Limited-perimeter vector matching fault-localisation protocol for transparent all-optical communication networks†

A.V. Sichani and H.T. Mouftah

Abstract: A novel fault-localisation protocol is constructed based on the principles of distributed control and management mechanisms. The proposed protocol has high scalability and speed, but at the cost of increased computational complexity. To provide the maximum level of transparency, the protocol skips any optical power monitoring or spectrum analysis at the intermediate nodes of established lightpaths. Moreover, to narrow down the associated time and space complexities, it restricts the fault-localisation area to a small area called limited-perimeter. These functions are implemented by means of five phases, namely pausing, flooding, multicasting, matching, and concluding. Although the protocol has been developed to pinpoint single failures, it could track down multiple failures that occur in nonoverlapped limited-perimeters. To evaluate the performance of the proposed method, time and space complexities are calculated and compared with a counterpart protocol that does not limit the fault-localisation area.

1 Introduction

Today interconnections of optical fibres with high capacities respond to increasing demands for more bandwidth. Although many different types of service disruptions, including hard and soft failures with long or very short time intervals may occur in practice, hard failures such as cable cuts or switch crashes are the most frequent and destructive ones. Cable cuts are even more widespread because of their expansion and exposure to environmental damages. Due to transporting a high volume of data, any severed wavelength division multiplexed (WDM) link leads to the loss of several terabits per second. As a consequence, the throughput of the network is harshly degraded and the robustness of system is shattered. Therefore, keeping the networks running unaffected or with reduced effects in the event of link failures is an imperative issue that needs to be carefully addressed.

In general, two types of dynamic restoration techniques, namely link and path restoration, are employed for recovery from link failures [1]. Debates over the pros and cons of dynamic restoration techniques continue as the introduced versions are improved and new versions proposed. However, the following comparisons highlight some of their merits and drawbacks and justify our focus to develop a fast and capable fault-localisation technique to be employed by link restoration.

Link restoration is a point-to-point, while path restoration is an end-to-end recovery technique. Link restoration restores the affected path with generally less number of links than path restoration. Consequently, link restoration decreases the number of to-be-configured switches and the associated restoration delays. In addition, since link restoration keeps a number of working links (segments of the working path) in place, it preserves the network load balancing. In other words, link restoration does not cause a sudden shock to the load balancing of system after failure detection. However, link restoration could create congestion around the failed link. Although this phenomenon could severely alter the protocol functionality, typically it only happens when a network is highly loaded. In addition, link restoration always follows a fault-localisation procedure. The fault-localisation time for a large network could be long and undesirable. However, at the expense of the fault-localisation delay, routing tables are mended and restoration processes are completed faster.

In contrast, path restoration does not require any fault localisation and is started immediately after a fault alarm is detected. However, path restoration could be time-consuming for distant s–d pairs and for highly loaded networks. This is mostly linked to performing configuration procedures and often related to repeating these procedures several times for alternate paths before a restoration path is successfully established. Furthermore, path restoration practically removes the affected path capacity from the network resources by excluding the entire set of affected paths, while link restoration only removes the failed-link capacity by pinpointing the failure location.

Thus we focus on link restoration due to the reviewed merits and in particular because of its high restoration speed. Consequently, in this work we propose a fast fault-localisation scheme to be used as a perquisite in link restoration.

2 All-optical networks

All-optical networks are designed based on different models and control mechanisms. In this section we introduce the employed infrastructure and the applied control mechanism for crafting our network. We distinguish between optical
components that are capable of power monitoring or spectrum analysis and ones that do not have such abilities in the crafted network. All-optical components such as optical amplifiers have limited or no electronic monitoring and analysis abilities. As a result, they may be able to detect loss of signal but cannot manage any fault-localisation procedures. In contrast, there are components capable of detecting failures and taking proper actions in response to service disruptions, for instance optical cross-connects (OXC).

Although fault localisation can be achieved in different layers, for example in the physical layer through electronic processes and using photodiodes and/or spectrum analyser, we focus on the optical layer fault localisation using all-optical components to preserve the high bandwidth of all-optical communications.

2.1 Overlay model

All-optical networks could be created based on different designs such as the overlay, augmented, peer-to-peer, or integrated. The all-optical WDM network architecture considered in this study is the overlay model. In this structural design, optical switches interconnect data links and create the data network, while the control units including E/O/E conversions and optical amplifiers interconnect control links and construct the control network. The data network, consisting of the optical switches and data channels, operates in a circuit-switched fashion, while the control network operates in a packet-switched mode. The traffic in the control network consists of small control packets resulting in much lighter traffic. Therefore the control channel is usually implemented by one or more dedicated wavelengths in the same fibre link. When a connection request arrives, a control packet in the control network routes and configures switches to create a transparent optical path, namely a lightpath. Different criteria are considered and various techniques are employed to set up the most resourceful lightpaths [2]. Unlike the SONET/SDH networks that operate in a point-to-point basis using the peer model, alloptical networks work in an end-to-end basis. Thus these networks could analyse data only at the end-points of set-up lightpaths (sinks).

2.2 Controlling mechanisms

Control mechanisms are primarily developed based on either centralised or distributed models. Despite this fact that centralised control mechanisms are relatively simple and work well for static traffic in small networks, they are considered to be infeasible for dynamic expanding systems. In contrast, distributed control mechanisms are complex but more scalable and reliable than the centralised ones. Thus a distributed control model is employed to manage dynamic traffic in large networks. In addition to these models, there is a hierarchical model that is a combination of centralised and distributed models. Hierarchical models are applied mostly to increasingly large networks and dynamic information systems because of their ability to co-ordinate the network controlling messages. However, hierarchical management models cannot be economically and practically implemented for any network topologies, and researchers are still pursuing distributed rather than hierarchical management structures for mesh networks.

2.3 Optical layer

Optical-layer protection schemes are similar to SONET/SDH techniques. However, their implementation is substantially different. The optical layer consists of the optical channel (OCh) layer, also known as path layer, the optical multiplex section (OMS) layer (line layer), and optical transmission section (OTS) layer [3]. Optical restoration schemes are performed in both the OCh and OMS layers. The OCh restoration schemes deal with lightpaths, while the OMS techniques handle restoration the entire group of affected wavelengths comprising a link (a fibre). Fig. 1 shows different layers that protection techniques can function within. Our proposed technique operates in the OCh-P layer based on the employed structural design, controlling model and definition. We assume that critical factors for having effective fault-localisation procedure in the optical layer have been properly taken care of. These critical factors are as follows: influences due to non-linearities, dispersion, and wavelength-to-wavelength interactions are not wrongly reported as an interruption of service; transporting the optical signals from one provider domain to another, monitoring different signal parameters is done with the same accuracy (interoperability is fully achieved).

3 Fault localisation in optical networks

In SONET the downstream node attached to the disconnected link is able to detect a failure and report it to the network management entity. The fault condition then is communicated with the neighbouring nodes to inhibit them from false alarming by the management. However, fault localisation in SONET requires examining overhead at each node, which slows down the fault-localisation procedure. In optical transport networks (long haul), the fundamental philosophy of SONET frame has been adopted with a more advanced suitable protocol for high WDM rates, known as digital wrapper [4]. This protocol is able to detect 16 errors and corrects eight errors. Although digital wrapper greatly improves the bit error rate (BER), it also consumes bandwidth by ~7% and suffers from the related delay. Although there are a few frameworks on fault protection and restoration for all-optical transparent networks, there are fewer on fault localisation. Introduced fault-localisation protocols consider different aspects of fault localisation such as signalling, alarming, monitoring, detecting and filtering over various topologies. For instance, in Sichani and Mouftah [5], a fault-localisation method named broadcasting fault-detection protocol is proposed that localises failures by propagating fault-detection signals through the supervisory channels. However, the controlling bandwidth usage is considerable. The rolling back protocol was further proposed by Sichani and Mouftah [6] for fast fault localisation, which reduces the number of controlling signals in the supervisory channels. Nevertheless, this protocol decreases the controlling bandwidth usage significantly, its implementation demands adding more monitoring equipment to the network. A work in Hailemariam et al. [7] partitions the network into
subnetworks called islands and discovers a faulty situation ‘a node or link failure’ with island-by-island restoration protocol. Island identification is an offline procedure, executed during network planning and occasionally updated when the network topology is changed. Then it is claimed that outperforms segmented restoration protocol [8] in terms of the time delay, overhead and complexity. To reduce the number of generated failure alarms another approach introduced by Stanic et al. [9], optimises the number of monitoring components using an alarm matrix. Another research work proposes a framework for fault detection using a set of monitoring cycles [10]. A fault-localisation algorithm has been proposed [11], which practically operates in the physical layer. This protocol is capable of localising multiple failures and filtering false alarms. However, the time and space complexity of the protocol could be considerable for large networks. A finite state machine method is proposed in [12], but its complexity for large-scale and dynamic networks impedes its deployment. There is also a proposal on employing GMLPS for fault detection on all-optical networks [13]. A MPLS-based fault-management technique has been introduced by the use of link management protocol (LMP) in [14]. Further extensions to the LMP fault-management procedure has been made in [15]. However, these LMP protocols are intended for use in opaque networks, such as SONET, SDH, and ethernet user ports.

4 Limited-perimeter vector matching fault-localisation protocol

We describe our proposed technique called limited-perimeter vector matching fault-localisation protocol (LVM). We use the mesh network in Fig. 2 for clarifying our descriptions. We assume that each fibre is wavelength multiplexed in the applied network. Then the number of lightpaths traversing a link could be as large as the multiplexing degree. In general, once such a link fails the following scenario will be shaped: A set of sinks connected to set up paths detects signal loss or high signal-to-noise ratio by examining data. Since we assume that the occurred failure for each lightpath is only detectible by the end terminal (at the dropping point of OXC or OADM), the closes sink to the failed link is the one that discovers the failure sooner than others. Let us call this sink node the potential executive-sink. If the related OXC drops the data of more than one lightpaths, then the shortest lightpath among them is selected as ‘potential executive-sink’. The LVM protocol is launched in response to an alarming potential executive-sink as follows:

- Every potential executive-sink reports its status to the network nodes by flooding an alarming signal. The alarm encapsulates the node ID and the path length. As a result, every potential executive-sink receives other potential executive-sink’s information.
- Potential executive-sinks pause for a short predetermined period of time to allow the entire network nodes to reach their steady-state conditions and to make sure they have exchanged information. A proper pausing interval can be defined according to the network characteristics, such as the average distance between s–d pairs and the average propagation delay between neighbouring nodes.
- Potential executive-sinks compete to be selected as an active executive-sink. Each recipient then compares its path length with those of other paths. If its length is the smallest among others, it wins the competition and becomes the executive-sink. Accordingly the executive-sink conducts the fault localisation from now on. To remove the confusion, in case there are several equal path lengths, the protocol also compares IDs. In such a case a sink with the shortest and smallest ID is selected as the executive-sink. A group of potential executive sinks has been illustrated in Fig. 3.
- In the next step the executive-sink creates a vector of its affected links. Let us call this vector affected link-vector (ALV). The executive-sink also defines the limited-perimeter that is an area comprising all the ALV neighbours. Fig. 4 illustrates the ALV nodes and the associated limited-perimeter. The executive-sink multicasts a copy of ALV to the nodes within the limited-perimeter.
- Each ALV recipient makes its link-vector and compares its elements with the ALV vector. The outcome of elements comparisons is presented in a binary vector equivalence to ALV. For an affected lightpath the corresponding cell of the binary vector is set to 1 if the same link is found in the recipient link-vector. In contrast, the corresponding cell of the binary vector is set to 0 if the examined link is not matched any elements of the recipient link-vector. For a working lightpath, the reverse order of bits shows similar conditions. The matching vector including an extra bit showing the recipient status (0 for failed and 1 for working) is reported back to the executive-sink.
- The executive-sink collects the binary vectors from the nodes within the limited-perimeter. It performs logical AND on the binary vectors to define the disconnection. If the executive-sink could not define the disconnection in this stage (if there are more than a single 1 in the binary vector), it could launch the second round of search by extending the limited-perimeter further. For instance, the
limited-perimeter could expand over the neighbours of the considered nodes.

- The executive-sink propagates the location of the failure to the entire network nodes. Network nodes update and amend their routing tables to accelerate ongoing restoration procedures and to make sure upcoming connections will be routed over working links.
- In the last step a fast link-restoration protocol such as loop-back [16, 17] is executed to restore the affected paths.

4.1 Example

We describe the procedure of matching and creating binary vectors in the LVM protocol by means of a mesh network shown in Fig. 1. For the sake of simplicity we assume that only few lightpaths have been established in the network. It is obvious that increasing the number of lightpaths will accelerate the matching process since additional binary vectors are considered. We demonstrate each link by its two end-nodes. Let us assume that link (12-15) is disconnected while the established lightpaths in the network are P1 = f5-12-15-19-20g, P2 = f10-11-12-15-16-25g, P3 = f3-6-9-13-16g, P4 = f11-12-15-16g, P5 = f14-11-12-13-7g, P6 = f14-11-12-15-16-17g, P7 = f4-5-12-15-20-22g, P8 = f5-12-15-20-24-23g, P9 = f17-25-24-23g, and P10 = f12-15-19-18g. Then the disconnection affects several paths, including P1, P2, P3, P4, P5, P7, P8, and P10. Fig. 3 shows the potential executive-sinks in the network. Since path P4 has the shortest length and also the smallest ID, sink 16 wins the competition among all potential executive-sinks and becomes the executive-sink. Accordingly, an area including all the neighbouring nodes of the executive-sink path is specified as limited-perimeter.

Hereafter, only paths within the defined limited-perimeter are involved in the fault-localisation procedure.

As the first step in the matching procedure the ALV vector of executive-sink is formed by sink 16 and disseminated to the limited-perimeter nodes. As a result, all the sinks receive a copy of ALV as follows:

<table>
<thead>
<tr>
<th>11-12</th>
<th>12-15</th>
<th>15-16</th>
<th>ALV</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

If we assume that the ALV recipients respond based on their distance from the executive-sink, then sink 17 would be the first recipient which makes its own link-vector and matches up its elements to the ALV vector.

<table>
<thead>
<tr>
<th>14-11</th>
<th>11-12</th>
<th>12-15</th>
<th>15-16</th>
<th>16-17</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

The related matching vector of this sink is tailored by setting the corresponding cells of an equivalent binary vector to ALV. The path status is attached and the resulting binary vector is sent back to the executive-sink. In practice, only a binary vector of 4 bits is sent back to the executive-sink.

Next, sink 13 generates its link-vector and compares it with ALV. Although this path has not been affected by the failure it should reply to the executive-sink since it has been extended within the limited-perimeter.

14-11   11-12   12-13

Then, after receiving this vector, the executive concludes that link (11-12) has not failed. Later, sink 25 makes its link-vector and compares it with the ALV vector.

10-11   11-12   12-15   15-16   16-25

As a result its mutual links and status is reported back to the executive-sink as follows

<table>
<thead>
<tr>
<th>11-12</th>
<th>12-15</th>
<th>15-16</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Then sink 20 generates its link-vector to discover the common terms with ALV.

5-12   12-15   15-19   19-20

The following matching vector is disseminated to the executive-sink:

<table>
<thead>
<tr>
<th>11-12</th>
<th>12-15</th>
<th>15-16</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Although executive-sink is still receiving binary vectors from other sinks (P7, P8, P10), it is able to identify the failed link (12-15) at this stage.

111-12   12-15   15-16
0        1        0

5 Performance evaluation

We evaluate the proposed protocol in terms of time and space complexities. To simplify the calculations we divide the fault-localisation procedure to the following distinct phases:

(i) flooding phase
(ii) pausing phase
(iii) multicasting phase
(iv) matching phase
(v) concluding phase

In the flooding phase, potential executive-sinks broadcast their information to the network nodes. The associated
flooding time interval is given by

$$T_1 = D\alpha$$  \hspace{1cm} (1)

where $D$ stands for the network diameter based on the number of hops and $\alpha$ is the average propagation delay between neighbouring nodes. In phase (ii), potential executive sinks pause for a very short interval time to make sure all the network nodes have exchanged information. Note that by the end of the flooding phase all the sinks have reached their steady-state conditions. Accordingly, at the end of this phase, a sink with the shortest path and lowest ID is selected to conduct the LVM fault-localisation protocol. The related pausing time interval can be determined by the average distance between source–destination pairs, that is $E[d_{sd}]$ and the average propagation delay.

$$T_2 = E[d_{sd}]\alpha$$  \hspace{1cm} (2)

The multicasting phase delay is also related to $E[d_{sd}]$ since the longest path within the limited-perimeter comprise at most two additional links (hops) to the shortest path. Therefore the associated delay equals

$$T_3 = (E[d_{sd}] + 2)\alpha$$  \hspace{1cm} (3)

The order of computational complexity of matching two vectors with lengths $L_1$ and $L_2$ is $O(L_1L_2)$. Therefore the computational complexity of comparing each two vectors of matching phase is of order of $O(d_{sd}^2d_{sd})$. Thus the computational complexity in average is of order of $O(E[d_{sd}]^2)$. If we assume that all recipients perform this procedure concurrently, the related delay for phase (iv) is proportional to the computational complexity as follows

$$T_4 = E[d_{sd}]^2\beta$$  \hspace{1cm} (4)

where $\beta$ represents the computational time cycle.

The last phase that the executive-sink logically AND binary matching vectors could be completed in a single time cycle, $\beta$. Therefore the accumulated time delay for the defined phases is

$$T_{LVM} = \sum_{i=1}^{5} T_i = E[d_{sd}]\alpha + D\alpha + (E[d_{sd}] + 2)\alpha + E[d_{sd}]^2\beta + \beta$$  \hspace{1cm} (5)

Let us now calculate the time complexity based on a matrix-formed mesh network with $M + 1$ by $N + 1$ nodes. The distance between any source–destination pairs could be presented as follows

$$d_{sd} = |x_s - x_d| + |y_s - y_d|$$  \hspace{1cm} (6)

where $(x_s, y_s)$ and $(x_d, y_d)$ are the $x$–$y$ co-ordinates of the source and destination, respectively. We assume that the co-ordinates of a source and a destination are independent and also $x$ and $y$ co-ordinates of them are independent uniform random variables distributed. Thus for a mesh matrix-formed network of $(M + 1)(N + 1)$ nodes, we have

$$P(x_s = m) = P(x_d = n) = \frac{1}{M + 1}, \hspace{0.5cm} \forall m, n = 0, 1, \ldots, M$$

$$P(y_s = p) = P(y_d = q) = \frac{1}{N + 1}, \hspace{0.5cm} \forall p, q = 0, 1, \ldots, N$$  \hspace{1cm} (7)

Then the expected value of the distance is given by

$$E[d_{sd}] = E[|x_s - x_d|] + E[|y_s - y_d|]$$  \hspace{1cm} (8)

To calculate the average distance we use the distribution of $|x_s - x_d|$ and $|y_s - y_d|$ as follows, respectively

$$P(|x_s - x_d| = i) = \sum_k P(x_s = k)P(x_d = i - k)$$

$$P(|y_s - y_d| = j) = \sum_k P(y_s = k)P(y_d = j - k)$$  \hspace{1cm} (9)

where $k$ in the summations take all possible values. We easily get

$$P(|x_s - x_d| = i) = \frac{2(M + 1 - i)}{M(M + 1)}, \hspace{0.5cm} \forall i = 1, 2, \ldots, M$$

$$P(|y_s - y_d| = j) = \frac{2(N + 1 - j)}{N(N + 1)}, \hspace{0.5cm} \forall i = 1, 2, \ldots, N$$  \hspace{1cm} (10)

Therefore we can get the following results

$$E[|x_s - x_d|] = \sum_{i=0}^{M} i \left(\frac{2(M + 1 - i)}{M(M + 1)}\right) = \frac{M + 2}{3}$$  \hspace{1cm} (11)

$$E[|y_s - y_d|] = \frac{N + 2}{3}$$  \hspace{1cm} (12)

$$E[d_{sd}] = \frac{M + N + 4}{3}$$  \hspace{1cm} (13)

Now that we calculated the expected value of the distance between source–destination pairs we are able to present the time complexity based on $M$ and $N$

$$T_{LVM} = \sum_{i=1}^{5} T_i = (M + N)\alpha + \left(\frac{M + N + 7}{3}\right)2\alpha + \left(\frac{M + N + 4}{3}\right)^2\beta + \beta$$  \hspace{1cm} (14)

If we assume that $\beta \geq \alpha$ the calculated delay is mainly related to the quadratic term. However, $\alpha$ for some networks could be bigger than $\beta$. Then we should investigate the effect of varying these variables on $T_{LVM}$ to find the dominant term. To reduce the number of variables let us assume that $M = N$ and $\beta = 1$. We vary $\alpha$ based on $N$, that is $\alpha = N$, $2N$, $\ldots$. Fig. 5 illustrates the results of the comparison. The drawing curves reveal that when $\alpha = 2N = M + N$ both terms have the equal effect on the delay (note that for a matrix-formed network of $M = N = 5$, $\alpha$ is calculated 10 when $\beta$ is 1) but as $\alpha$ increases, its related delay becomes dominant.

On the other hand, the space complexity can be calculated by defining two phases: the multicasting phase, and the matching phase. In the first phase, nodes within limited-perimeter received a copy of ALV. Assuming that $m$ bits represent each link, we have

$$S_1 = mE[d_{sd}]$$  \hspace{1cm} (15)

In the matching phase, the executive-sink receives matching vectors with the ALV length from $n$ recipients. Then

$$S_2 = nE[d_{sd}]$$  \hspace{1cm} (16)
removing possible uncertainties in the case of having multiple failures surrounded by a limited-perimeter. It is clear that according to the second condition the time interval between two failures have to longer than the time taken to complete the first failure-pausing phase. Since in practice the actual time interval between two failures is not zero and also for the reason that time taken for defining a limited-perimeter is very short, we infer that the LVM protocol could efficiently respond to multiple failures.

6 Concluding remarks

We have proposed a novel fault-localisation protocol applicable to all-optical networks. The proposed protocol namely, limited-perimeter vector matching fault-localisation protocol (LVM), is highly efficient due to operating in the optical layer and because of bypassing electronic processes. The protocol could be implemented in five phases: first, potential executive-sinks flood their information through the network; secondly, after a short pausing period, the connected sink to the shortest affected path is identified to conduct the fault-localisation protocol as the executive node; thirdly, based on the executive-sink path nodes, a relatively small region called limited-perimeter is defined; then the executive-sink information (affected-link-vector) is disseminated to the nodes within the limited-perimeter; fourthly, each recipient compares its own link-vector with ALV and sends its matching results via an ALV correspondence binary vector back to the executive-sink; in the last phase, the executive-sink, through performing logical AND on the received binary vectors, discovers the failed link. We evaluated LVM in terms of time and space complexities against a similar but boundless centralised protocol. The analytical results reveal that the time and space complexities are less than those of the counterpart protocol. The difference in performance grows as the network is expanded to a larger one via increasing the nodes (the diameter) or/and the distance (the average propagation delay). These outcomes were expected since LVM dynamically defines and limits the fault-localisation area. In addition to providing a lower computational complexity the protocol is able to pin down and locate multiple failures if they occur in nonoverlapped limited-perimeters. Thus we conclude that LVM is a powerful fault-localisation method with remarkable advantages for transparent all-optical networks.

7 References

14 Lang, J.: ‘The link management protocol (LMP)’. Internet Engineering Task Force RFC 4204, 2005