HEURISTIC ALGORITHM FOR FAULT DETECTION AND PATH PERFORMANCE MONITORING IN MESHED ALL-OPTICAL NETWORKS

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology
In
Computer Science and Engineering

By
SOUMYARANJAN KAR

Department of Computer Science and Engineering
National Institute of Technology
Rourkela
2007

HEURISTIC ALGORITHM FOR FAULT DETECTION AND PATH PERFORMANCE MONITORING IN MESHED ALL-OPTICAL NETWORKS

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
Bachelor of Technology
In
Computer Science and Engineering

By
SOUMYARANJAN KAR

Under the Guidance of
PROF. B. D. SAHOO

Department of Computer Science and Engineering
National Institute of Technology
Rourkela
2007

National Institute of Technology

Rourkela

CERTIFICATE

This is to certify that the thesis entitled **“Heuristic algorithm for fault detection and path performance monitoring in meshed all-optical networks”** Submitted by **Soumyaranjan Kar, Roll No: 10306010** in the partial fulfillment of the requirement for the degree of **Bachelor of Technology** in **Computer Science Engineering**, National Institute of Technology, Rourkela , is being carried out under my supervision.

To the best of my knowledge the matter embodied in the thesis has not been submitted to any other university/institute for the award of any degree or diploma.

Prof. B. D. Sahoo
Department of Computer Science
and Engineering
National Institute of Technology
Rourkela-769008

Date:

ACKNOWLEDGEMENT

We avail this opportunity to extend our hearty indebtedness to our guide **Prof. B. D. Sahoo** , Computer Science Engineering Department, for their valuable guidance, constant encouragement and kind help at different stages for the execution of this dissertation work.

We also express our sincere gratitude to **Dr. S.K.JENA**, Head of the Department, Computer Science Engineering, for providing valuable departmental facilities.

Submitted by:

Soumyaranjan Kar
Roll No: 10306010
Computer Science Engineering
National Institute of Technology
Rourkela

CONTENTS

Page

No

ABSTRACT

Fault detection is critical for all-optical networks (AONs). This paper introduces the concept of monitoring cycles and proposes a mechanism for fault detection and path performance monitoring based on decomposing AONs into monitoring cycles. Two monitoring cycle finding algorithms are developed: heuristic depth first searching (HDFS) and shortest path Eulerian matching (SPEM). The two algorithms are compared in terms of wavelength overhead in nodes and links. It is shown that the proposed fault detection mechanism based on monitoring cycles is effective and cost efficient.

Heuristic depth first searching (HDFS):

1) Given graph $\boldsymbol{G(V, E)}$, let the cycle cover $\boldsymbol{C = null}$; number all nodes in *V*; and label all nodes in ***V*** and all links in ***E*** as "**uncovered**";

2) Select an uncovered link ***e*** in E, if multiple such links are available; select the uncovered link whose endpoints are also uncovered. Start DFS from *e* and go to that Uncovered endpoints of e if possible; if no uncovered link with uncovered endpoint is available, apply the largest/smallest rule described below;

3) At each step of the **DFS**, select an uncovered link if possible. If multiple links are available, alternatively use the largest/smallest node number first rule in the iteration, e.g. if the last time we selected the node with the largest number among multiple nodes with the same priority, then this time we select the node with the smallest number;

4) Once a link returns to the previously visited part, a cycle *c* can be formed and add the cycle to the cover ***C***; label all the links and nodes in cycle ***c*** as "**covered**";

5) Repeat (2)-(4) until all links in *E* are "covered".

Shortest path Eulerian matching (SPEM):

(1) For a non-Eulerian graph $\boldsymbol{G\ (V, E)}$, find the set $\boldsymbol{V'}$ of odd-degree nodes;

(2) Start from a node $\boldsymbol{x \in V'}$ and find the shortest path to every other node, select the smallest one among them, denote as $\boldsymbol{p(x, y)}$**.** Add path $\boldsymbol{p(x, y)}$ to G (now some links in G *are* "doubled") and remove x, y from V' ;

(3) Repeat (2) until $\boldsymbol{V' = null}$. Now $\boldsymbol{G\ (V, E)}$ is Eulerian;

(4) Find an Eulerian cycle of the augmented $G(V, E)$ and decompose it to a cycle cover as mentioned above.

This paper introduced the concept of monitoring cycles and proposed a fault detection and path performance monitoring mechanism based on decomposing AONs into monitoring cycles. The heuristic depth first searching (HDFS) and shortest path Eulerian matching (SPEM) algorithms are developed for finding monitoring cycles in AONs. The two algorithms are compared with respect to the maximum and average number of wavelengths occupied by monitoring in nodes and links. The results for the 4 network examples show that the wavelength overhead is pretty low with this mechanism. Thus the proposed mechanism based on monitoring cycles is a promising fault detection method for AONs. It is also applicable to path transmission performance monitoring. The results also suggest that the SPEM algorithm is better than the HDFS algorithm in terms of the wavelength overhead

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

1. INTRODUCTION

1.1. Fault Detection

With the development and deployment of dense wavelength division multiplexing (DWDM) and all-optical networking technologies, telecommunication transmission networks continue to evolve towards higher data rates and increased wavelength numbers and density. It greatly improves the data transmission efficiency but, at the same time, even a very short disruption of service caused by a network fault may lead to a very high data loss in such networks. Consequently the network function for monitoring and fault detection is critical for such networks.

There are numerous fault detection mechanisms for traditional electrical communication networks. Unfortunately, such mechanisms cannot be applied directly to all-optical networks (AONs) due to the lack of electrical terminations in AONs. Even some detection methods deployed in optical networks with opto-electro-opto (OEO) conversion cannot be adapted to AONs.

1.2. Algorithm History

Optical power detection, optical spectral analysis, pilot tone and optical time domain reflectometry can be deployed for fault detection and are also applicable to attack detection in AONs. Y**. Hamazumi,** M. **Koga,** K**. Kawai,** H. **Ichino,** K**. Sato**, developed a fault detection scheme by assigning monitors to the sinks of each optical multiplex section and optical transmission section. A heuristic algorithm was proposed to efficiently assign monitors and thus reduce the required number of monitors. This kind of scheme is channel-based and introduces large numbers of monitors thus it is not feasible in current AONs due to channel dynamics, scalability and costs factors. Other methods, e.g. a finite state machine described were proposed but their complexity for large-scale and dynamic networks impedes their deployment.

Most routing protocols, e.g. OSPF and IS-IS, also have inherent fault detection functionality. Some key OSPF parameters were optimized by M. Goyal, K. K. Ramakrishnan, and W.-C. Feng to achieve fast fault detection . A joint optical and IP layer method was proposed by C. Assi, Y. Ye, A. Shami, S. Dixit, and M. Ali to accelerate the detection speed. Usually the fault detection time of routing protocols is in the seconds range, even with some accelerating techniques. However, the typical time constraint for fault recovery in optical networks is 50 milliseconds. This constraint inhibits moving the fault detection from the optical layer to the IP layer. Thus some effective and efficient fault detection mechanisms at optical layer are still expected.

In this paper, we propose a mechanism for fault detection and path performance monitoring in AONs that utilizes dedicated wavelengths as supervisory channels. The monitors are assigned based on cycle covers of the network topology. Two cycle cover finding algorithms are compared in terms of the cycle numbers and length, as well as the maximum and average number of occupied wavelengths in nodes and links, for four typical networks: NSFNET, ARPA2, SmallNet and Bellcore.

Chapter 2

LITERATURE REVIEW

What's an AON?
Multiplexing techniques
Wavelength division multiplexing
Application of AONs
An example

2. LITERATURE REVIEW

2.1 What's an AON?

An all-optical network (AON) is a network that uses lightwave communication exclusively within the network. More precisely, in an AON all network-tonetwork interfaces are based on optical transmission, all user-to-network interfaces use optical transmission on the network side of the interface, and all switching and routing within AON network nodes is performed optically. The principal advantage of maintaining an optical network core in comparison to using electro-optic components at nodes or in transmission systems is higher bandwidth: optical bandwidths are generally one thousand fold those of electronic bandwidths, and avoiding optical/electronic/optical conversions therefore promises roughly one thousand times greater data rates than possible with electro-optic networks. Transparency is an optical network feature that allows routing and switching of data within the network without interpretation or regeneration of the individual data streams. While transparency has many desirable features (e.g. terminal upgrades do not require network upgrades), it has important ramifications for security.

2.2 Multiplexing techniques:

Optical fiber can provide a large bandwidth, approximately 24 THz. To share this bandwidth, various multiplexing techniques have been proposed for AONs. These techniques include Wavelength Division Multiplexing (WDM), Optical Time Division Multiplexing (OTDM), and Optical Code Division Multiplexing (OCDM). Recently, hybrid-multiplexing techniques such as WDM/OTDM and WDM/OCDM have also been proposed for all-optical networks. A brief idea about WDM is given below.

2.2a Wavelength Division Multiplexing WDM

In WDM, two or more optical signals having different wavelengths are combined and simultaneously transmitted in the same direction over an optical fiber. WDM is a rate and format independent technology, and it can support any combination of interface rates including synchronous or asynchronous Optical Channel (OC) OC–3, OC–12, OC–48, or OC–192 on the same fiber at the same time. It is already at an advanced stage of development and WDM networks can be deployed using commercially available components and systems. There are three variations of WDM: Narrowband WDM (NWDM), Wideband WDM (WWDM) and Dense WDM (DWDM). Typically, NWDM is implemented by using two wavelengths: 1533 and 1577 nanometers (nm). WWDM, on the other hand, is 3 implemented by combining a 1310 nm wavelength with another wavelength into the lowloss window of an optical fiber cable between 1528 nm and 1560 nm in wavelength. Technically, WDM and DWDM are similar, however, as the name implies, DWDM supports many more wavelengths. The number of wavelengths that a DWDM system can support depends on the ability of the system to accurately filter and separate them. Initial implementations of DWDM systems support either 8 or 16 wavelengths. However, current DWDM systems are capable of supporting 32 or 40 wavelengths. Recently, DWDM systems capable of supporting as many as 80 and 128 wavelengths have been announced.

2.3 Application of AONs

There is a wide range of potential AONs applications (Figure 1) in commercial, government, scientific, and academic arenas. In the near-term, most of these applications can be individually supported by electronic or electro-optic networks. However, the aggregation of many services and the cost, flexibility, and transparency supported by

AONs may prove superior to electronic or electro-optic networks. The range of applications can be classified into three categories. The first category contains applications utilizing traditional digital services. The data rates of these applications range from Kb/s to Gb/s. They can be further classified into four subcategories that include applications requiring Gb/s, fast-packet switching such as ATM, visualization, simulation, and supercomputer interconnects. The second covers applications requiring 100 Mb/s data such as LAN interconnections, compressed High Definition Television (HDTV), digitized conventional video, and workstation interconnects. The third consists of 10 Mb/s applications such as multi-channel digital audio and Ethernet class computer networks, and the fourth subcategory covers applications requiring 1 Mb/s or less such as telephone services and RS-232 class.

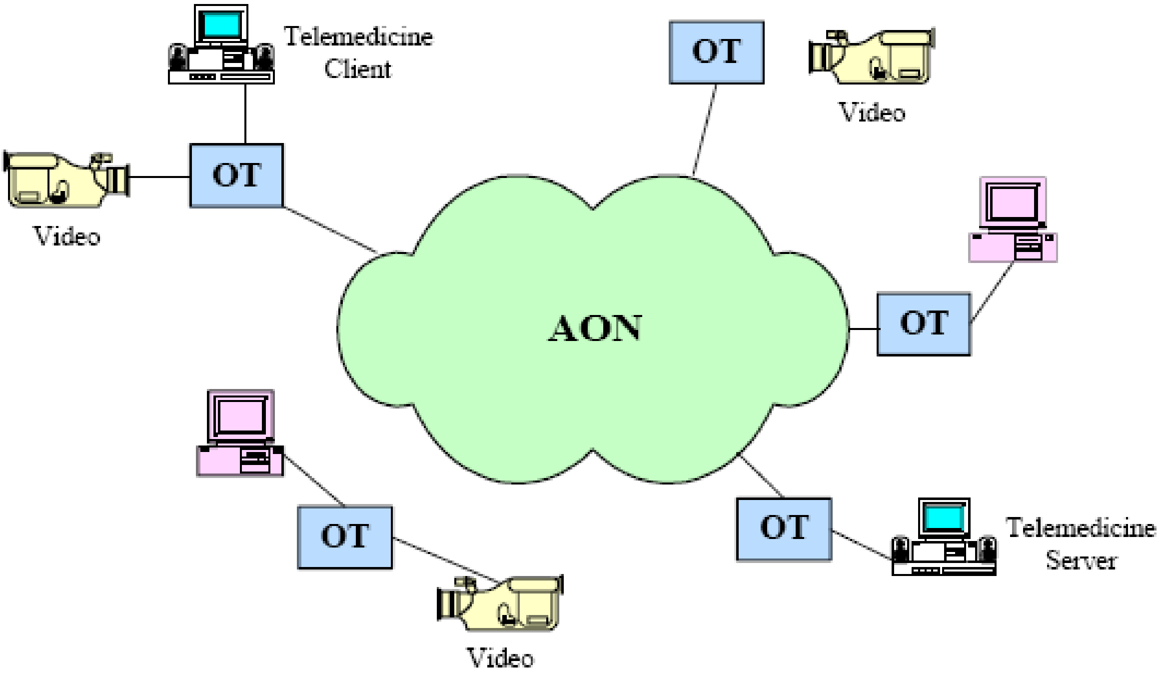


Figure 2.1: AON Applications

2.4 Example of an AON

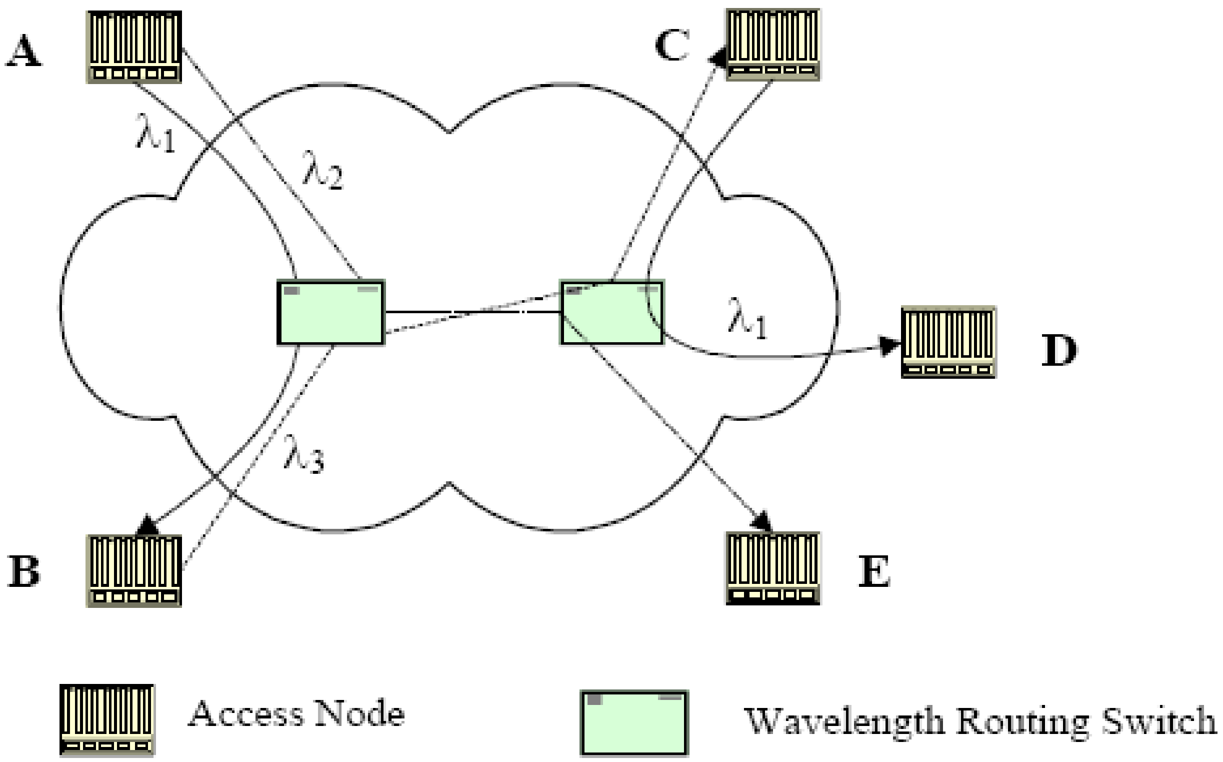


Figure 2.2: Bellcore's Multi-wavelength All-Optical Network Architecture

As shown in Figure 1, the network architecture consists of an all-optical inner portion that contains wavelength-routing switches. According to the wavelength-continuity constraint for wavelength-routed networks, two lightpaths (a.k.a. optical communication channels) that share a common fiber link should not be assigned the same wavelength. In the Figure, λ1, λ2, and λ3 are available wavelengths of the network. Wavelength λ1 is being used in two lightpaths for A-B and C-D connections since the two paths use 15 different wavelength-routing switches. λ2, and λ3 can occupy the path between wavelength-routing switches because they are different frequencies. However, if a switching or routing node is also equipped with a wavelength converter, then the wavelength-continuity constraint disappears, and a lightpath can be switched between different wavelengths on its route from its source to its destination. For this reason, routing and wavelength assignment is a major challenge in wavelength-routed networks without wavelength converters. Further, the lack of wavelength conversion increases the probability of connection blocking because the same wavelength may not be available at all nodes for a particular lightpath on its route from its source to its destination. Wavelength converters are very expensive and are not used at all nodes in the network. Bellcore’s architecture uses wavelength converters only at selected nodes in its network.

Chapter 3

MONITORING CYCLES

Definition

Feasibility

3. MONITORING CYCLES

3.1 Monitoring Cycles

In AONs the fault impact scopes are various in terms of wavelengths (channels or paths). Some faults only affect a single or some specific wavelengths, e.g. transmitter laser failure, optical crossconnect (OXC) port blocking, etc. Others may affect all the wavelengths passing through the faulty module, e.g. fibre cut, optical amplifier (OA) saturation, etc. The characteristics of AON faults are strongly related to the specific network components [8]. For AONs, faults affecting specific wavelengths usually can be handled by in-place fault control or management entities. On the other hand, network faults affecting all wavelengths put an impact on more user traffic and generate much more alarms in the network, thus degrade the network performance more severely. Therefore it is critical to develop some mechanisms to address the latter kind of network faults.

In this section we describe a scheme for detecting the network node or link faults that affect all lightpaths passing through the node/link. The main idea is to decompose a network into cycles. All nodes and links of the given network are covered at least by one cycle. We define these cycles as "***monitoring cycles***". A pair of transmitter and receiver is assigned to one node in each monitoring cycle and thus a loopback dedicated supervisory channel (using an independent wavelength) is set up. Such a mechanism can achieve fault detection and path performance monitoring in AONs.

3.2 Feasibility

A network can be modeled as a finite undirected graph G (V, E), where V is the set of vertices (network nodes) and E is the set of edges (network link). A *cycle* (denoted as c)

of the graph G is a sub-graph of G, which is connected and regular of degree two. A cycle is often identified with its edge-set. A *cycle cover* (denoted as *C*) of a graph is a family of cycles in which each vertex and edge of the graph appears at least in one of these cycles. Figure 1 gives an example of a network and an instance of its cycle cover. This cover consists of 4 cycles. Some nodes and links appear only in one cycle of the cover.

For example node *c*, link *bc* and *cg*, are covered by cycle *(4)* only. But some others appear in multiple cycles, e.g. edge *bd* is covered by both cycle *(1)* and *(4)*, node *f* by cycle *(2)*, *(3)* and *(4)*.

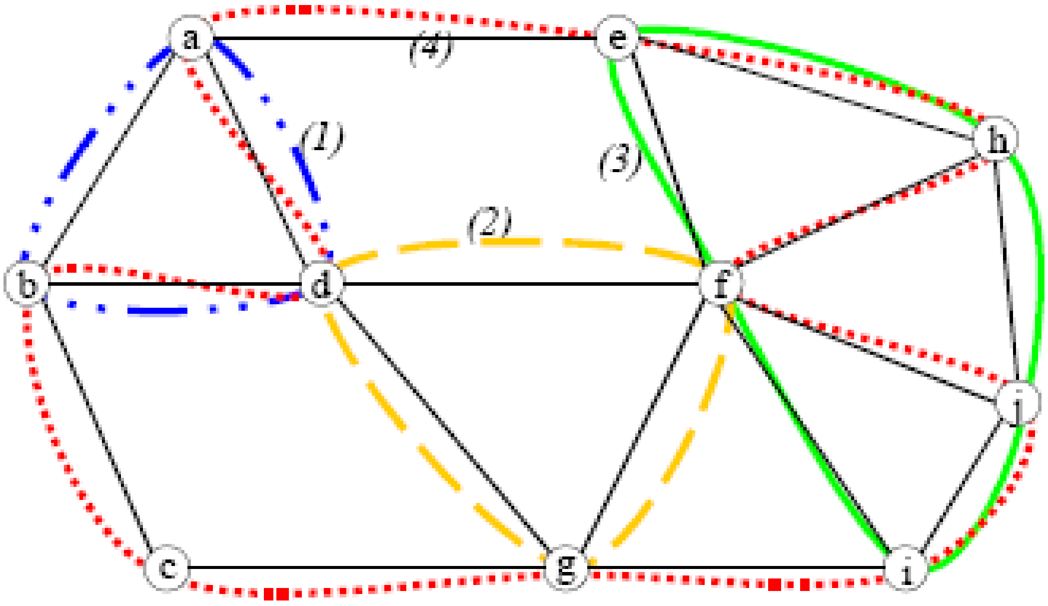


Figure 3.1: A graph example and its cycle cover

To achieve fault detection and path performance monitoring for all nodes and links of a given network, a cycle cover has to be found. At the same time, for maximizing the network resource utilization, we also have to minimize the number of wavelengths occupied by monitoring channels in nodes and links. That is to say, the goal is to find a cycle cover C for graph G (V, E) that minimizes the number of each node and link's occurrence in all monitoring cycles. C will be called a ***cycle double cover*** if every edge appears in exactly two of those cycles of C. A minimum counterexample to the cycle double cover conjecture must be a snark that has girth at least seven. A snark is a cyclically 4-edge-connected cubic graph of girth at least five. It is worth to note that no snark of girth at least seven is known. In fact, some literatures had conjectured that such snarks did not exist. Thus it is safe to say, even if the counterexamples to the conjecture do exist, it is not expected that communication networks with such topologies will be encountered in the real world.

The cycle double cover conjecture not only shows the feasibility of the setup of monitoring cycles, but also gives a reference for evaluating the performance of monitoring cycle finding algorithms in terms of the network resource utilization. That is, an achievable reference of the average number of occupied wavelengths is two for monitoring in all links.

In the following sub-sections we describe two heuristic algorithms for finding a cycle cover in given graphs: the heuristic depth first searching [**HDFS**] algorithm and the shortest path Eulerian [**SPEM**] matching algorithm.

Chapter 4

Heuristic depth first searching (HDFS)

4. Heuristic depth first searching (HDFS)

Given a graph $\boldsymbol{G}\ (\boldsymbol{V}, \boldsymbol{E})$, starting from any node $\boldsymbol{n} \in \boldsymbol{V}$, we can traverse all links in $\boldsymbol{E}$ by depth first searching (**DFS**). Let the traversed part of G be $\boldsymbol{G'}\ (\boldsymbol{V'}, \boldsymbol{E'})$ during the DFS. While a link *e* from node *x* to node *y* being traversed, if $\boldsymbol{y} \in \boldsymbol{V'}$, then there must exist a path $\boldsymbol{p}(\mathbf{y}, \cdot\cdot\cdot, \boldsymbol{x}) \in \boldsymbol{G'}$. Thus path $\boldsymbol{p}(\mathbf{y}, \cdot\cdot\cdot, \boldsymbol{x})$ and link $\boldsymbol{e}(\boldsymbol{x}, \boldsymbol{y})$ consist a cycle. Based on this fact, a heuristic cycle cover finding algorithm is developed as below,

4.1 HDFS: The Algorithm:

1) Given graph $\boldsymbol{G}\ (\boldsymbol{V}, \boldsymbol{E})$, let the cycle cover $\boldsymbol{C} = \boldsymbol{null}$; number all nodes in ***V***; and label all nodes in *V* and all links in ***E*** as **"uncovered"**;

2) Select an uncovered link **e** in **E**, if multiple such links available, select the uncovered link whose endpoints are also uncovered Start **DFS** from **e** and go to that uncovered endpoint of **e** if possible;

3) At each step of each **DFS**, select an uncovered link. If multiple links available, alternatively apply the largest/smallest node number first rule, e.g. if in last **DFS** we select a link whose end-node has the largest number among multiple nodes with same priority, then in the current **DFS** we select the link whose end-node has the smallest number;

4) Once a link returns to the previously visited part, a cycle ***c*** can be formed and added to the cover ***C***; label all the links and nodes in cycle ***c*** as "**covered**";

5) Repeat (**2**)-(**4**) until all links in ***E*** are **"covered"**.

The selection of starting link tries to avoid covering a link with many different cycles in the cover. The alternative largest/smallest numbered node first rule distributes cycles evenly among nodes and links. Both heuristic rules therefore avoid occupying large number of wavelengths for monitoring.

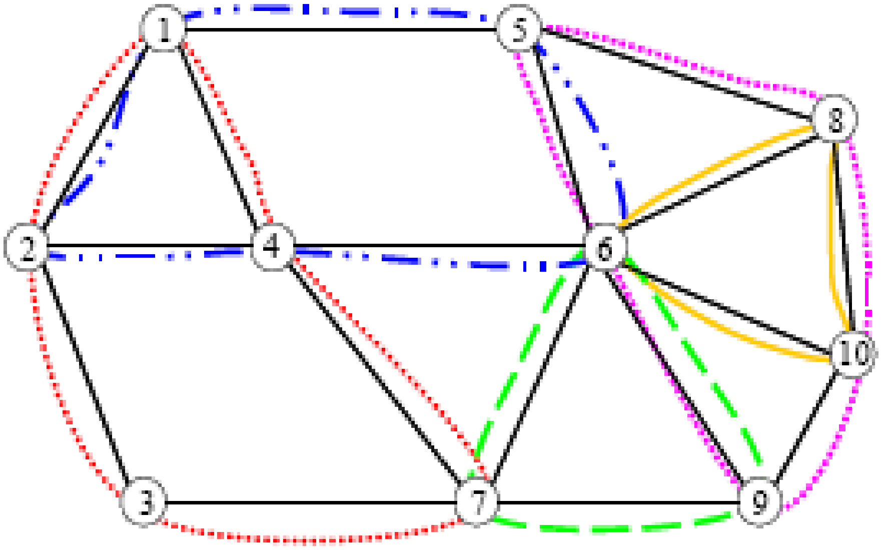


Figure 4.1: Cycle covers obtained by HDFS

Fig. 4 describes the results of HDFS applying to the graph example given in Fig.3. Before starting the DFS, the nodes are numbered from 1 to 10. All nodes and links are labeled as "uncovered" and set *C* = *null*. During the DFS the following iterations are executed,

Iteration 1: start from node 1 and there are 3 "uncovered" links: ***(1, 2), (1, 4)*** *and* ***(1, 5).*** Applying the smallest numbered node first rule, we select link ***(1, 2)***. After a **DFS** iteration the cycle ***1-2-3-7-4-1*** is obtained and added to ***C***. Nodes in set ***{1,2,3,7,4}*** and links in set ***{(1,2), (2,3), (3,7), (7,4), (4,1)}*** are labeled as "**covered**".

Iteration 2: start from node **5** and there are **3** "**uncovered**" links: ***(5, 1), (5, 6)*** *and* ***(5, 8)***. Since node ***6*** and ***8*** are uncovered, link ***(5, 6)*** and ***(5, 8)*** are prior to ***(5, 1).*** Alternatively to iteration 1, we apply the largest numbered node first rule and select link ***(5, 8)***. After the

DFS the cycle ***5-8-10-9-6-5*** is obtained and added to *C*. Nodes in set ***{5, 8, 10, 9, 6}*** and links in set ***{(5, 8), (8, 10), (10, 9), (9, 6), (6, 5)}*** are labeled as "**covered**".

Similarly, cycles ***6-4-2-1-5-6***, ***8-6-10-8***, and ***6-7-9-6*** are obtained and added to ***C*** in the remainder iterations. After **iteration 5** all links in the graph are covered and a 5-cycle cover is obtained as depicted in **Fig. 4**.

Chapter 5

SHORTEST PATH EULERIAN MATCHING (SPEM)

5. SHORTEST PATH EULERIAN MATCHING (SPEM)

For an Eulerian graph, there exists an Eulerian cycle that covers all links once. If we traverse the Eulerian cycle by following links in it until a node is re-visited, the traversed part forms a sub-cycle. Then we remove this part from the Eulerian cycle and traverse the remainder part until all links are removed. In this way, the Eulerian cycle is decomposed into a cycle cover *C* consisting of all sub-cycles obtained in the traversing. Due to the fact that no two cycles in *C* have a common link; the minimum number of monitoring wavelengths incident to each link can be achieved.

Euler proved that a graph is Eulerian if and only if every node has an even degree. Thus a non-Eulerian graph has some nodes with odd degrees. Since each link connects two nodes, the total number of odd-degree nodes is even. We can augment the given graph to construct an Eulerian graph by adding links between pairs of odd degree nodes, i.e. Eulerian matching. In the matching each added new link corresponds to a path consisting of existing links between the node pair in the original graph. Links included in one augment will be covered one more time in a cycle cover. To minimize the average number of wavelengths occupied by monitoring in links, i.e. minimize the average link cover times, the shortest path augments between odd-degree node pairs are added. This Heuristic shortest path Eulerian matching (SPEM) is described below,

5.1. The Algorithm

(1) For a non-Eulerian graph ***G* (*V, E*)**, find the set $\boldsymbol{V'}$ of odd-degree nodes;

(2) From $\boldsymbol{V'}$, start from a node ***x*** and find the shortest path to every other node, select the smallest one among them, denote as ***p*(*x, y*)**. Add the path ***p*(*x, y*)** to ***G*** (now some links in ***G*** are "**doubled**") and remove ***x, y*** from $\boldsymbol{V'}$;

(3) Repeat **(2)** until $\boldsymbol{V' =}$ ***null***. Now ***G* (*V, E*)** is **Eulerian**;

(4) Find an **Eulerian cycle** of the augmented ***G* (*V, E*)** and decompose it to a cycle cover as abovementioned .

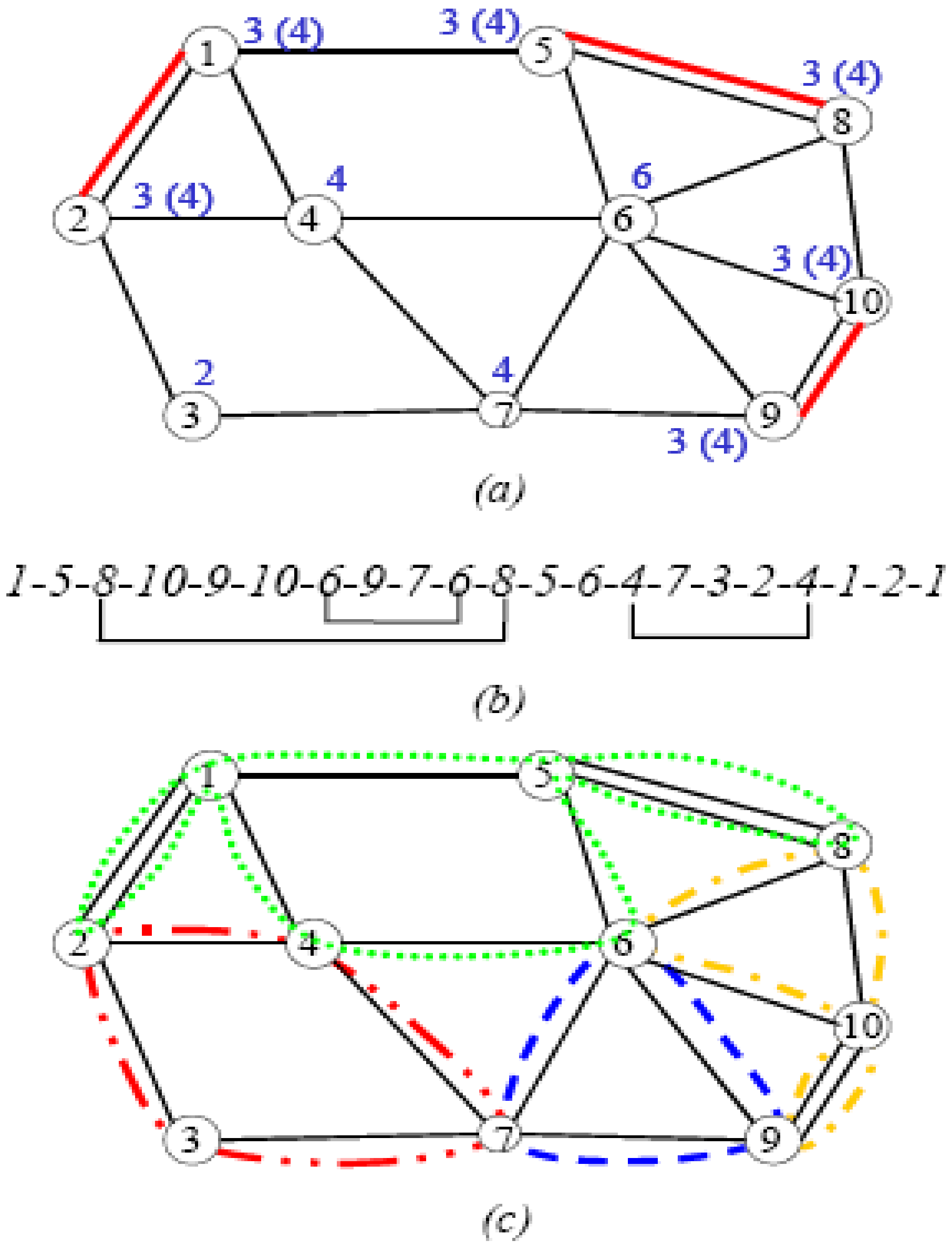


Figure 5.1: The graph example for SPEM:

(a) The shortest path Eulerian matching;(b) An Eulerian cycle and decomposing;

(c) The cycle cover obtained by SPEM

For example, in the graph given in Fig. 3 we firstly label the degree for all nodes and the odd-degree node set is ***{1, 2, 5, 8, 9, 10}.*** The degrees of all nodes are labeled in **Fig. 5(a).** For **node 1**, the shortest path to another node is ***(1, 5)*** and ***(1, 2)***, which are a single hop. Select ***(1, 2)*** and remove node ***1, 2*** from the odd-degree node set. Repeatedly we get the matching path set **{*1-2, 5-8, 9-10*}** (total length is 3) as shown in Fig. 5(a). If we select **(1, 5)** at the first step, the matching path set would be ***{1-5, 2-3-7-9, 8-10}***. The total length is 5, larger than the first matching path set and thus was dropped. In this way by enumerating all possible tiers at each step, we can get the shortest path matching. The node degrees changed by the matching paths are labeled in the brackets in Fig. 5(a). Now all node degrees are even and the augmented graph is Eulerian. An Eulerian cycle can be found by any existing traversing algorithm, such as DFS. In this example an Eulerian cycle is listed in **Fig. 5(b).** This Eulerian cycle is decomposed into 4 cycles as shown in **Fig. 5(c).** Note that a two-edge cycle, e.g. ***10-9-10***, is not considered as a **"real" cycle.**

Chapter 6

EVALUATION METRICS

Effects on wavelength utilization
Cost analysis
Examples

6. EVALUATION METRICS

6.1. Effects on wavelength utilization

Since some wavelengths are assigned to monitoring channels within each node and link, the wavelength utilization of the network will decrease. The maximum number of occupied wavelengths in a node or link is the maximum number of wavelengths assigned for monitoring in this single node or link. It is equivalent to the maximum cover times of a single node/link in a cycle cover. The average number of occupied wavelengths is the average number of wavelengths assigned for monitoring in all nodes or links. It is equivalent to the average number of appearance of all nodes or links in a cycle cover.

To quantitatively analyze the relative degradation of wavelength utilization, we defined the maximum and average *wavelength overhead* brought to the network by monitoring cycles per node and per link respectively,

***H*max = maximum # of occupied λs / number of total available λs** (per node or link)
***H*ave = average # of occupied λs / number of total available λs** (per node or link)

Here ***H*ave** indicates the average degradation of wavelength utilization and ***H*max** is for the worst case of wavelength utilization degradation among all nodes/links. Nowadays along with the deployment of DWDM technology, the number of wavelengths in a single link tends to become larger and larger. For example, it is reported even in 2001 that 432 wavelengths could be multiplexed into a single fibre. Therefore we set the number of total available wavelengths in a node or link to be 432 for calculating ***H*ave** and ***H*max** in example networks given in Section …...

6.2. Cost analysis

The cardinality is the measurement of the size of a countable set. If a countable set is finite, its cardinality is the number of its elements. A countable but infinite set is said to have the cardinality ℵ0. For a finite undirected graph, each cycle cover is a finite set of cycles. The *cardinality of a cycle cover* is defined as the number of cycles in the cover. In monitoring-cycle based fault detection, since a transceiver is assigned to each cycle and a dedicated supervisory channel is established along the cycle for monitoring (one wavelength is used), the cardinality (cycle number) of a cycle cover represents the total required transceivers as well as the wavelengths for monitoring. Thus it is the measurement of the costs for the fault detection based on monitoring cycles.

6.3. Examples

We tested and compared the proposed fault detection mechanism using the described two heuristic algorithms respectively upon four example networks: NSFNET, ARPA2, SmallNet, and Bellcore, as shown in Fig. 6. The performance of the two cycle finding algorithms are compared in terms of the metrics described in Section IV. The comparison results are listed in Table 1. The results in Table 1 have shown that the numbers of occupied wavelengths per link or node in all examples are small: in [maximum 5 λs / node and 3 λs/ link] using HDFS, and 3 λs/node and 2 λs/link using SPEM. With the assumption of available wavelengths (432 wavelengths available per link/node), the maximum overhead (in worst case) is only 1.16% per node and 0.7% per link using HDFS, 0.7% per node and 0.5% per link using SPEM. Such overhead doesn’t impact the network utilization much, if it is not negligible.

The required number of transceivers for monitoring is determined by the cycle cover cardinality of each network example. We define the ratio between the number of required transceivers and the number of links in a network, to measure the relative costs for fault detection

***R*link = # of required transceivers for monitoring / # of links in the network**

The ratios are pretty small for all network examples.

For example, the worst case is the SmallNet that *Rlink* is 36.4% and 40.9% for using HDFS and SPEM respectively. Comparing to the conventional fault detection schemes, e.g. one-monitor-per-link case, the proposed mechanism cut more than half of the costs of the transceivers. Therefore the proposed fault detection mechanism is cost efficient for meshed AONs.

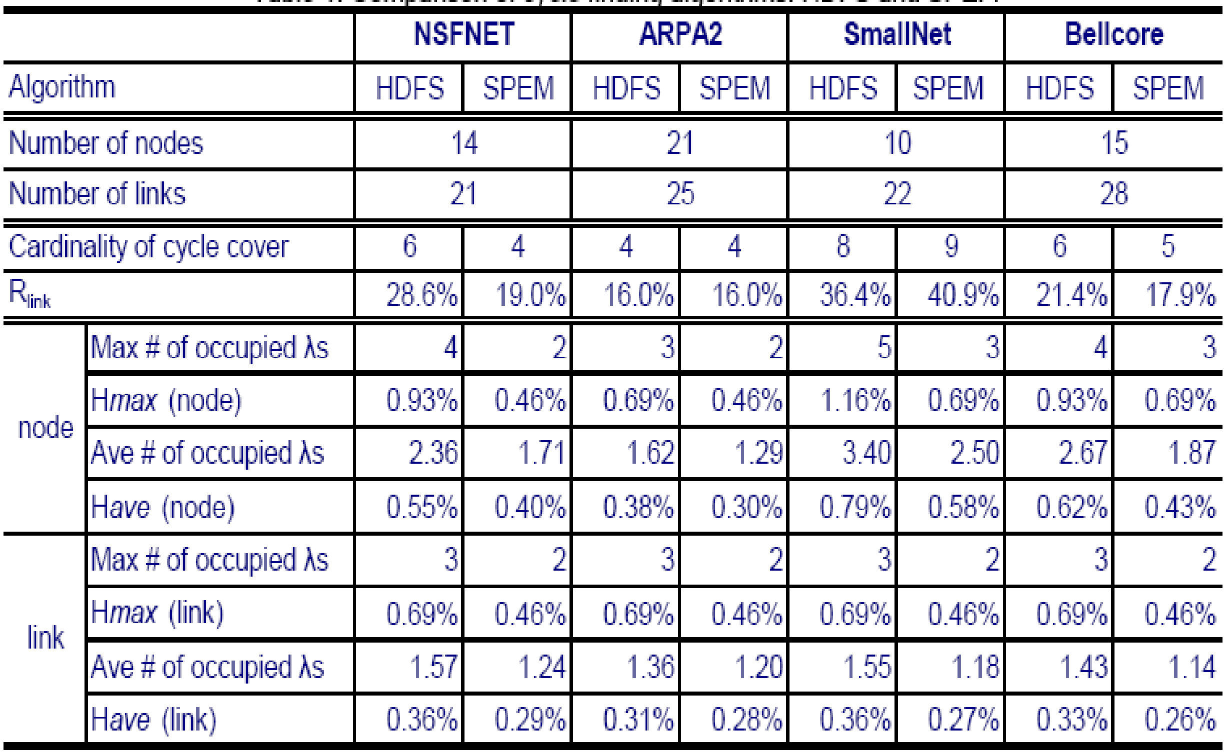
Chapter 7

CONCLUSION

7. CONCLUSION

This paper introduced the concept of monitoring cycles and proposed a fault detection and path performance monitoring mechanism based on decomposing AONs into monitoring cycles. The heuristic depth first searching (HDFS) and shortest path Eulerian matching (SPEM) algorithms are developed for finding monitoring cycles in AONs. The two algorithms are compared with respect to the maximum and average number of wavelengths occupied by monitoring in nodes and links. The results for the 4 network examples show that the wavelength overhead is pretty low with this mechanism. Thus the proposed mechanism based on monitoring cycles is a promising fault detection method for AONs. It is also applicable to path transmission performance monitoring. The results also suggest that the SPEM algorithm is better than the HDFS algorithm in terms of the wavelength overhead.

Table 1. Comparison of cycle finding algorithms: HDFS and SPEM



		NSFNET		**ARPA2**		**SmallNet**		**Bellcore**	
Algorithm		HDFS	SPEM	HDFS	SPEM	HDFS	SPEM	HDFS	SPEM
Number of nodes		14		21		10		15	
Number of links		21		25		22		28	
Cardinality of cycle cover		6	4	4	4	8	9	6	5
R_{link}		28.6%	19.0%	16.0%	16.0%	36.4%	40.9%	21.4%	17.9%
node	Max # of occupied λs	4	2	3	2	5	3	4	3
	Hmax (node)	0.93%	0.46%	0.69%	0.46%	1.16%	0.69%	0.93%	0.69%
	Ave # of occupied λs	2.36	1.71	1.62	1.29	3.40	2.50	2.67	1.87
	Have (node)	0.55%	0.40%	0.38%	0.30%	0.79%	0.58%	0.62%	0.43%
link	Max # of occupied λs	3	2	3	2	3	2	3	2
	Hmax (link)	0.69%	0.46%	0.69%	0.46%	0.69%	0.46%	0.69%	0.46%
	Ave # of occupied λs	1.57	1.24	1.36	1.20	1.55	1.18	1.43	1.14
	Have (link)	0.36%	0.29%	0.31%	0.28%	0.36%	0.27%	0.33%	0.26%

REFERENCES

[1] Y. Kobayashi, Y. Tada, S. Matsuoka, N. Hirayama, and K. Hagimoto, “Supervisory systems for all-optical network transmission systems,” *IEEE Globecom’96*, pp.933-937, 1996

[2] M. Médard, D. Marquis, and S. R. Chinn, “Attack detection methods for all-optical networks,” *Network and Distributed System Security Symposium*, 1998

[3] Y. Hamazumi, M. Koga, K. Kawai, H. Ichino, K. Sato, “Optical path fault management in layered networks,” *IEEE Globecom’98*, vol. 4, pp. 2309-2314, 8-12 Nov. 1998

[4] www.en.wikipedia.org