)
        return CCS_SET;
    else
        size++;
    end if
end while
return CCS_SET;
```

2) ILP Formulation:

The following notations are adopted in our ILP formulation:

M : The set of input m-cycles (generated by the ILP [7]).

S : The set of all possible candidate cycle-sets found in the first step.

r_{ij} : Binary variable, where 1 m-cycle i is in CCS j, 0 otherwise.

n_j : Number of m-cycles in CCS j.

cs_j : Binary variable, where 1 means a super monitor is to be placed at the common node in CCS j, 0 otherwise.

Our ILP for solving the monitor placement problem is formulated below.

Objective:

$$Maximize \sum_{j=0}^{N-1} cs_j \cdot (n_j - 1) \quad (1)$$

Constraint:

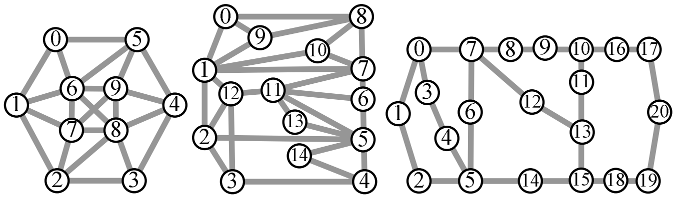
$$\sum_{j=0}^{N-1} r_{ij} \cdot cs_j \le 1, \forall i \in M \quad (2)$$

The objective is to minimize the total hardware cost, which is the same as maximizing the saving in hardware cost. Note that one super monitor can cut down the hardware cost by $(n-1)T$. Assume that there are N super monitors in total. The number of receivers in each super monitor is n_j (j=0,1,…,N-1), respectively. The hardware cost saving can be formulated by $\sum_{j=0}^{N-1}(n_j - 1)T$. Since the hardware cost saving is proportional to $\sum_{j=0}^{N-1}(n_j - 1)$, we use formula (1) to describe the objective of our ILP. In addition, one m-cycle is related with only one monitor (either a simple monitor or super monitor), so we use constraint (2) to guarantee that each m-cycle is in at most one selected CCS.

III. NUMERICAL RESULTS AND DISCUSSION

Many cycle-based link failure detection schemes can be found [3]-[7]. Among them, HDFS [3], SPEM [3], HST [4] and M^2-CYCLE [6] are heuristics. They are efficient but cannot guarantee optimal performance. Notably, the ILP in [7] considers both simple and non-simple m-cycles and is capable of providing the lowest monitoring cost under the framework defined in [7].

In this section, we choose to use the set of m-cycles obtained by solving the ILP in [7]. Then we use it as the input to our enumeration algorithm for candidate cycle-sets. Numerical results based on three benchmark networks [7] are obtained, namely, SmallNet in Fig. 4(a) (10 nodes and 22 links), BellCore in Fig. 4(b) (15 nodes and 28 links), and ARPA2 in Fig. 4(c) (21 nodes and 25 links). Without loss of generality, we assume that the number of wavelengths available on each link is 30. In order to make sure each link has sufficient wavelengths to carry user data, we have modified the ILP in [7] by only allowing up to 3 wavelengths per link for carrying m-cycles. In other words, each link has at least 27 wavelengths for user data. Numerical results are found with a Due-Core 2.66 GHz Windows PC with 2G bytes of memory, where the algorithm for candidate cycle-set enumeration is implemented in Visual Studio 2010 and our ILP is solved by CPLEX 11.0.



(a) SmallNet (b) BellCore (c) ARPA2

Figure 4. Three networks for simulation.

Fig. 5 shows the set of m-cycles found using the modified ILP in [7]. To understand the notations used in Fig. 5, let us consider Fig. 5(a) for SmallNet. Each entry in the table denotes if a particular m-cycle passes through a link ("1") or not ("0"). We can see that SmallNet requires 9 m-cycles. For BellCore and APAR2, 11 and 5 m-cycles are required respectively.

Then we use the set of m-cycles in Fig. 5 as the input to our two-step algorithm for solving the monitor placement problem. For simplicity, we do not list out the candidate cycle-sets found using the enumeration algorithm. Instead, Tables I-III directly provide the final monitor placement solutions, namely, the number of (super) monitors required (i.e. column Monitor Index), the type of each monitor (Monitor Type) used, the chosen candidate cycle-set or a single m-cycle that is served by the particular monitor (Chosen CCS/m-cycle), and the monitor location (Location).

Tables IV and V show the mornitoring hardware cost and management costs with and without using super monitors. Table VI gives the corresponding saving in percentage. From the three tables, it is interesting to notice that although our monitor placement problem aims at minimizing hardware cost, the actual percentage saving on management cost is more pronounced than hardware cost.

Link	c_8	c_7	c_6	c_5	c_4	c_3	c_2	c_1	c_0
(0, 1)	1	0	0	0	0	0	0	0	0
(0, 5)	1	0	0	0	1	0	0	0	0
(0, 6)	0	0	0	0	1	0	0	0	0
(1, 2)	0	0	0	1	0	0	0	0	0
(1, 6)	1	0	0	0	0	1	0	0	0
(1, 7)	0	0	0	1	0	1	0	0	0
(2, 3)	0	1	0	0	0	0	0	0	0
(2, 7)	0	0	0	1	0	0	0	1	0
(2, 8)	0	1	0	0	0	0	0	1	0
(3, 4)	0	0	0	0	0	0	0	0	1
(3, 8)	0	1	0	0	0	0	0	0	1
(4, 5)	0	0	1	0	0	0	0	1	0
(4, 8)	0	0	0	0	0	0	0	1	1
(4, 9)	0	0	1	0	0	0	0	0	0
(5, 6)	1	0	0	0	0	0	0	1	0
(5, 9)	0	0	1	0	1	0	0	0	0
(6, 7)	0	0	0	0	0	0	1	0	0
(6, 8)	0	0	0	0	0	1	0	1	0
(6, 9)	0	0	0	0	1	0	1	0	0
(7, 8)	0	0	0	0	0	1	0	0	0
(7, 9)	0	0	0	0	0	0	1	1	0
(8, 9)	0	0	0	0	0	0	0	1	0

(a) SmallNet

Link	c_{10}	c_9	c_8	c_7	c_6	c_5	c_4	c_3	c_2	c_1	c_0
(0, 1)	1	0	0	0	0	1	0	0	0	0	0
(0, 8)	0	1	0	0	0	1	0	0	0	0	0
(0, 9)	1	1	0	0	0	0	0	0	0	0	0
(1, 2)	0	0	0	1	0	0	0	0	0	0	0
(1, 7)	0	0	1	0	0	1	0	0	0	0	0
(1, 9)	1	0	1	0	0	0	0	0	0	0	0
(1, 10)	0	0	0	0	0	0	0	0	1	0	0
(1, 12)	0	0	0	1	0	0	0	0	1	0	0
(2, 3)	0	0	0	0	0	0	0	1	0	1	0
(2, 5)	0	0	0	0	0	0	0	1	0	1	1
(2, 12)	0	0	0	1	0	0	0	0	0	0	1
(3, 4)	0	0	0	0	0	0	0	0	0	1	1
(3, 12)	0	0	0	0	0	0	0	1	0	0	1
(4, 5)	0	0	0	0	0	0	0	0	0	1	0
(4, 14)	0	0	0	0	0	0	0	0	0	0	1
(5, 6)	0	0	0	0	1	0	0	1	0	0	0
(5, 11)	0	0	0	0	1	0	1	0	0	0	0
(5, 13)	0	0	0	0	0	0	1	0	0	0	0
(5, 14)	0	0	0	0	0	0	0	0	0	0	1
(6, 7)	0	0	0	0	1	0	0	0	0	0	0
(6, 11)	0	0	0	0	0	0	0	1	0	0	0
(7, 8)	0	0	0	0	0	1	0	0	0	0	0
(7, 10)	0	0	1	0	0	0	0	0	1	0	0
(7, 11)	0	0	0	0	1	0	0	0	1	0	0
(8, 9)	0	1	1	0	0	0	0	0	0	0	0
(8, 10)	0	0	1	0	0	0	0	0	0	0	0
(11, 12)	0	0	0	0	0	0	0	1	1	0	0
(11, 13)	0	0	0	0	0	0	1	0	0	0	0

(b) BellCore

Links	c_4	c_3	c_2	c_1	c_0
(0, 1), (1, 2), (2, 5)	1	0	0	0	0
(0, 3), (3, 4), (4, 5)	1	1	0	0	0
(0, 7)	0	1	0	0	0
(5, 6), (6, 7)	0	1	0	0	1
(5, 14), (14, 15)	0	0	0	0	1
(7, 8), (8, 9), (9, 10)	0	0	1	0	0
(7, 12), (12, 13)	0	0	1	0	1
(10, 11), (11, 13)	0	0	1	1	0
(10, 16), (16, 17), (17, 20) (19, 20), (18, 19), (15, 18)	0	0	0	1	0
(13, 15)	0	0	0	1	1

(c) ARPA2

Figure 5. M-cycles obtained using the ILP in [7] for the three example. Networks in Fig. 4.

TABLE I. SOLUTION FOR ARPA2

Monitor Index	**Monitor Type**	**Chosen CCS/ m-cycle**	**Location**
1	Super Monitor	c_0, c_4	Node 5
2	Super Monitor	c_2, c_3	Node 7
3	Simple Monitor	c_1	Node 10 or 11 or13 or 15 or 16 or 17 or 18 or 19 or 20

TABLE II. SOLUTION FOR SMALLNET

Monitor Index	**Monitor Type**	**Chosen CCS/ m-cycle**	**Location**
1	Super Monitor	c_0, c_3	Node 8
2	Super Monitor	c_1, c_4	Node 5 or 6 or 9
3	Super Monitor	c_2, c_6	Node 9
4	Super Monitor	c_5, c_7	Node 2
5	Simple Monitor	c_8	Node 0 or 1 or 5 or 6

TABLE III. SOLUTION FOR BELLCORE

Monitor Index	**Monitor Type**	**Chosen CCS/ m-cycle**	**Location**
1	Super Monitor	c_0, c_6	Node 5
2	Super Monitor	c_2, c_{10}	Node 1
3	Super Monitor	c_3, c_4	Node 5 or 11
4	Super Monitor	c_7, c_8	Node 1
5	Simple Monitor	c_1	Node 2 or 3 or 4 or 5
6	Simple Monitor	c_5	Node 0 or 1 or 7 or 8
7	Simple Monitor	c_9	Node 0 or 8 or 9

TABLE IV. HARDWARE COST

Scheme	**Hardware cost**		
	SmallNet	BellCore	ARPA2
Two-step Approach	19	25	11
ILP in [7]	27	33	15

TABLE V. MANAGEMENT COST

Scheme	**Management cost**		
	SmallNet	BellCore	ARPA2
Two-step Approach	5	7	3
ILP in [7]	9	11	5

TABLE VI. PERCENTAGE IMPROVEMENT OVER THE ILP IN [7]

Cost	**Percentage improvement (%)**		
	SmallNet	BellCore	ARPA2
Hardware Cost	29.6	24.2	26.7
Management Cost	44.4	36.4	40.0

IV. CONCLUSION

We proposed the notion of super monitor for cutting down the hardware cost involved in a cycle-based link failure detection scheme. Instead of having a dedicated monitor for each m-cycle, a super monitor is placed at the junction of a set of m-cycles with at least one common node but no common link. Supervisory signals from a single laser source are split simultaneously onto multiple m-cycles using an optical splitter. We then formulated the problem of finding the optimal number of super monitors as well as their locations as the monitor placement problem. A two-step approach was then adopted for finding its solution. Numerical results showed that by properly placing super monitors, both hardware and management costs required for link failure detection can be greatly reduced.

REFERENCES

[1] C. Assi, Y. Ye, A. Shami, S. Dixit, and M. Ali, “A hybrid distributed fault-management protocol for combating single-fiber failures in mesh-based DWDM optical networks,” in *Proc. IEEE GLOBECOM’02*, vol. 3, pp. 2676–2680, Nov. 2002.

[2] M. Goyal, K. K. Ramakrishnan, and W.-C. Feng, “Achieving faster failure detection in OSPF networks,” in *Proc. IEEE ICC’03*, vol. 1, pp. 296–300, May 2003.

[3] H. Zeng, C. Huang, A. Vukovic, and M. Savoie, “Fault detection and path performance monitoring in meshed all-optical networks,” in *Proc. IEEE GLOBECOM ’04*, vol. 3, pp. 2014–2018, Dec. 2004.

[4] H. Zeng, C. Huang and A. Vukovic, “Spanning tree based monitoringcycle construction for fault detection and localization in mesh AONs,” in *Proc. IEEE ICC’05*, vol. 3, pp. 1726-1730, May 2005.

[5] Y.G. Wen, V. W. S. Chan and L. Z. Zheng, “Efficient fault diagnosis algorithms for all-optical WDM networks with probabilistic link Failures (*invited paper*),” *IEEE/OSA Journal of Lightwave Technology*, vol. 23, no. 10, pp. 3358-3371, Oct. 2005.

[6] B. Wu and K. L. Yeung, “M^2-CYCLE: an optical layer algorithm for fast link failure detection in all-optical mesh networks,” in *Proc. IEEE GLOBECOM'06*, pp. 1–5, Dec. 2006.

[7] B. Wu, K. L. Yeung and P.-H. Ho, “Monitoring cycle design for fast link failure localization in all-optical networks,” *IEEE/OSA Journal of Lightwave Technology*, vol. 27, no. 10, pp. 1392–1401, May 2009.

[8] B. Wu, P.-H. Ho and K. L. Yeung, “Monitoring trail: on fast link failure localization in WDM mesh networks,” *IEEE/OSA Journal of Lightwave Technology*, vol. 27, no. 18, pp. 4175-4185, Sept. 2009.

[9] B. Wu, Pin-Han Ho, J. Tapolcai, P. Babarczi, “Optimal Allocation of Monitoring Trails for Fast SRLG Failure Localization in All-Optical Networks,” in *Proc. IEEE GLOBECOM 2010*, to appear.

[10] S. Ahuja, S. Ramasubramanian and M. Krunz, “SRLG failure localization in all-optical networks using monitoring cycles and paths,” in *Proc. IEEE INFOCOM '08*, pp. 700-708, Apr. 2008.

[11] R. Ramaswami and K. N. Sivarajan, *Optical networks: a practical perspective*, 2nd ed. San Francisco, Calif.: Morgan Kaufmann, 2002.

[12] S. Stanic, S. Subramaniam, H. Choi, G. Sahin and H.-A. Choi, “On monitoring transparent optical networks,” in *Proc. Int. Conf. on Parallel Process. Workshops*, pp. 217-223, Aug. 2002.