

# M-Burst: A Framework of SRLG Failure Localization in All-Optical Networks

Mohammed L. Ali, Pin-Han Ho, János Tapolcai, and Basem Shihada

**Abstract**—Fast and unambiguous failure localization for shared risk link groups (SRLGs) with multiple links is essential to build fully survival and functional transparent all-optical mesh networks. Monitoring trails (m-trails) has been proposed as an effective approach to achieve this goal. However, each m-trail traverses through each link by constantly taking a wavelength channel (WL), which causes a significant amount of resource consumption. In this paper, a novel framework of all-optical monitoring for SRLG failure localization is proposed. We investigate the feasibility of periodically launching *optical bursts* along each m-trail instead of supervisory lightpath to probe the set of fiber segments along the m-trail, aiming to achieve a graceful compromise between resource consumption and failure localization latency. The paper will firstly define the proposed framework and highlight the relevant issues regarding its feasibility. We provide theoretical justifications of the scheme. As a proof of concept, the paper formulates the optimal burst scheduling problem via an Integer Linear Program (ILP) and implements the method in networks of all possible SRLGs with up to  $d = 3$  links. A heuristic method is also proposed and implemented for multiple link SRLG failure localization keeping all the assumptions same as the ILP method. Numerical results for small networks show that the scheme is able to localize single and multiple link SRLG failures unambiguously with very small amount of failure localization latency.

**Index Terms**—Shared risk link groups (SRLGs), m-trail, unambiguous failure localization.

## I. INTRODUCTION

Monitoring trail (m-trail) has been reported as an effective approach for all-optical failure localization via out-of-band monitoring [1]-[11]. In the past, an m-trail was implemented as a supervisory lightpath (S-LP) launched with constant optical signal, in which a monitoring node (MN) that terminates the m-trail can obtain the failure status of the group of links traversed by the S-LP. An m-trail solution is composed of a set of m-trails, and is said to achieve *local unambiguous failure localization* (L-UFL) [10] at a particular monitoring node (MN) if the MN can localize any SRLG failure by inspecting the locally available on-off status of the m-trails.

To achieve solutions with the number of transmitters/monitors in the order of logarithm to the number of SRLGs, out-of-band monitoring via simple/non-simple m-cycles and m-trails were investigated in [7][8][9]. These studies take the S-LP allocation problem as a topology coding process by assigning each SRLG a unique code, where 1

is assigned to the  $j$ -th bit of the code for an SRLG if the  $j$ -th m-trails traverses through the SRLG, and 0 otherwise. When a SRLG fails which interrupts each m-trail traversing through the SRLG, the receiver of each interrupted m-trail will issue an alarm. By collecting all the alarms issued, the network controller or routing entities are expected to come up with a valid alarm code where the  $j$ -th bit position is 1 for every  $j$ -th S-LP affected by the failure, while the other bit position is 0 since the corresponding m-trails are link-disjoint from the failed SRLG. The alarm code is read and uniquely indicates an SRLG failure event. Note that the alarm dissemination should take place in the network control plane real-time right after the occurrence of failure, which not only takes a significant amount of network resources of the control plane during fault events, but also introduces additional failure localization latency and control complexity.

Some studies suggested to monitor a set of m-trails terminated at a single monitoring node (MN) which can completely remove the alarm dissemination complexity [5][6][10]. In particular, the studies in [10] further suggested that a node can obtain the on-off status of a lightpath by tapping the optical signal, by which the failure localization decision can be made locally at the node. Although with the signaling complexity resolved, every link needs to consume a single wavelength channel (WL) to support the traversal of an m-trail, which leads to significant increase of the resource consumption.

To further reduce monitoring resource consumption, some research suggested to implement the concept of m-trails using optical probes instead of static supervisory lightpaths [12][13], where a control layer protocol is used to inspect each fiber segment along an m-trail via a "permissible probe". If there is no faulty link along the m-trail, the permissible probe will arrive at the designated receiver, and will not appear otherwise. Thus the failure localization decision is made by concluding the inspection results on a set of m-trails. Although effective in reducing the monitoring resource consumption, it causes longer failure localization latency and additional operation overhead.

Monitoring bursts (m-bursts) is a framework of failure localization in all-optical mesh networks firstly introduced in [14]. A single MN is traversed by a set of m-trails in a form of cycle, or called m-cycles. The MN launches optical bursts along the m-cycles to detect the on-off status of the m-cycles. Due to close-loop shape of an m-cycle, the launched optical bursts will be received by the MN in case all the links along the m-cycle are working properly, but the burst will be lost if any link along the m-cycle has failed. The same as the conventional m-trail approach, the MN designates a set of m-cycles each

Mohammed L. Ali and Pin-Han Ho are with University of Waterloo, Waterloo, ON, N2L 3G1 Canada (e-mail: {mlali,p4ho}@uwaterloo.ca).

János Tapolcai is with Budapest University of Technology and Economics (BME), Budapest 1117, Hungary (e-mail: tapolcai@tmit.bme.hu).

Basem Shihada is with King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia (e-mail: basem.shihada@kaust.edu.sa)

providing a single bit in the alarm code based on whether the corresponding burst is received at the expected time instant or not. Since each m-burst is a short-duration optical flow, we no longer need to statically reserve a whole WL along each link of an m-cycle. Multiple optical bursts can be multiplexed in the time domain on the same WL, which significantly reduces the consumed monitoring resources. As a result, each link may consume as few as a single WL for fault localization to support multiple m-cycles traversing through the link. This is possible if provided with a proper burst scheduling mechanism at the MN, which manipulates the launching time of each optical burst such that burst collision at any link can be completely avoided. However, the study in [14] is rather a preliminary one; the focus is on the single link failure localization, and the formulated problem is solved using only integer linear programs (ILPs).

The paper extends [14] by providing a complete study on the m-burst framework under multi-link SRLGs. We define the m-burst scheduling problem, and develop comprehensive analysis for gaining deeper understanding on the m-trail allocation problem with a single MN. Given the maximum number of simultaneous link faults  $d$  that can possibly occur in the network, the proposed approach finds a set of m-trails to identify any SRLG failure and meanwhile determines the instants of launching m-bursts in order to avoid burst collision along any link of the network.

The rest of the paper is organized as follows. Section II presents the related work on multi-link SRLG failure localization in all-optical networks. Problem formulation is given in Section III. Theoretical analysis of the multi-link SRLG failure localization algorithm is in Section IV. In Section V, an ILP is formulated to solve the proposed m-trail allocation and burst scheduling problem. In Section VI, a heuristic algorithm to localize multiple link SRLG failures is provided. Numerical experiments were conducted and the results are given in Section VII. Section VIII concludes the paper.

## II. BACKGROUND

### A. Failure Localization in GMPLS

Fault management is defined under the control framework of generalized multiprotocol label switching (GMPLS) [15] as a set of real-time tasks performed sequentially right after the occurrence of any failure, including failure localization, notification, and correlation. The former two tasks (i.e., localization and notification) in the optical network domain are defined via a series of electronic signaling mechanisms. By applying Link Management Protocol (LMP-te) coupled with a signaling protocol such as RSVP-TE, each downstream node of a failed working lightpath is subject to loss of light (LOL) and will send an alarm to its upstream node. After receiving the alarm, the upstream node checks the corresponding input port and forwards the alarm to further upstream if the node is also subject to LOL. Otherwise, the faulty link is determined in the downstream, and the upstream node initiates protection/restoration procedures. Guo and Kuo [16] proposed an enhanced scheme to expedite the fault localization process by having the node that sensed the LOL to send alarms to both upstream and downstream nodes.

The aforementioned GMPLS based approaches are subject to many weaknesses. Firstly, alarms could be simultaneously issued by multiple downstream nodes, and the number of alarms is determined by the number of lightpaths traversing through the faulty link(s) and the length of the lightpaths. Such a large number of alarms could easily lead to an *alarm storm* in the control plane and bring risks of crashing the whole network. Secondly, the aforementioned approaches cannot handle multi-link SRLG failure events, due to the fact that a node can only be aware of the faulty link which is in the downlink of any lightpath traversing the node, but has no way to know the status of a link that none of the traversing lightpaths of the node goes through the link. Therefore, when a multi-link SRLG fails, a node may only be able to identify the failure of a subset of the links in the SRLG, and thus may select a protection path for restoration that is nonetheless subject to the failure, too. Thirdly, due to extensive electronic signaling mechanism and nodal processing, the aforementioned approaches may take hundreds milliseconds of delay just for the failure localization/notification process, and the delay is added up to the overall restoration time. Note that a slow restoration not only causes data loss but also imposes vicious impact on the upper network/transport layer protocols such as OSPF and TCP.

To improve the GMPLS based approaches, link-based monitoring [3][6][7][8][9][17] has been considered such that every link is exclusively monitored via a single-hop supervisory lightpath (S-LP). Once a failure occurs, the monitor(s) subject to LOL will issue an alarm, which is further shipped to the network controller or the corresponding decision nodes (e.g., edge routers) for subsequent restoration processes.

Although being an effective solution that can mitigate the awkward situation in the conventional GMPLS based approach, the link-based monitoring approach requires  $|E|$  WLs along with  $|E|$  transmitters/monitors that are considered precious resources in optical networks. Besides, similar to the GMPLS based approaches, link-based monitoring strongly relies on electronic signaling for failure notification, which leads to considerable control complexity and long restoration time. Thus, numerous schemes based on sophisticated designs and various assumptions were extensively reported in the past decades, which are surveyed in the next subsection.

### B. Literature Review

Harvey et al. [13] outlined non-adaptive probing schemes based on Combinatorial Group Testing (CGT). The methods find fault-free link sets in a network  $G$  and identify probes to localize faulty links using each link set as a hub. The schemes are for highly connected networks and cannot be applied in networks with limited connectivity to achieve the bounds on the minimum number of probes  $L^*(G, d)$ .

Ahuja et al. [5] proposed a multi-link SRLG failure localization method using monitoring cycle (MC) and monitoring path (MP). The complexity of the method is upper bounded by  $O(|\Psi|^2|V|\log_2|V|)$  where  $|\Psi|$  is the number of SRLGs under consideration. The required number of m-trails  $|\mathfrak{M}|$  is upper bounded by  $O(|\Psi|^2)$ ; the upper bound is reduced to  $O(|\Psi|)$  in this paper.

Wu et al. [18] provided an approach for optimally allocating m-trails to achieve unambiguous SRLG failure localization using ILP where decimal alarm codes of each pair of SRLGs are kept dissimilar. Thus the number of constraints is in  $O(|\Psi|^2)$  which is reduced to  $O(|E||\Psi|)$  in this paper. Similar ideas of [18] were further explored on bidirectional monitoring trail (bm-trail) in [11], where Greedy Code Swapping (GCS) was developed to form each m-trail. The worst-case complexity of the method is  $O(d^4|V||E|^2 \log^2 |E|(1 + h_\Delta))$  where  $d$  is the maximum number of link in a SRLG and  $h_\Delta$  is the difference between maximum and minimum Hamming weight of the generated CGT codes. The method is applied in a *dense-SRLG* scenario.

Babarczyk et al. [19] proposed an algorithm called Adjacent-Link Failure Localization (AFL) that deals with *sparse-SRLGs* that includes all single-link and some adjacent-link SRLGs. Although effective, the complexity of AFL is  $O(\Delta^2|V||E|^3 \log^2 |E|)$  where  $\Delta$  is maximal nodal degree of the graph. We improve the worst-case complexity to  $O((|E| + 1)^d(|V| \log_2 |V| + |E|))$  in this paper where  $d$  is 2 for all single-link and adjacent-link SRLGs.

### III. PROBLEM DESCRIPTION

#### A. M-Burst Framework

With the proposed m-burst framework, a short-duration lightpath (i.e., an optical burst) is launched along each m-trail to probe its status. Thus time-domain multiplexing of multiple m-trails along a single wavelength channel (WL) is possible, which allows for a significant saving of monitoring resources. However, optical bursts can only cut through the optical domain without buffering. To achieve the time-domain multiplexing of optical bursts, the MN has to schedule the timing of launching the burst for each m-trail, such that no collision will occur along any link. A naïve burst schedule is that each m-trail is sequentially probed (i.e., the MN does not launch the burst for the next m-trail until the current probing result is obtained), which nonetheless yields the longest delay by summing up the propagation time along all the m-trails. Thus, simultaneously probing multiple m-trails while avoiding any burst collision is essential.

Fig. 1 gives an example on the proposed m-burst framework operation and burst schedule. As shown in Fig.1(a), node A is the MN and  $T_1$  and  $T_2$  are two m-trails in dashed and dotted lines, respectively. Node A launches bursts  $\beta_1$  and  $\beta_2$  at time instants  $t_1$  and  $t_2$  along  $T_1$  and  $T_2$ , respectively. The length of each burst is  $L$  ms. By assuming a single monitoring WL along each link, the periods that  $\beta_1$  and  $\beta_2$  traverse through links  $(C, E)$  and  $(D, A)$  (which are taken by both m-trails) should be interleaved by at least  $L$ . This creates two constraints on the timing of launching the two bursts at MN:  $t_1 + t_{ab} + t_{bc} + L \leq t_2 + t_{ac}$  or  $t_1 + t_{ab} + t_{bc} \geq t_2 + t_{ac} + L$  for link  $(C, E)$  and  $t_1 + t_{ab} + t_{bc} + t_{ce} + t_{ed} + L \leq t_2 + t_{ac} + t_{ce} + t_{ef} + t_{fd}$  or  $t_1 + t_{ab} + t_{bc} + t_{ce} + t_{ed} \geq t_2 + t_{ac} + t_{ce} + t_{ef} + t_{fd} + L$  for link  $(D, A)$ , where  $t_{ab}$ ,  $t_{bc}$ ,  $t_{ce}$ ,  $t_{ed}$ ,  $t_{ac}$ ,  $t_{ef}$  and  $t_{fd}$  are the propagation times of links  $(A, B)$ ,  $(B, C)$ ,  $(C, E)$ ,  $(E, D)$ ,  $(A, C)$ ,  $(E, F)$  and  $(F, D)$ , respectively.

In a general scenario let there be  $b$  m-trails denoted as  $T_1, T_2, \dots, T_b$ . The burst schedule can be expressed as vector

of  $b$  entries  $\mathcal{B} = [t_1, t_2, \dots, t_b]$ , which defines the relative timing of launching burst along each m-trail. We assume the MN continuously probes the m-trails according to the burst schedule, and after all the m-trails are probed the MN immediately starts the next round of burst schedule. A *monitoring period* is defined as the time period required to complete a round of m-trail probing where the MN starts to probe  $T_i \forall i$  until the probing results of all the other  $b - 1$  m-trails are obtained, which directly contributes to failure localization latency. In other words, once the probing result on an m-trail is positive indicating a link failure event, the decision on failure localization will be made after the probing results of the next  $b - 1$  m-trails are obtained, and this period accounts to failure localization latency.

Note that the optical signal propagation is precise; thus with a burst schedule, the timing of arrival and departure of a burst upon any link is defined accordingly. It yields an important fact that the MN can determine an m-trail as failed if the MN did not receive a burst from a specific input port at a specific time instant. On the other hand, an intermediate node of an m-trail knows when and how to configure its cross-connect without relying on any additional signaling. This is possible since the m-trail deployment only depends on the network topology, which is considered rather static; and the signaling of burst schedule for each node can be done once at the beginning using the resource reservation protocol with traffic engineering (RSVP-TE), where each node can periodically configure its cross-connect at the reference time instant according to the assigned schedule.

Fig. 1(b) shows the timing diagram for probing the two m-trails at the MN by assuming  $\delta$  as a uniform propagation delay of each link. The constraints stated above become  $t_1 + 2\delta + L \leq t_2 + \delta$  or  $t_1 + 2\delta \geq t_2 + \delta + L$  for link  $(C, E)$  and  $t_1 + 4\delta + L \leq t_2 + 4\delta$  or  $t_1 + 4\delta \geq t_2 + 4\delta + L$  for link  $(D, A)$ . After the simplification, the constraints are  $t_1 + L + \delta \leq t_2$  or  $t_1 \geq t_2 + L - \delta$  for link  $(C, E)$  and  $t_1 + L \leq t_2$  or  $t_1 \geq t_2 + L$  for link  $(D, A)$ . Since both  $T_1$  and  $T_2$  traverse through links  $(C, E)$  and  $(D, A)$ , the maximum failure localization latency can be minimized by utilizing the monitoring WL of these links as much as possible at the same time satisfying the constraints. If we assign  $t_1 = L$  and  $t_2 = 0$  for the first monitoring period, maximum failure localization latency will be  $t_1 + 5\delta + L = 2L + 5\delta$  which is the least possible value. Thus, each m-trail is probed once every  $2L + 5\delta$  ms and  $t_1$  always lags  $t_2$  by  $L$  ms.

In the example, node C has to enable a cross-connect from A to E at every time instant  $\delta + k(2L + 5\delta)$ , and from B to E at every time instant  $L + 2\delta + k(2L + 5\delta)$  for  $k = 0, 1, 2, 3, \dots$ . Similarly, E connects C to F at every time instant  $2\delta + k(2L + 5\delta)$  and C to D at every time instant  $L + 3\delta + k(2L + 5\delta)$ ; while D has to set up a connection from F to A at every time instant  $4\delta + k(2L + 5\delta)$  and from E to A at every time instant  $L + 4\delta + k(2L + 5\delta)$ .

To summarize, the proposed m-burst framework aims to solve a number of legacy problems inherent in the context of failure localization in all-optical mesh networks. Firstly, the m-burst framework overcomes the deficits of link-based monitoring as given in II.A, which serves as an alternative to the state-

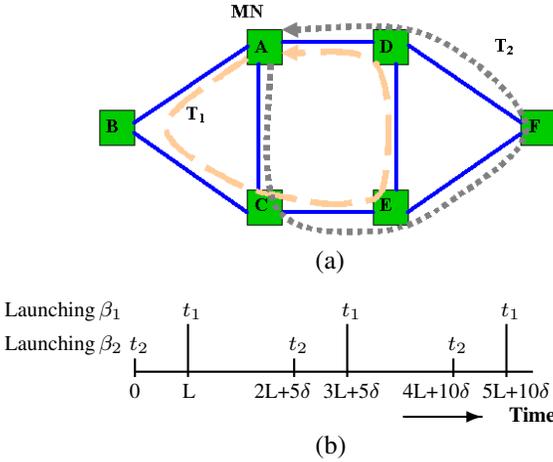


Fig. 1. Timing diagram for 2 m-bursts.

of-the-art industry solution. Secondly, the conventional m-trail based approaches may consume a huge amount of monitoring resources which make them not practical. For example, [10] shows that on average two WLs along each link are needed in order to localize six SRLGs in a 10-node topology; while [11] shows that more than 10 WLs along each link are needed to localize any SRLG with up to 3 links in randomly generated large topologies. The m-burst framework makes those conventional m-trail based approaches more feasible due to significantly reduced resource consumption by time-multiplexing the traversing m-trails along each link. Thirdly, since the optical switch of each network node is configured in time and space in advance, there will not be any real-time control complexity and signaling overhead; and the resultant failure monitoring system is truly all-optical, signaling-free, and deterministic.

### B. M-trail Allocation Problem Formulation

M-trail allocation with a single MN for multi-link failure localization has been investigated in [18] and [5]. However, the two studies have very different design premises from that of this paper. In [18], an ILP is formulated where m-trails are freely routed, while [5] used a very simple method for m-trail allocation which yields a complexity of  $O(|SRLG|^2|V|\log_2|V|)$  where  $|V|$  is the number of nodes. This subsection provides detailed formulation of the m-trail allocation problem of interest in the study.

The m-trail allocation problem is to find a sequence of directed links  $(v_0, v_1), (v_1, v_2), \dots, (v_{n-2}, v_{n-1}), (v_{n-1}, v_n)$  in the network  $G = (V, E)$ , where  $v_0 = v_n$ ,  $(v_i, v_{i+1}) \neq (v_j, v_{j+1})$  if  $i \neq j$ , and  $(v_i, v_{i+1}) \in E$  for  $i = 0, \dots, n-1$ . If an m-trail traverses  $(u, v)$ , it has to traverse  $(v, u)$  too. But if an m-cycle traverses  $(u, v)$ , it can not traverse  $(v, u)$ . On the other hand, an m-trail or an m-cycle can visit a node multiple times as they are non-simple structures. An m-cycle is considered as a special case of an m-trail.

Generally, a solution consists of a set of  $J$  m-trails  $m^{(J-1)}, \dots, m^1, m^0$ . Each m-trail has a specific bit position in an alarm code denote as  $[a^{(J-1)}, \dots, a^1, a^0]$ . In the alarm code of a SRLG,  $a^j = 1$  means that m-trail  $m^j$  traverses the SRLG and

$a^j = 0$  otherwise. By properly designing a set of m-trails with respect to the SRLGs under consideration, the on-off status of the m-trails during a failure event can define a valid alarm code that maps the failure event to a particular SRLG. The alarm code table (ACT) maintains the mapping between alarm codes and SRLG failure events. Therefore, if each single or multiple link SRLG is traversed by a unique set of m-trails, any single or multiple link SRLG failure can be localize unambiguously by identifying the set of disrupted m-trails.

Let  $\forall (u, v), (v, u) \in E$  the set of m-trails traversing through undirected link  $(u, v)$  be denoted as  $\varphi_{(u,v)}$ . If  $m^0, m^1$  and  $m^2$  traverse through undirected link  $(w, x)$ , and  $m^2, m^3$  and  $m^4$  traverse through undirected link  $(y, z)$ , we have  $\varphi_{(w,x)} = \{m^0, m^1, m^2\}$  and  $\varphi_{(y,z)} = \{m^2, m^3, m^4\}$ , respectively. Again, let the set of m-trails traversing SRLG  $\psi_i$  be denoted as  $\varphi_{\psi_i}$ . The set of m-trail traversing a single link SRLG is obviously the set of m-trails traversing through the link of the SRLG. Now, if a SRLG  $\psi_j = \{(w, x)\}$ , we have  $\varphi_{\psi_j} = \varphi_{(w,x)} = \{m^0, m^1, m^2\}$ . On the other hand, the set of m-trail traversing a multi-link SRLG is the union of the sets of m-trails traversing through the links of the SRLG. Here,  $\forall \psi_i \in \Psi$ :  $\psi_i$  is a multi-link SRLG,  $\varphi_{\psi_i} = \bigcup_{(u,v) \in \psi_i} \varphi_{(u,v)}$ . This is equivalent to bitwise logical OR operation on the link alarm codes of SRLG  $\psi_i$ . Now, if a multi-link SRLG  $\psi_k = \{(w, x), (y, z)\}$ , we have  $\varphi_{\psi_k} = \varphi_{(w,x)} \cup \varphi_{(y,z)} = \{m^0, m^1, m^2, m^3, m^4\}$ . To localize single or multiple link SRLG failures unambiguously, the set of m-trails traversing each SRLG should be unique.

## IV. THEORETICAL ANALYSIS

Let the sets of the SRLGs, the m-trails and the m-trails traversing through undirected link  $(u, v)$  be denoted as  $\Psi, \mathfrak{M}$  and  $\varphi_{(u,v)}$ , respectively.

**Lemma 1.**  $\forall (u, v), (w, x) \in E$ , if  $\varphi_{(u,v)} \subset \varphi_{(w,x)}$  or  $\varphi_{(w,x)} \subset \varphi_{(u,v)}$ , multiple link SRLG failures involve  $(u, v)$  and  $(w, x)$  can not be unambiguously localized from the MN by detecting only disrupted m-trails.

*Proof:* We use prove by contradiction. Assuming that the lemma is false, i.e., in case of  $\varphi_{(u,v)} \subset \varphi_{(w,x)}$  or  $\varphi_{(w,x)} \subset \varphi_{(u,v)}$ , multiple link SRLG failures involve  $(u, v)$  and  $(w, x)$  can be unambiguously localized from the MN by detecting only disrupted m-trails.

Let two SRLGs be  $\psi_i = \{(u, v)\}$  and  $\psi_j = \{(u, v), (w, x)\}$ , and the sets of m-trails traversing links  $(u, v)$  and  $(w, x)$  be  $\varphi_{(u,v)} = \{m^0, m^1, m^2\}$  and  $\varphi_{(w,x)} = \{m^0, m^2\}$ , respectively. Here,  $\varphi_{(w,x)} \subset \varphi_{(u,v)}$ . If the SRLG  $\psi_j$  becomes faulty, m-trails  $m^0, m^1$  and  $m^2$  will be disrupted. Detecting disrupted m-trails, the MN can not determine whether  $\psi_i$  or  $\psi_j$  become faulty because both the SRLGs are traversed by m-trails  $m^0, m^1$  and  $m^2$ . It is a contradiction. ■

**Corollary 1.**  $\forall (u, v), (w, x) \in E$ , if  $\varphi_{(u,v)} \not\subseteq \varphi_{(w,x)}$  and  $\varphi_{(w,x)} \not\subseteq \varphi_{(u,v)}$ , single link faults can be unambiguously localized from the MN by detecting only disrupted m-trails.

**Corollary 2.**  $\forall (u, v) \in E$ :  $(u, v)$  is not adjacent to the MN, at least two m-trails must traverse  $(u, v)$  to enable unambiguous

localization of multiple link SRLG faults from the MN by detecting only disrupted m-trails.

**Lemma 2.**  $\forall \psi_i, \psi_j \in \Psi$ :  $\psi_i \neq \psi_j$  and  $|\psi_i| = |\psi_j|$ , if the set of m-trails traversing  $\psi_i$  is not the same set of m-trails traversing  $\psi_j$ , i.e.,  $\varphi_{\psi_i} \neq \varphi_{\psi_j}$ , multi-link SRLG faults involve  $\psi_i$  and  $\psi_j$  can be unambiguously localized from the MN by detecting only disrupted m-trails.

*Proof:* From inspection, faults involve  $\psi_i$  and  $\psi_j$  will disrupt mutually different sets of m-trails because  $\psi_i$  and  $\psi_j$  are traversed by two different sets of m-trails. ■

**Lemma 3.**  $\forall \psi_i, \psi_j \in \Psi$ :  $\psi_i \subset \psi_j$ , if the set of m-trails traversing  $\psi_i$  is a proper subset of those traversing  $\psi_j$ , i.e.,  $\varphi_{\psi_i} \subset \varphi_{\psi_j}$ , multi-link SRLG faults involve  $\psi_i$  and  $\psi_j$  can be unambiguously localized from the MN by detecting only disrupted m-trails.

*Proof:* From inspection, at least one more m-trail traverses the superset SRLG. Thus link faults involving the two SRLGs will disrupt different numbers of m-trails. ■

To identify multiple link faults unambiguously, it is evident that a) m-trails traversing a link cannot be a subset of those traversing another link of the network, b) each SRLG with the same number of links should be traversed by a unique set of m-trails and c) a set of links must be traversed by strictly less numbers of m-trails than those traversing its superset.

The proposed method ensures that during a failure event each healthy link of the network is traversed by at least one uninterrupted m-trail. Faulty links are identified from the MN by detecting disrupted m-trails. Now we prove that the set of m-trails identified as solution by the proposed method has the three above mentioned characteristics. As a consequence, the method has the capability to localize multiple link SRLG faults unambiguously.

**Theorem 1.**  $\forall \psi_i \in \Psi$ , if each link in  $E \setminus \psi_i$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_i$ , then

- i) m-trails traversing each link is not a subset of those traversing another link of the network;
- ii) each SRLG having the same number of links are traversed by a unique set of m-trails; and
- iii) m-trails traversing each SRLG is a proper subset of those traversing any of its superset SRLGs.

*Proof:* i)  $\forall (u, v), (w, x) \in E$ , let single link SRLGs be  $\psi_j = \{(u, v)\}$  and  $\psi_k = \{(w, x)\}$ . As each link in  $E \setminus \psi_j$  including link  $(w, x)$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_j$ , at least one m-trail that does not traverse  $\psi_j$  will traverse  $\psi_k$  because  $(w, x) \in \psi_k$ . Hence,  $\varphi_{(w, x)} \not\subseteq \varphi_{(u, v)}$ .

Similarly, as each link in  $E \setminus \psi_k$  including link  $(u, v)$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_k$ , at least one m-trail that does not traverse  $\psi_k$  will traverse  $\psi_j$  because  $(u, v) \in \psi_j$ . Hence,  $\varphi_{(u, v)} \not\subseteq \varphi_{(w, x)}$ .

ii)  $\forall \psi_j, \psi_k \in \Psi$  where  $\psi_j \neq \psi_k$  and  $|\psi_j| = |\psi_k|$  indicates that there exists at least one link that is in  $\psi_j$  but not in  $\psi_k$  and vice versa. Assume  $(u, v)$  and  $(w, x)$  are such links where  $(u, v) \in \psi_j \wedge (u, v) \notin \psi_k$  and  $(w, x) \in \psi_k \wedge (w, x) \notin \psi_j$ .

As each link in  $E \setminus \psi_j$  including link  $(w, x)$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_j$ , at least one m-trail that does not traverse  $\psi_j$  will traverse  $\psi_k$  because  $(w, x) \in \psi_k$ . Hence,  $\varphi_{\psi_k} \not\subseteq \varphi_{\psi_j}$ .

Similarly, as each link in  $E \setminus \psi_k$  including link  $(u, v)$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_k$ , at least one m-trail that does not traverse  $\psi_k$  will traverse  $\psi_j$  because  $(u, v) \in \psi_j$ . Thus,  $\varphi_{\psi_j} \not\subseteq \varphi_{\psi_k}$ . Hence,  $\psi_j$  and  $\psi_k$  are traversed by unique sets of m-trails.

iii)  $\forall \psi_j, \psi_k \in \Psi$ :  $\psi_j \subset \psi_k$ . As  $\psi_j \subset \psi_k$ , all the m-trails that traverse  $\psi_j$  will also traverse  $\psi_k$ . Hence,  $\varphi_{\psi_j} \subseteq \varphi_{\psi_k}$ .

Again,  $\psi_j \subset \psi_k$  indicates that there exists at least one link that is in  $\psi_k$  but not in  $\psi_j$ . Assume that  $(w, x)$  is such a link where  $(w, x) \notin \psi_j \wedge (w, x) \in \psi_k$ . As each link in  $E \setminus \psi_j$  including link  $(w, x)$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_j$ , at least one m-trail that does not traverse  $\psi_j$  will traverse  $\psi_k$  because  $(w, x) \in \psi_k$ . Thus,  $|\varphi_{\psi_j}| < |\varphi_{\psi_k}|$ . ■

**Theorem 2.**  $\forall \psi_i \in \Psi$ , if each link in  $E \setminus \psi_i$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_i$ , single and multiple link SRLG faults can be unambiguously localized from the MN by detecting only disrupted m-trails.

*Proof:*  $\forall \psi_j, \psi_k \in \Psi$ :  $|\psi_j| \leq |\psi_k|$ . Thus,  $\psi_j$  may or may not be a subset of  $\psi_k$ . But  $\psi_k$  can not be a subset of  $\psi_j$ . Thus, there arise two cases.

Case 1,  $\psi_j \subset \psi_k$ : In this case  $|\psi_j|$  must be less than  $|\psi_k|$ . It implies that all the m-trails that traverse  $\psi_j$  will also traverse  $\psi_k$ . i.e.,  $\varphi_{\psi_j} \subseteq \varphi_{\psi_k}$ . Moreover,  $\exists (w, x) \in E$ :  $(w, x) \notin \psi_j \wedge (w, x) \in \psi_k$ . As each link in  $E \setminus \psi_j$  including link  $(w, x)$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_j$ , we have  $|\varphi_{\psi_j}| < |\varphi_{\psi_k}|$ .

Case 2,  $\psi_j \not\subset \psi_k$ : In this case  $|\psi_j|$  is less than or equal to  $|\psi_k|$ . In either situation  $\exists (u, v), (w, x) \in E$ :  $(u, v) \in \psi_j \wedge (u, v) \notin \psi_k$  and  $(w, x) \notin \psi_j \wedge (w, x) \in \psi_k$ . As each link in  $E \setminus \psi_k$  including link  $(u, v)$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_k$ , we have  $\varphi_{\psi_j} \not\subseteq \varphi_{\psi_k}$ . Again, as each link in  $E \setminus \psi_j$  including link  $(w, x)$  is traversed by at least one m-trail that is disjointed from the SRLG  $\psi_j$ , we have  $\varphi_{\psi_k} \not\subseteq \varphi_{\psi_j}$ .

Thus, each SRLG is traversed by a unique set of m-trails. Hence, disrupted m-trails will identify the faulty SRLG. ■

We observe that theorem 2 is a sufficient but not a necessary condition for unambiguous localization of single and multiple link SRLG faults from the MN by detecting only disrupted m-trails. To show that the theorem is not a necessary condition, let two SRLGs be  $\psi_i$  and  $\psi_j$  where  $\psi_j \not\subset \psi_i$  and they are traversed by two sets of m-trails namely  $\varphi_{\psi_i} = \{m^1, m^2, m^3\}$  and  $\varphi_{\psi_j} = \{m^1, m^2\}$ , respectively. Thus,  $\psi_i$  and  $\psi_j$  are traversed by unique sets of m-trails. Consequently, any fault involves  $\psi_i$  or  $\psi_j$  can be unambiguously localize even though  $\psi_j$  is not traverse by at least one m-trail which is disjointed from  $\psi_i$ .

**Example:** Fig. 2 shows a three link network to explain how the proposed multi-link SRLG failure localization method works. Node  $a$  is the MN. Five SRLGs are given as  $\psi_1 = \{(a, b)\}$ ,  $\psi_2 = \{(b, c)\}$ ,  $\psi_3 = \{(c, a)\}$ ,  $\psi_4 = \{(a, b), (b, c)\}$ ,  $\psi_5 = \{(b, c), (c, a)\}$ . Four m-trails are needed to identify the

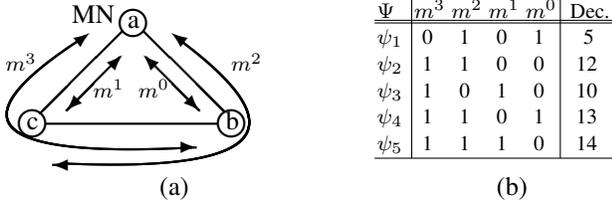


Fig. 2. a) An m-trail solution for a network with 3 node and 3 link; b) Alarm code table (ACT)

five SRLG failures as shown in Fig. 2(a). For each of the single link SRLG  $\psi_1$ ,  $\psi_2$  and  $\psi_3$  failure, the remaining two links are traversed by  $m^1$  and  $m^3$ ,  $m^0$  and  $m^1$ , and  $m^0$  and  $m^2$ , respectively. For each of the double link SRLG  $\psi_4$  and  $\psi_5$  failure, the remaining link is traversed by  $m^1$  and  $m^0$ , respectively. Hence,  $\forall \psi_i \in \Psi$ , each link in  $E \setminus \psi_i$  is traversed by at least one m-trail that is disjoint from the SRLG  $\psi_i$ .

Fig. 2(b) shows the ACT derived from the m-trails. Dec. represents decimal alarm codes. The unique decimal alarm codes for the SRLGs  $\psi_1$ ,  $\psi_2$ ,  $\psi_3$ ,  $\psi_4$  and  $\psi_5$  are 5, 12, 10, 13 and 14, respectively. Since each SRLG has a unique alarm code, it indicates that each SRLG is traversed by a unique set of m-trails. Therefore, the MN can localize SRLG faults unambiguously by detecting only disrupted m-trails.

## V. ILP FORMULATION

The proposed ILP is a manipulation of Theorem 2 in such a way that  $\forall \psi \in \Psi$  each link in  $E \setminus \psi$  is traversed by at least one m-trail that is link-disjoint from SRLG  $\psi$  while visiting the MN. Different from any previous research, the ILP determines the burst schedule and minimize the monitoring period by manipulating the routes of m-trails.

To reduce the problem size, we use enumerated m-trails as input to the ILP. For each node of the network,  $k$  unique shortest cycles/trails (m-trails) are derived by using Suurballe's algorithm for shortest pairs of disjoint paths [21] and Yen's algorithm for  $k$ -shortest loop-less paths [22].

The network is denoted as  $G = (V, E)$ , where  $V$  is the set of all nodes of the network and  $E$  is the set of all unidirectional links.  $r_1, r_2$  are two predefined cost ratios, and  $bl$  is the burst length in ms.  $J$  is the number of enumerated m-trails, and  $N$  represents infinity.

Pre-calculated parameters are  $ee_{uv}^j$ ,  $es_{\psi}^j$ ,  $des_{\psi}^{jab}$ ,  $p_k^{jr}$  and  $tp^j$ .  $ee_{uv}^j$  is 1 if the  $j$ -th enumerated m-trail traverses unidirectional link  $(u, v)$  and 0 otherwise.  $es_{\psi}^j$  is 1 if the  $j$ -th enumerated m-trail traverses any link of  $\psi$  and 0 otherwise.  $des_{\psi}^{jab}$  is 1 if  $es_{\psi}^j = 0 \wedge (ee_{ab}^j = 1 \vee ee_{ba}^j = 1)$  where undirected link  $(a, b) \in E \setminus \psi$  and 0 otherwise.  $p_k^{jr}$  is propagation delay in ms of the m-burst from the MN to  $k$ -th node of the  $j$ -th enumerated m-trail following given or reverse spin  $sp^{jr}$ .  $tp^j$  is total propagation delay in ms of the m-burst along the  $j$ -th enumerated m-trail.

Binary variables are  $m^j$ ,  $sp^{jr}$ ,  $e_{uv}^j$  and  $g_{uv}^{irjt}$ .  $m^j$  is 1 if the  $j$ -th enumerated m-trail is in an ILP solution and 0 otherwise.  $sp^{jr}$  is called the spin of  $j$ -th enumerated m-trail. Spin  $sp^{j0}$  is 1 if  $j$ -th m-trail in an ILP solution traverses its on-trail links in the enumerated direction and 0 otherwise. Similarly, spin

$sp^{j1}$  is 1 if  $j$ -th m-trail in an ILP solution traverses its on-trail links against the enumerated direction and 0 otherwise.  $e_{uv}^j$  is 1 if the  $j$ -th m-trail in an ILP solution traverses unidirectional link  $(u, v)$  with spin  $sp^{j0}$  or unidirectional link  $(v, u)$  with spin  $sp^{j1}$ , and 0 otherwise.  $g_{uv}^{irjt}$  is 1 if the m-burst along  $i$ -th m-trail in an ILP solution with spin  $sp^{ir}$  arrives to node  $u$  before the m-burst along  $j$ -th m-trail in the ILP solution with spin  $sp^{jt}$  and both the bursts traverse unidirectional link  $(u, v)$ , and 0 otherwise.

Integer variables are  $n$  and  $\alpha_{\psi}$ .  $n$  is the maximum number of m-trails in an ILP solution traversing through any unidirectional link in  $E$ .  $\alpha_{\psi}$  is the decimal alarm code of  $\psi$ .

Real variables are  $s^{jr}$  and  $T$ .  $s^{jr}$  is the m-burst launching time in ms from the MN along  $j$ -th enumerated m-trail.  $T$  is the maximum failure localization latency in ms.

### A. ILP for M-trail Allocation and M-burst Scheduling

The specific ILP formulation is provided as follows. Objective:

$$\text{Minimize } \{T + r_1 * n + r_2 * \sum_j m^j\} \quad (1)$$

Subject to the following constraints:

At least one enumerated m-trail in an ILP solution must traverse undirected link  $(a, b)$  that does not traverse any link of SRLG  $\psi$ . This is the key constraint that implements the proposed multiple link SRLG fault localization algorithm.

$$\sum_j m^j * des_{\psi}^{jab} \geq 1, \quad \forall \psi \in \Psi, \forall (a, b) \in E \setminus \psi \quad (2)$$

Each enumerated m-trail in an ILP solution has only one non-zero spin

$$sp^{j0} + sp^{j1} = m^j, \quad \forall j \quad (3)$$

The constraint ensures that only the  $j$ -th enumerated m-trail with valid spin in an ILP solution can be assigned positive burst starting time

$$s^{jr} \leq sp^{jr} * N, \quad \forall j, r \quad (4)$$

Constraints (5) and (6) enforce minimum difference between arrival times of two m-bursts to node  $u$  to the burst length  $bl$  when the m-bursts along  $i$ -th and  $j$ -th enumerated m-trails traverse same outgoing link  $(u, v)$  from the node. Thus, they ensure collision free burst propagation through any link of the network. Here, node  $u$  is the  $k$ -th node of  $i$ -th enumerated m-trail considering spin  $sp^{ir}$  and the  $l$ -th node of  $j$ -th enumerated m-trail considering spin  $sp^{jt}$ . Moreover, node  $v$  is the next hop node along the  $i$ -th and  $j$ -th enumerated m-trail considering their spins. Now  $\forall i, r, k, j, t, l, (u, v)$

$$(s^{ir} + p_k^{ir}) - (s^{jt} + p_l^{jt}) \geq (1 - (N + 1) * g_{uv}^{irjt}) * bl + (sp^{ir} + sp^{jt} - 2) * N \quad (5)$$

$$(s^{jt} + p_l^{jt}) - (s^{ir} + p_k^{ir}) \geq (g_{uv}^{irjt} - (1 - g_{uv}^{irjt}) * N) * bl + (sp^{ir} + sp^{jt} - 2) * N \quad (6)$$

The maximum failure localization latency is the largest elapse time for any m-burst with valid spin from the start of the monitoring period to the return time of the last bit of the m-burst to the MN

$$T \geq s^{jr} + m^j * tp^j + bl + (sp^{jr} - 1) * N, \quad \forall j, r \quad (7)$$

Link traversal by m-trails in an ILP solution considering their spins

$$e_{uv}^j \geq m^j * ee_{uv}^j + sp^{j0} - 1, \quad \forall (u, v) \in E \quad \forall j \quad (8)$$

$$e_{vu}^j \geq m^j * ee_{uv}^j + sp^{j1} - 1, \quad \forall (u, v) \in E \quad \forall j \quad (9)$$

Calculation of the maximum number of m-trails traversing through any unidirectional link of the network

$$n \geq \sum_j e_{uv}^j, \quad \forall (u, v) \in E \quad (10)$$

The following constraint is for the calculation of decimal alarm codes, but it is not a constraint of the ILP per se. However, using the alarm codes, we can verify if the method derives a unique alarm codes for each SRLG or not.

$$\alpha_\psi = \sum_j 2^j * es_\psi^j, \quad \forall \psi \in \Psi \quad (11)$$

Objective (1) aims to minimize mainly the maximum failure localization latency. In order to find the failure localization latency, the number of m-trails traversing through any unidirectional link of the network and the total number of m-trails  $|\mathfrak{M}|$  are also minimize.

From Eq. (2), we have the number of constraints in the order of  $O(|E||\Psi|)$  where  $|E|$  is the number of network links and  $|\Psi|$  is the number of SRLGs under consideration. As  $|\Psi| \leq (|E| + 1)^d$ ,  $O(|E||\Psi|)$  is equivalent to  $O(|E|(|E| + 1)^d)$  where  $d$  is the maximum number of links in any SRLG. From Eq. (11), we have the number of variables in the order of  $O(|\Psi|)$  based on the variable  $\alpha_\psi$ .

## VI. A HEURISTIC ALGORITHM

The ILP is not scalable to the problem size; thus a heuristic algorithm is provided in Fig. 3. All links are considered undirected. We initialize the m-trail solution set  $\mathfrak{M}$  and ACT  $\mathfrak{A}$  as empty in step 1. Step 2 finds an m-trail for each node of  $G$  based on the node's shortest path to the MN. In step 3, the shorter m-trail based on shortest distances  $d(u)$  and  $d(v)$  is selected from the two m-trails that terminate at the end nodes  $u$  and  $v$  of each link. Then the link is concatenated to the m-trail as its last link to derive a new m-trail. The new m-trail is added to  $\mathfrak{M}$  and  $\mathfrak{A}$ . In step 4, each SRLG  $\psi \in \Psi$  is considered in turn. At first, each m-trail that traverses any link of  $\psi \in \Psi$  is deleted from  $\mathfrak{A}$  to form a temporary ACT  $\mathfrak{A}_\psi$ . Then for each link not in the SRLG  $\psi$  and if the alarm code of the link becomes 0 in  $\mathfrak{A}_\psi$ , the link is added to an array  $Q_\psi$ . Next, for each link in  $Q_\psi$  if the alarm code of the link remains 0 in  $\mathfrak{A}_\psi$ , a new m-trail is found that traverses the link but is link-disjoint from the SRLG  $\psi$  using the helper function given in the Fig. 4. Finally, the new m-trail is added to  $\mathfrak{M}$ ,  $\mathfrak{A}$  and  $\mathfrak{A}_\psi$ . Step 5 returns the solution set of m-trails  $\mathfrak{M}$  and ACT  $\mathfrak{A}$ .

---

Step 1: Initialize  $\mathfrak{M}, \mathfrak{A} \leftarrow \phi$   
Step 2: Find shortest distances from the MN in  $G$   
For each node  $u \in V$ , build an m-trail  $m_u^i$  based on the shortest path  $p(u)$  from the MN  
Step 3: For each link  $(u, v) \in E$ , build an m-trail  $m^j$  with  $(u, v)$  and  $m_u^i$  if  $d(u) \leq d(v)$ . Otherwise, build  $m^j$  with  $(u, v)$  and  $m_v^i$ . Add  $m^j$  to  $\mathfrak{M}$  and  $\mathfrak{A}$   
Step 4: For each  $\psi \in \Psi$  do  
For each  $m^j \in \mathfrak{M}$ :  $m^j \in \varphi_\psi$ , delete  $m^j$  from  $\mathfrak{A}$  to form  $\mathfrak{A}_\psi$   
For each link  $(a, b) \in E \setminus \psi$ :  $\mathfrak{A}_\psi^{ab} == 0$ , add  $(u, v)$  to  $Q_\psi$   
For each  $(a, b) \in Q_\psi$ :  $\mathfrak{A}_\psi^{ab} == 0$ , call the function FindMTrail( $\mathfrak{A}_\psi, \psi, (a, b)$ ) to get an m-trail  $m^j$ . Add  $m^j$  to  $\mathfrak{M}$ ,  $\mathfrak{A}$ , and  $\mathfrak{A}_\psi$   
End For  
Step 5: Return  $\mathfrak{M}, \mathfrak{A}$

---

Fig. 3. Pseudocode of the multiple link SRLG fault localization method.

Function in Fig. 4 finds a new m-trail that traverses the link  $(a, b)$  but is link-disjoint from SRLG  $\psi$ . In step 1, ordered list  $L$  stores nodes  $a$  and  $b$  in ascending order of the lengths of their shortest paths  $p(a)$  and  $p(b)$  from the MN, respectively. The first node in  $L$  will be used to find the new m-trail before the second one in step 2. In step 2, neighbourhoods of both nodes are checked in turn. If any neighbor  $N(v)$  of node  $u$  is visited by an m-trail that is link-disjoint from SRLG  $\psi$ , the portion of the m-trail from the MN to node  $v$  is added with links  $(u, v)$  and  $(a, b)$  to build the new m-trail. If such an m-trail is not found in step 2, the link  $(a, b)$  and all the links of the SRLG  $\psi$  are deleted from  $G$  to derive a truncated network  $G_\psi$  in step 3. Then, two shortest paths  $p(a)$  and  $p(b)$  to the nodes  $a$  and  $b$ , respectively are found in  $G_\psi$ . The new m-trail is derived by using  $(a, b)$  and the shorter of the two shortest paths  $p(a)$  and  $p(b)$ .

Step 1 of the helper function in Fig. 4 requires  $O(1)$  steps. Step 2 iterates only twice: one for each end node of link  $(a, b)$ . For each node, it requires  $O(1)$  steps to check the neighbors of node  $a$  or  $b$  for uninterrupted m-trails. It takes at most  $O(\log_2 |\mathfrak{M}|)$  steps to search an uninterrupted m-trail. Thus step 2 needs  $O(d \log_2 (|E| + 1))$  steps as  $|\mathfrak{M}| \in O(|\Psi|)$  shown below and  $|\Psi| \leq (|E| + 1)^d$ , where  $d$  is the maximum number of simultaneous faulty links. Step 3 requires at most  $O(|V| \log_2 |V| + |E|)$  steps via Dijkstra's algorithm. Thus, complexity of the helper function is  $O(|V| \log_2 |V| + |E|)$ .

The overall worst case complexity of the method is derived as follows. Step 2 of Fig. 3 requires  $O(|V| \log_2 |V| + |E|)$  steps. Step 3 requires  $O(|E|)$  steps. In step 4, outer for loop requires  $O(|\Psi|)$  steps. Deleting m-trails from  $\mathfrak{A}$  needs  $O(|E|)$  steps. To populate  $Q_\psi$  requires  $|E|$  steps. As each SRLG can disrupt a small number of m-trails, the number of links that are not being traversed by at least one uninterrupted m-trail will also be small. Thus,  $|Q_\psi|$  can be considered as constant. To find a new m-trail for a link in  $Q_\psi$  requires complexity of  $O(|V| \log_2 |V| + |E|)$ . Thus step 4 takes the worst case

---

```

Function FindMTrail( $\mathfrak{A}_\psi, \psi, (a, b)$ )
Step 1: Build an ordered list  $L = [a, b]$ , If  $d(a) > d(b)$ , swap
positions of  $a$  and  $b$  in  $L$ 
Step 2: For each  $u \in L$  do
  For each  $v \in N(u)$  and for each  $w \in N(v)$  do
    If  $v \neq a \wedge v \neq b \wedge w \neq a \wedge w \neq b \wedge \mathfrak{A}_\psi^{vw} \geq 1$ ,
    find any m-trail  $m^i$  that visits  $v$ , get the portion  $m_v^i$ 
of the m-trail from the MN to  $v$ , add  $(a, b)$  and
 $(u, v)$  to  $m_v^i$  to form an m-trail  $m^j$ , and return  $m^j$ 
  End For
End For
Step 3:  $G_\psi \leftarrow$  delete  $(a, b)$  and each  $(u, v) \in \psi$  from  $G$ 
Find shortest distances from the MN in  $G_\psi$ 
Build an m-trail  $m^j$  based on the shortest path  $p(a)$ 
from the MN and  $(a, b)$  if  $d(a) \leq d(b)$ . Otherwise,
build  $m^j$  based on the shortest path  $p(b)$  from the MN
and  $(a, b)$ 
Return  $m^j$ 
End Function

```

---

Fig. 4. Pseudocode of the function FindMTrail.

complexity of  $O(|\Psi|(|V| \log_2 |V| + |E|))$ .

Next, we derive asymptotic upper bound of the number of m-trails denoted as  $|\mathfrak{M}|$ . In step 3 of Fig. 3,  $|E|$  m-trails are added to the solution. In step 4, upper bound of m-trails to cover links in  $Q_\psi$  while disjoint from any link of  $\psi$  is  $O(1)$  because a small number of m-trails can be devised to cover all such links. Thus, we have  $|\mathfrak{M}| \in O(|\Psi|)$ , compared with  $|\mathfrak{M}| \in O(|\Psi|^2)$  in [5].

Once a set of m-trails  $\mathfrak{M}$  that provides a unique alarm code for each SRLG is derived by the SRLG fault localization method given in Fig. 3, the starting times  $s$  of the m-bursts along the m-trails from the MN are derived by using a burst scheduling heuristic described below. The primary objective of the burst scheduling heuristic is to find the minimum failure localization latency  $T$  avoiding burst collisions altogether. Initially,  $T$  is  $\infty$ .

The set of m-trails in the solution is stored in an ordered list  $seq$ . The near optimal m-trail sequence that provides the minimum failure localization latency is found by swapping positions of m-trail pairs in  $seq$  following Tabu search rules. The starting times  $s_{new}$  and the failure localization latency  $T_{new}$  of each sequence of m-trails in  $seq$  are calculated as explained below.

For each sequence of m-trails in  $seq$ , the starting time of each m-burst from the MN is calculated following the order of the sequence. The starting time of the m-burst along the first m-trail in  $seq$  is 0 and the starting times of the remaining m-bursts are initialized as 0. Each m-burst is kept non-overlapping with other m-bursts by increasing its starting time. The burst scheduling heuristic tries to find earliest possible burst launching time  $s^j$  of the m-burst along  $j$ -th m-trail  $m^j$  in  $seq$  avoiding collision with the m-bursts along m-trails in the preceding positions of  $seq$ . Specifically,  $\forall (u, v) \in E$  and  $\forall i \in \{0, \dots, (j-1)\}$ : both  $i$ -th and  $j$ -th m-trails in  $seq$  traverse

link  $(u, v)$ , if  $(s^j + p_{uv}^j + bl) > (s^i + p_{uv}^i)$  but  $(s^j + p_{uv}^j) < (s^i + p_{uv}^i + bl)$ ,  $s^j$  is increased to  $(s^i + p_{uv}^i + bl - p_{uv}^j)$ . Here  $p_{uv}^i$  and  $p_{uv}^j$  are the burst propagation delays from the MN to node  $u$  along m-trails  $m^i$  and  $m^j$ , respectively, and  $bl$  is the burst length. The set of burst starting times of the sequence is  $s_{new}$ .

Again, for each sequence of m-trails in  $seq$ , the failure localization latency  $T_{new}$  is the maximum time taken by any m-burst to return completely to the MN from the start of a monitoring period.  $\forall m^j \in \mathfrak{M}$ ,  $T_{new}$  is  $\max\{s^j + tp^j + bl\}$  where  $tp^j$  is the total burst propagation delay along m-trail  $m^j$ . If  $T_{new}$  is less than  $T$ , the starting times of m-bursts  $s$  and the minimum failure localization latency  $T$  are updated using  $s_{new}$  and  $T_{new}$  of the sequence, respectively. Thus, the burst scheduling heuristic itself is a min-max optimization method.

## VII. NUMERICAL RESULTS

We develop a model of the ILP described in Section V in the modeling language GNU MathProg and translate the MathProg model using GLPK solver glpsol into a MPS file to run the program in ILOG CPLEX 11.1.

We assume that the burst length  $bl$  is 20 ms, burst propagation delay through any link  $(u, v)$  is 2 ms, and there is at most one supervisory WL along each unidirectional link in the numerical experiment. We also assume that  $r_1$  is 0.1 and  $r_2$  is 0.01. As minimization of  $n$ , the maximum number of m-trails traversing any unidirectional link, helps to minimize the failure localization latency indirectly,  $r_1$  is assigned a low value. Again, as WL consumption is almost fixed for a network and bursts are launched from and returned to the MN, minimization of the number of m-trails turns into a secondary issue. Thus,  $r_2$  is assigned even a lower value than that of  $r_1$ .

We conduct the experiment on a network with 7 nodes and 12 links. Node 0 is assigned as the MN. A SRLG could consist of up to  $d = 3$  links, where each SRLG with two or three links is node-disjoint from the MN. In our case, there are in total 96 SRLGs consisting of 12 single, 28 double and 56 triple link SRLGs as shown in Table I.

The ILP solution has  $|\mathfrak{M}| = \sum_j m^j = 19$  m-trails as shown in Fig. 5. The set of m-trails populates the ACT which is the mapping between each possible alarm code and the corresponding SRLG. Each SRLG has an entry in the ACT of the network. The alarm code of an SRLG is determined by the m-trails traversing through the SRLG. For example, link  $(1, 2)$  is traversed by m-trails  $m^2, m^8, m^{16}$  and  $m^{18}$ , thus its alarm code is 101 0000 0001 0000 0100. As link  $(3, 4)$  is traversed by m-trails  $m^4, m^9, m^{10}$  and  $m^{15}$ , its alarm code is 000 1000 0110 0001 0000. Similarly, link  $(5, 6)$  is traversed by m-trails  $m^{14}, m^{15}, m^{16}$  and  $m^{18}$ , hence its alarm code is 101 1100 0000 0000 0000. The alarm code of SRLG  $\psi = \{(1, 2), (3, 4), (4, 5)\}$  is 101 1100 0111 0001 0100, which is the bit-wise OR of its link alarm codes.

When an SRLG fails, a unique set of m-trails will be disrupted, which results in a unique alarm code. For example, if  $m^2, m^5, m^7, m^8, m^{10}, m^{16}, m^{17}$  and  $m^{18}$  are disrupted, the generated alarm code will be 111 0000 0101 1010 0100. The corresponding decimal alarm code is  $2^{18} + 2^{17} + 2^{16} + 2^{10} + 2^8 + 2^7 + 2^5 + 2^2 = 460196$ .

TABLE I  
LIST OF SRLGS

(0 1)	(1 3)(5 6)	(1 2)(1 6)(3 4)	(1 3)(2 6)(5 6)
(0 4)	(1 6)(2 3)	(1 2)(1 6)(4 5)	(1 3)(3 4)(4 5)
(0 5)	(1 6)(2 6)	(1 2)(1 6)(5 6)	(1 3)(3 4)(5 6)
(0 6)	(1 6)(3 4)	(1 2)(2 3)(2 6)	(1 3)(4 5)(5 6)
(1 2)	(1 6)(4 5)	(1 2)(2 3)(3 4)	(1 6)(2 3)(2 6)
(1 3)	(1 6)(5 6)	(1 2)(2 3)(4 5)	(1 6)(2 3)(3 4)
(1 6)	(2 3)(2 6)	(1 2)(2 3)(5 6)	(1 6)(2 3)(4 5)
(2 3)	(2 3)(3 4)	(1 2)(2 6)(3 4)	(1 6)(2 3)(5 6)
(2 6)	(2 3)(4 5)	(1 2)(2 6)(4 5)	(1 6)(2 6)(3 4)
(3 4)	(2 3)(5 6)	(1 2)(2 6)(5 6)	(1 6)(2 6)(4 5)
(4 5)	(2 6)(3 4)	(1 2)(3 4)(4 5)	(1 6)(2 6)(5 6)
(5 6)	(2 6)(4 5)	(1 2)(3 4)(5 6)	(1 6)(3 4)(4 5)
(1 2)(1 3)	(2 6)(5 6)	(1 2)(4 5)(5 6)	(1 6)(3 4)(5 6)
(1 2)(1 6)	(3 4)(4 5)	(1 3)(1 6)(2 3)	(1 6)(4 5)(5 6)
(1 2)(2 3)	(3 4)(5 6)	(1 3)(1 6)(2 6)	(2 3)(2 6)(3 4)
(1 2)(2 6)	(4 5)(5 6)	(1 3)(1 6)(3 4)	(2 3)(2 6)(4 5)
(1 2)(3 4)	(1 2)(1 3)(1 6)	(1 3)(1 6)(4 5)	(2 3)(2 6)(5 6)
(1 2)(4 5)	(1 2)(1 3)(2 3)	(1 3)(1 6)(5 6)	(2 3)(3 4)(4 5)
(1 2)(5 6)	(1 2)(1 3)(2 6)	(1 3)(2 3)(2 6)	(2 3)(3 4)(5 6)
(1 3)(1 6)	(1 2)(1 3)(3 4)	(1 3)(2 3)(3 4)	(2 3)(4 5)(5 6)
(1 3)(2 3)	(1 2)(1 3)(4 5)	(1 3)(2 3)(4 5)	(2 6)(3 4)(4 5)
(1 3)(2 6)	(1 2)(1 3)(5 6)	(1 3)(2 3)(5 6)	(2 6)(3 4)(5 6)
(1 3)(3 4)	(1 2)(1 6)(2 3)	(1 3)(2 6)(3 4)	(2 6)(4 5)(5 6)
(1 3)(4 5)	(1 2)(1 6)(2 6)	(1 3)(2 6)(4 5)	(3 4)(4 5)(5 6)

TABLE II  
FAILURE LOCALIZATION LATENCY

$m^j$	$s^j + p^j + bl = T^j$	$m^j$	$s^j + p^j + bl = T^j$
0	0 + 4 + 20 = 24	10	34 + 12 + 20 = 64
1	0 + 8 + 20 = 28	11	40 + 4 + 20 = 64
2	98 + 8 + 20 = 126	12	74 + 4 + 20 = 98
3	32 + 12 + 20 = 64	13	24 + 4 + 20 = 48
4	0 + 12 + 20 = 32	14	98 + 8 + 20 = 126
5	58 + 8 + 20 = 86	15	54 + 12 + 20 = 86
6	78 + 8 + 20 = 106	16	14 + 10 + 20 = 44
7	94 + 12 + 20 = 126	17	20 + 6 + 20 = 46
8	54 + 12 + 20 = 86	18	72 + 14 + 20 = 106
9	98 + 8 + 20 = 126		

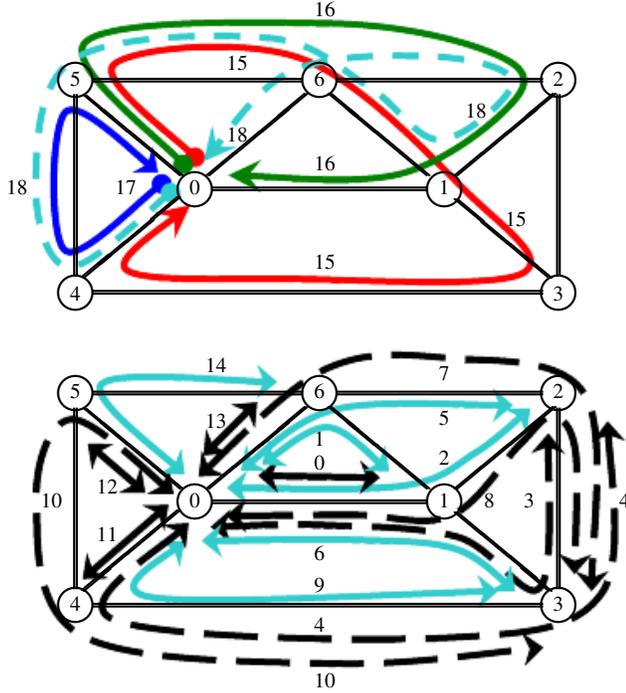


Fig. 5. M-trail solution set for a network with 7 nodes and 12 links. M-cycles are shown at the top part. Trail based m-trails are shown at the bottom part.

The starting time of the bursts  $s^j$ , total propagation delay through the m-trail  $p^j$ , burst length  $bl$ , and failure localization latency  $T^j$  for the  $j$ -th m-burst is shown in Table II. Maximum failure localization latency  $T$  is 126 ms. Thus,  $T$  is derived in the solution from m-trail  $m^2$ ,  $m^7$ ,  $m^9$  or  $m^{14}$ .

The solution allows at most  $n = 6$  m-trails through any unidirectional link of the network.

Due to space constraint, Fig. 6 shows collision-free m-burst traversal timing through only links from the MN to node 3 via nodes 1 and 2, and back to the MN using the same links during a monitoring period. Each rectangle represents an m-

burst and the m-burst is labeled with the corresponding m-trail. The traversal period of an m-burst through a link is equal to the burst propagation delay  $\delta$  through the link and the burst length  $bl$ . Thus, the period is  $\delta + bl = 2 + 20 = 22$ ms.

The periods when two m-bursts traverse through a link back-to-back are indicated by black rectangles in Fig. 6. The m-bursts along m-trails  $m^6$  and  $m^2$ ,  $m^{18}$  and  $m^2$ , and  $m^3$  and  $m^8$  traverse links (0, 1), (1, 2) and (2, 3), respectively, back-to-back. Similarly, the m-bursts along m-trails  $m^0$  and  $m^{16}$ ,  $m^{16}$  and  $m^3$ ,  $m^8$  and  $m^6$ , and  $m^6$  and  $m^2$  traverse link (1, 0) back-to-back. Each black rectangle in the figure spans 2ms, which is the same as the burst propagation delay through a link. It is obvious that whenever two m-bursts traverse a link back-to-back, they remain collision-free in the link. On the other hand, the m-burst along m-trail  $m^8$  starts traversing links (0, 1) or (1, 0) once the m-burst along m-trail  $m^3$  completed traversal of links (0, 1) or (1, 0), respectively. Thus, the m-bursts along m-trails  $m^3$  and  $m^8$  remain 2ms apart in links (0, 1, 0).

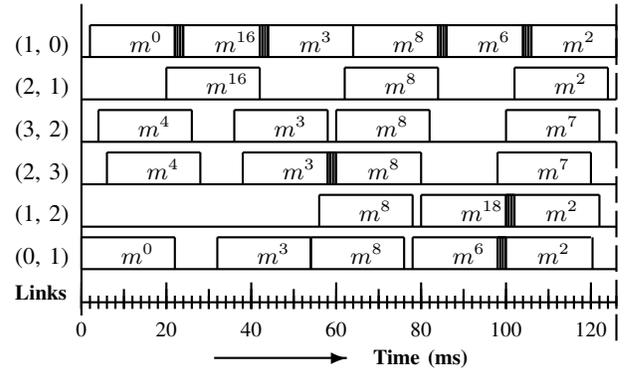


Fig. 6. Collision-free link traversal by multiple m-bursts.

To verify the proposed algorithm, we run the ILP on three more networks with an arbitrarily chosen MN in each network while keeping all other assumptions the same as the above experiment.

The simulation results in terms of the number of m-trails  $\sum_j m^j$ , the maximum number of m-trails through any unidirectional link  $n$ , and the maximum failure localization latency  $T$  of the solutions are given in Table III. The results for single link SRLG fault localization are taken from [14]. The results for single and multiple link SRLG fault localization are shown in columns  $S$  and  $M$ , respectively.

For each of the four networks, the method is able to identify

a set of m-trails to achieve unambiguous failure localization (UFL) for the multiple link SRLGs failures. Maximum failure localization latency  $T$  for each network is optimal. Maximum number of m-trails through any unidirectional link  $n$  for each network is low. For each network, the method is able to achieve collision-free burst scheduling by keeping the m-bursts along the m-trails non-overlapping through any link of the network.

By taking the single-link SRLG result as reference, when multi-link SRLGs are considered and increased by 1.67, 2.25, 4.89 and 8 times, the number of m-trails is increased by 3, 3, 3.75 and 3.17 times, the maximum number of m-trails traversing through any unidirectional link is increased by 3, 3, 2.5 and 3 times, and the maximum failure localization latency is increased by 2.62, 2.54, 2.42 and 2.63 times, respectively. It shows that the proposed approach is scalable to the number of SRLGs.

TABLE III  
RESULTS RELATED WITH ADDITIONAL NETWORKS

Performance metrics	Networks							
	4 nodes 6 links		5 nodes 8 links		6 nodes 9 links		7 nodes 12 links	
	S	M	S	M	S	M	S	M
$\sum_j m^j$	3	9	4	12	4	15	6	19
$n$	1	3	1	3	2	5	2	6
$T$ ms	26	68	26	66	48	116	48	126

The proposed heuristic method is implemented on an Intel Centrino Due, 1.00GB RAM, 2.00GHz, Windows XP Home Edition Version 2002 Service Pack 2 Toshiba Satellite T2500 Laptop by keeping all the assumptions the same as the ILP experiments. In addition to all the networks used in the single and multiple link SRLG failure ILP experiments, we test the heuristic method on 4 larger topologies modified from CERNet, SmallNet, NSFNet and Bellcore such that they become  $d + 1$  connected. Note that  $d + 1$  connected is a necessary condition for the topologies where a single MN can localize any SRLG up to  $d$  links as disclosed in [5]. The result is shown in column  $H$  in Table IV, where the number of m-trails derived by the heuristic is seen very closed to that by the ILP.

TABLE IV  
RESULTS FROM THE HEURISTIC METHOD

Networks	No. of SRLGs		Performance metrics					
			$\sum_j m^j$			$T$ ms		
	S	M / H	S	M	H	S	M	H
4 nodes, 6 links	6	10	3	9	8	26	68	94
5 nodes, 8 links	8	18	4	12	11	26	66	94
6 nodes, 9 links	9	44	4	15	15	48	116	138
7 nodes, 12 links	12	96	6	19	20	48	126	152
8 nodes, 12 links	12	132	4	-	27	52	-	228
9 nodes, 14 links	14	179	5	-	24	50	-	164
CERNet	16	236	-	-	32	-	-	258
SmallNet	22	1162	-	-	64	-	-	502
NSFNet + 1 link	22	1162	-	-	93	-	-	832
Bellcore + 1 link	29	2053	-	-	86	-	-	742

VIII. CONCLUSIONS

The paper presented a novel framework of all-optical monitoring for SRLG failure monitoring and localization, called

monitoring burst (m-burst), which aims to achieve a truly all-optical, signaling-free, and deterministic failure localization plane. A novel problem formulation under the proposed m-burst framework was provided, including m-trail allocation and m-burst scheduling. Detailed analysis on the proposed problem was conducted to gain deep understanding on its performance behavior along with its related implementation issues. An ILP was formulated based on the developed theories, which are further exercised as the basis of the developed heuristic approach for large topologies. By conducting extensive experiments, we found that the proposed approach can effectively minimize the maximum number of m-trails traversing through any unidirectional link and the maximum failure localization latencies, and can serve as a competitive complement to the state-of-the-art LMP based failure localization methods.

REFERENCES

- [1] H. Zeng, C. Huang, A. Vukovic and M. Savoie, "Fault Detection and Path Performance Monitoring in Meshed All-Optical Networks," in the proceedings of *IEEE GLOBECOM '04*.
- [2] H. Zeng, C. Huang and A. Vukovic, "Spanning-Tree based Monitoring-Cycle Construction for Fault Detection and Localization in Meshed AONs," in the proceedings of *IEEE International Conference on Communications 2005*, Seoul, Korea, May 16-20, 2005.
- [3] B. Wu and K. L. Yeung, "M<sup>2</sup>-CYCLE: an Optical Layer Algorithm for Fast Link Failure Detection in All-Optical Mesh Networks," in the proceedings of *IEEE GLOBECOM '06*, Dec. 2006, pp. 1-5.
- [4] H. Zeng, C. Huang and A. Vukovic, "A Novel Fault Detection and Localization Scheme for Mesh All-Optical Networks based on Monitoring-Cycles," *Photonic Network Communications*, vol. 11, no. 3, pp. 277-286, May 2006.
- [5] S. Ahuja, S. Ramasubramanian and M. Krunz, "SRLG Failure Localization in All-optical Networks using Monitoring Cycles and Paths," in the proceedings of *IEEE INFOCOM, 2008*, pp. 181-185.
- [6] S. Ahuja, S. Ramasubramanian and M. Krunz, "Single Link Failure Detection in All-Optical Networks using Monitoring Cycles and Paths," *IEEE/ACM Transactions on Networking (TON)*, vol. 17, no. 4, pp. 1080-1093, August 2009.
- [7] J. Tapolcai, B. Wu and P.-H. Ho, "On Monitoring and Failure Localization in Mesh All-Optical Networks," in the proceedings of *IEEE InfoCom 09*, Apr. 2009, pp. 1008-1016.
- [8] 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 Lighthwave Technology*, vol. 27, no. 10, pp. 1392-1401, May 2009.
- [9] B. Wu, P.-H. Ho and K. L. Yeung, "Monitoring Trail: On Fast Link Failure Localization in All-Optical WDM Mesh Networks," *IEEE/OSA Journal of Lighthwave Technology*, vol. 27, no. 18, pp. 4175-4185, Sept. 2009.
- [10] B. Wu, P.-H. Ho, J. Tapolcai and X. Jiang, "A Novel Framework of Fast and Unambiguous Link Failure Localization via Monitoring Trails," in the proceedings of *IEEE Infocom 2010 WIP*.
- [11] J. Tapolcai, P.-H. Ho, L. Ronyai, P. Babarcsi and B. Wu, "Failure Localization for Shared Risk Link Groups in All-Optical Mesh Networks using Monitoring Trails", *Journal of Lighthwave Technology*, vol. 29, no. 10, pp 1597-1606, May 15, 2011.
- [12] Y. Wen, V.W.S. Chan and L. Zheng, "Efficient Fault-Diagnosis Algorithms for All-Optical WDM Networks with Probabilistic Link Failures," *IEEE/OSA Journal of Lighthwave Technology*, vol. 23, no. 10, pp. 3358-3371, Oct. 2005.
- [13] N. Harvey, M. Pătrașcu, Y. Wen, S. Yekhanin and V.W.S. Chan, "Non-Adaptive Fault Diagnosis for All-Optical Networks via Combinatorial Group Testing on Graphs," in the proceedings of the *IEEE INFOCOM 2007*, pp. 697-705.
- [14] M. L. Ali, P.-H. Ho, B. Wu, J. Tapolcai and B. Shihada, "Monitoring Burst (M-Burst) - A Novel Framework of Failure Localization in All-Optical Mesh Networks," in the proceedings of the *8th International Workshop on Design of Reliable Communication Networks, IEEE DRCN 2011*, Krakow, Poland, Oct. 2011.

- [15] D. Papadimitriou and E. Mannie, "Analysis of Generalized Multi-Protocol Label Switching (GMPLS)-based Recovery Mechanisms (including Protection and Restoration)," *RFC 4428*, March, Tech. Rep., 2006.
- [16] H.-B. Guo and G.S. Kuo, "Improvements on Fault Localization in GMPLS-based Networks," in the proceedings of the *2005 Workshop on High Performance Switching and Routing, 2005 HPSR*, pp. 94-99, 12-14 May 2005.
- [17] J.H. Lee, N. Yoshikane, T. Tsuritani and T. Otani, "Optical Link Performance Monitoring using Extended Link Management Protocol for Transparent Optical Networks," in the proceedings of the *Optical Fiber Communication Conference, OFC 2009*, San Diego, CA.
- [18] B. Wu, P.-H. Ho, J. Tapolcai and P. Babarcsi, "Optimal Allocation of Monitoring Trails for Fast SRLG Failure Localization in All-Optical Networks," in the proceedings of *IEEE Global Telecommunication Conference (GLOBECOM) 2010*, pp. 1-5.
- [19] P. Babarcsi, J. Tapolcai and P.-H. Ho, "Adjacent Link Failure Localization with Monitoring Trails in All-Optical Mesh Networks," *IEEE/ACM Transactions on Networking*, vol. 19, no. 3, pp. 907-920, June, 2011.
- [20] R. Yadav and R.R. Aggarwal, "Survey and Comparison of Optical Switch Fabrication Techniques and Architectures," *Journal of Computing*, vol. 2, no. 4, Apr. 2010, ISSN 2151-9617.
- [21] J.W. Suurballe and R.E. Tarjan, "A Quick Method for Finding Shortest Pairs of Disjoint Paths," *Networks*, vol. 14, pp. 325-336, 1984.
- [22] J.Y. Yen, "Finding the K Shortest Loopless Paths in a Network," *Management Science*, vol. 17, no. 11, pp. 712-716, July 1971.

**Mohammed L. Ali** received the B.Sc. in electrical engineering degree from Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh in 1978 and the B.C.S. (Honours) and M.Sc. degrees in computer science from University of Windsor, Windsor, ON, Canada in 2004 and 2007, respectively. He is currently pursuing the Ph.D. degree in David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada.

His research interests include fault localization and routing in communication networks.

**Pin-Han Ho (M10)** joined the Electrical and Computer Engineering (ECE) Department, University of Waterloo, Waterloo, ON, Canada in 2002. He is the author/coauthor of more than 200 refereed technical papers and book chapters, and the coauthor of a book on optical networking and survivability. Dr. Ho is the recipient of the Distinguished Research Excellence Award in the ECE Department, University of Waterloo; the Early Researcher Award in 2005; the Best Paper Award at SPECTS 2002 and IEEE ICC 2005 Optical Networking Symposium; and the Outstanding Paper Award in HPSR 2002.

**János Tapolcai (M09)** received the M.Sc. degree in technical informatics and Ph.D. degree in computer science from Budapest University of Technology and Economics (BME), Budapest, Hungary, in 2000 and 2005, respectively.

Currently, he is an Associate Professor with the High-Speed Networks Laboratory, Department of Telecommunications and Media Informatics, BME. He is an author of over 80 scientific publications. He has been involved in several related European and Canadian projects. His research interests include applied mathematics, combinatorial optimization, mathematical programming, optical and IP level routing and survivability, availability analysis, and distributed computing.

Dr. Tapolcai is the recipient of the Best Paper Award in IEEE ICC 2006 and in DRCN 2011.

**Basem Shihada** received a bachelor's degree in computer science from UAE University, a master's degree in computer science from Dalhousie University, Halifax, Canada, and the Ph.D. degree in computer science from David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada in 1997, 2001 and 2007, respectively.

Currently, he is an Assistant Professor of computer science in Mathematics and Computer Science and Engineering Division, KAUST, Saudi Arabia. He is also a visiting faculty in Stanford Computer Science, Stanford University. His research interests include wired and wireless broadband communication networking, Fiber-Wireless (FIWI) network integration, and optical networking.