Detection of multiple failures in wavelength division multiplexed optical network using graph based light path restoration method

Md.Saifur Rahman*, Nargis Parvin, Md.Tofael Ahmed***, Md. Selim Reza****, Halida Homyara*****, Farhana Enam******

*, *** (Department of Information and Communication Technology, Comilla University, Comilla,Bangladesh)
** (Department of Electrical and Electronic Engineering, University of Information Tecnology and Sciences, Bangladesh)
****, *****, ****** (Department of Information and Communication Engineering, Rajshahi University, Bangladesh)

ABSTRACT

In an Optical WDM network, link/path failure may occur in a very normal way which may cause a great amount of data loss if the link is not restored to reroute the affected traffic accurately and quickly using the redundancy the network. Wavelength Division Multiplexed (WDM) networks, employing wavelength routing have emerged as the dominant technology to satisfy this growing demand for bandwidth. As the amount of traffic carried has increased, any single failure can be catastrophic. Survivability becomes indispensable in such networks. Therefore, it is imperative to design networks that can quickly and efficiently recover from failures. Most research to date in survivable optical network design and operation focuses on single link failures, however, the occurrence of multiple-link failures are not uncommon in a network topology. Multi-link failure scenarios can arise out of two common situations. First, an arbitrary link may fail in the network, and before that link can be repaired, another link fails, thus creating a multi-link failure sequence. Secondly, it might happen in practice that two distinct physical links may be routed via the same common duct or physical channel. In our proposed method predefined cycles are used to survive a failure link. When a link fails at first its adjacent shorter cycle is used as its backup path. This can decrease the time complexity. Then these adjacent cycles are updated by using this restoration method can survive multiple link failures. This method theoretically ensures better performances than existing restoration method.

Keywords – Network survivability, Network topology, Optical networks, Wavelength Division Multiplexing (WDM).

1. INTRODUCTION

Failure in a system is a very common scenario to face. Since networking system is very much sophisticated, the failure is some what very natural and high in this case. Failure may occur between two nodes in a transmitted path, again total path can be failed by any situation. Thus a network failure in such system may seriously impair service continuity to a countless number of users. So, network survivability is essential. Networks capable of protecting itself against the failure are called survivable networks [1]. Different restoration schemes have been proposed to ensure survivability. But an efficient restoration scheme is a demanding request in this field.

Our approach is to implement efficient algorithm for survivability in the case of two or more link failure occurrence. We want to eliminate the problems in existing restoration scheme and improve its performance. To do that we propose a method that can be able- To reduce the time complexity in existing restoration scheme when it search for a new backup path. To recover a backup path in case of multiple link failure. To reduce the loss of data efficiently when network failure occurs. The rest of the article is organized as follows. Section 2 provides some description of WDM optical Network and failures in WDM optical Network have also been discussed. Section 3 provides Existing Restoration Scheme and Their Drawbacks. Section 4 provides proposed rearrangement based light path restoration scheme with multiple link failure in WDM network. Section 5 provides Description of Experimental Result of Our Proposed Method and Comparison to Other Method and Section 6 concludes the paper.

2. Optical Network, WDM Optical Network and Failure Occurrence

In this section, we provide a brief overview of optical networks, WDM Optical Networks and their different generations.

First generation optical networks employ fiber only as a transmission medium. These networks essentially replace copper cables with optical fibers [2]. The key feature of first generation optical networks is that all processing is carried out in the electronic domain. The electronics at a node must handle all data intended for that node as well as all data passing through the node and destined to other nodes in the network. Second generation optical networks use WDM technology to split the huge bandwidth provided by a fiber into multiple wavelength channels, that can be used to support multiple transmissions simultaneously. Also, some of the switching and routing functions that are performed by the electronics in first generation optical network can be carried out in the optical domain in second generation optical networks. Second generation optical networks are over different types of services to the higher network layers. The most commonly used service is the lightpath service. A lightpath [3] is a dedicated connection on a wavelength between two nodes in the network, such that no electronic conversion takes place on the path between these two nodes.

Second generation optical networks employ WDM technology, they are also known as WDM networks. A typical WDM link is shown in Fig. 1

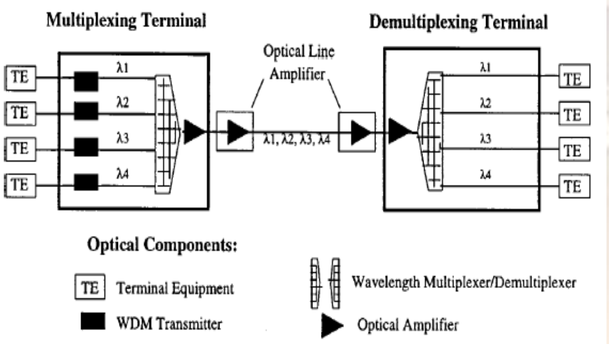


Fig. 1 A typical WDM network

It consists of a set of transmitters, optical amplifiers and receivers. The transmitters are lasers, each supporting one wavelength. The outgoing signals from different transmitters are multiplexed together using a multiplexer. The power amplifier immediately after the multiplexer amplifies the combined signal. The signal after traveling some distance on fiber may need amplification again due to attenuation; this task is carried out by an amplifier. Finally, at the destination, the combined signal is amplified again and de-multiplexed. Due to demultiplexing, the signal is split into different wavelengths which are converted to the electronic domain using photodetectors, where each photo detector is tuned to a specific wavelength.

In WDM optical networks the basic following types of failures can be occurred by [4].Link failures: Link failure usually occurs because of cable cuts. When a link fails, all its constituents fibers will fail. Node failures: Nodes failures occur because of equipment failure at network nodes. This type of failure may be the cause of all link failures which are connected to this node. Fiber failures: A fiber may fail due to the failure of its end components (multiplexers/de-multiplexers) in WXC (Wavelength Cross Connect).Channel failures: This also possible in WDM networks. A channel failure is usually caused by the failure of transmitting and/or receiving equipment operating on that channel (wavelength).

3. Existing Restoration Scheme

In survivable WDM networks, the lightpath that carries traffic during normal operations is known as the primary lightpath. When a primary Lightpath fails, the traffic is rerouted over a new lightpath known as the backup lightpath. In protection schemes, both primary and backup lightpath are computed before a failure occurs. In restoration schemes, a backup lightpath is discovered dynamically for a primary lightpath after a failure occurs. There similar restoration methods are may be two categories one is reactive and another is proactive. The proactive or reactive schemes can be either link based or path based [5].

The proactive or reactive schemes can be either link based or path based [1]. In *link based methods* select an alternate path between the end nodes of the failed link. This alternate path along with the intact part of the primary path is used for the recovery. The method is illustrated in Fig. 2. the figure shows a primary lightpath, A-B-D-G-H, and a backup lightpath, A-B-C-E-D-G-H is used when link B-D fails. It can be observed that A-B-C-E-D-G-H is routed around link B-D while retaining the working segment of A-B-D-G-H.

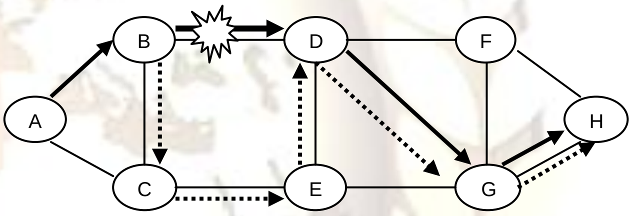


Fig. 2 link-based backup path reservations

In the case of path based methods, a backup path is computed between the end nodes of the failed primary lightpath. The backup path can use any wavelength independent of the one used by the corresponding primary lightpath [1]. The path-based restoration method is illustrated in Figure 3. Figure shows a primary lightpath, A-B-D-G-H, and its backup lightpath, A-C-E-G-H, on a given wavelength. Note that A-C-E-G-H is established between the end nodes of A-B-D-G-H, and the working segment of A-B-D-G-H is not utilized by A-C-E-G-H.

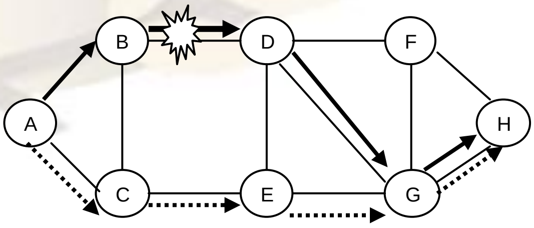


Fig. 3 path-based backup path reservations

In the multi-backup method, the FNM is sent directly to the source from the first node next to the failure if it has a link disjoint backup path with the source and if it has no link disjoint backup path with the source then it sends this FNM to the node along the primary path which has a link disjoint backup path with the source [6]. In figure 4.8, multi-backup method is shown. When link B-C fails in the primary path A-B-C-G-K. Then C sends the Setup Message to the source node A using the path C-F-E-H-A to configure the cross connects along the path. Then the source node resends the lost traffic through the path A-H-E-F-C and rest of active link C-G-K in primary path is used. Thus hops needed to restore the path are 4.

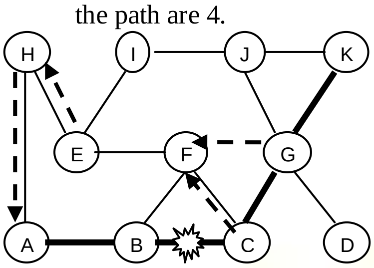


Fig.4. (a) multi-backup method with failure notification,

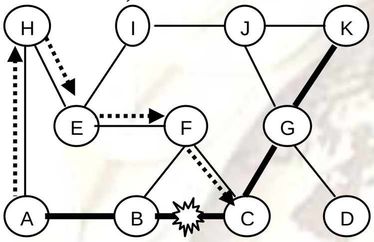


Fig.4. (b) retransmission of data using selected backup path.

In [7] a new scheme is proposed, according to this scheme backup lightpath is computed in advance like active method but unlike the active method it uses some alive links in the primary path for making the traffic travels less path thus making the restoration time less i.e. link disjoint is not required for the backup lightpath for the sake of restoration time. According to this method backup path between two end nodes vi and vj is calculated dynamically using the routing table of the adjacent nodes of the first node next to the failure where the backup path may use some of the links of the primary path if they are alive after the failure. Like in multi-backup active method we are referring the failure notification message as FNM and Setup Message to configure the intermediate cross-connects as SM. According to this method, under single link failure assumption, a primary light path is computed [7]. A backup path from each node along the primary path is also computed. Before failure the primary path is determined between a source and destination node pair by using Dijkstra's algorithm that includes the path with minimum number of hops between the source and destination nodes. If a link fails, the first node next to the failure detects the failure and it then sends a Setup Message to the source using the shortest path to the source. During the calculation of the backup path, shortest path from each node to the source node is considered as the backup path and is computed according to the following algorithm [8].

3.1 The Algorithm

Let the next node after the failure to be **f** and **f** has a backup path **b** to the source.

1. Set **p** = infinite
2. **for** each node adjacent to **f**
3. Calculate the shortest path to the source by using Dijkstra's Shortest Path algorithm and call it **temp_p**

If **temp_p** is shorter than **p then** set **p** = **temp_p**

6. **If** cost of **p** is < cost of **b then**
7. Send the Setup Message to the source using **p** and send the lost data from the **source**.

3.2 Example of the Algorithm

Existing rearrangement based lightpath restoration scheme for link failure restoration is described for the following network shown in figure 4. (a). Here in figure 5. (a), link failure occurs between node B and C along the primary path A-B-C-G-K where node

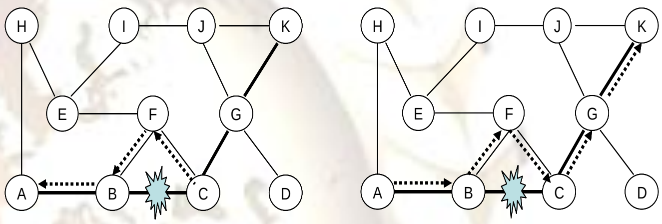


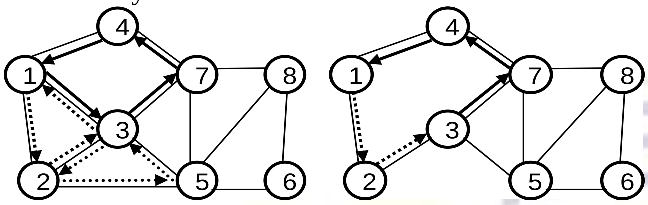
Fig. 5 rearrangement based light path restoration scheme (a) sending FNM and (b) restoring path

A is the source node and node K is the destination. After the failure, node C detects it and sends the SM using the path C-F-B-A though it has the path B-A which is not link disjoint to the primary path. Then A resends the lost traffic through A-B-F-C path (fig. 5 (b)) and uses the path C-G-K as well. Here, hops needed to restore is 3 which is reduced in this network and will be reduced in a great amount when number of nodes will be much higher than this network.

4. Proposed rearrangement based light path restoration scheme

In proposed method to transmit traffic from source to destination Dijkstra's algorithm is used to establish primary shortest path between sources and destination. After selecting the primary path source transmits data to destination. Here every link weight is equal, or unequal and shortest path is the path with shortest path length. To survive single or multiple links failure all the independent cycles are determined. Every node or host stores its all adjacent cycles. Every link can be attached with one or two cycles. To restore a backup path of failure link the shortest adjacent cycles of this failure link is used. After restore a failure link all the nodes of this link adjacent cycles update their cycles. Descriptions of the proposed method are illustrated with example in the following.

According to our proposed method fig. 6 (a) consists of the independent cycles 1-2-3-1, 2-5-3-2, 1-3-7-4-1, 3-5-7-3, 7-8-5-7 and 5-8-6-5. Here node 1 contains the cycles 1-2-3-1 and 1-3-7-4-1 and all the link backup paths of these cycles, similarly other nodes contain their adjacent cycles and their link backup paths. Here 1-2-3-1 cycle can provide the backup path for link 1-2 by 1-3-2, 2-3 by 2-1-3 and 1-3 by 1-2-3.



(a) (b)

Fig. 6 (a) independent cycles (b) update of cycles.

Similarly other cycles can provide backup paths for their adjacent links. In this figure 1-2 is an external link so it is adjacent to one independent cycle and has one backup path, 2-3-1 on the other hand 1-3 is an internal link, so it is adjacent with two independent cycles and has two backup paths, one is 1-2-3 and another is 3-7-4-1. The shortest cycle is used to recover the backup path in case of internal link. Fig. 6(b) shows that when external link 2-5 fails then its adjacent cycle is destroyed. When internal link 1-3 fails its two adjacent cycles 1-2-3-1 and 1-3-7-4-1 updated into 1-2-3-7-4-1.

In fig. 7 if node-2 is the source and node-9 is the destination then using Dijkstra's Algorithm 2-3-5-6-9 is selected as primary path. Now if the link 2-5 goes down then at first node-5 detects the failure and link 2-5 is protected by its adjacent cycle 2-5-3-2. Then node-5 sends the FNM to the source along 5-3-2. When the FNM is reached to the source node-2 then the source node retransmits the affected traffics along the restored path 2-3-5-6-9. Though the link 2-5 is an external link, its adjacent cycle 2-5-3-2 is destroyed.

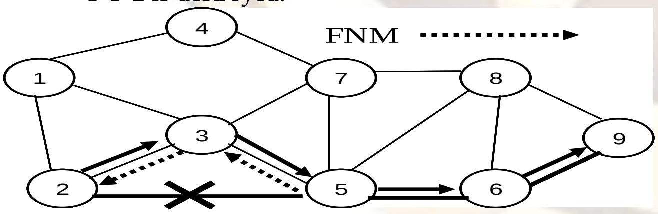


Fig. 7 single link failure protection

In fig. 8 if node-3 is the source and node-11 is the destination then the primary path 3-5-6-8-9-11 is selected. Now if the links 5-8, 5-7 and 8-9 fail at the same time or one after another then at first the node-8 detects the 5-8 failure then it tries to send the FNM to the source using the failure adjacent cycle 5-8-7-5(active path 8-7-5 is used as the backup path for 5-8). But the failure of link 7-5 blocks the FNM. Then node-8 uses it's another failure backup cycle 5-8-6-5 (the active portion of the backup cycle as 8-6-5 is used as the backup path for link 5-8) and sends the FNM to the source along 8-6-5-3. Here 5-8 is an internal link its adjacent two cycles 5-6-8-5 and 5-8-7-5 are converted into a new cycle 5-6-8-7-5, again for the failure 5-7, its adjacent cycles are converted into a new cycle. Now source node-3 retransmit the affected traffic along the restored path 3-5-6-8-9-11 but the failure of link 8-9 (internal link) again blocks the traffics. Then its adjacent shorter cycle is similarly used to restore the new backup path against failure link 8-9.

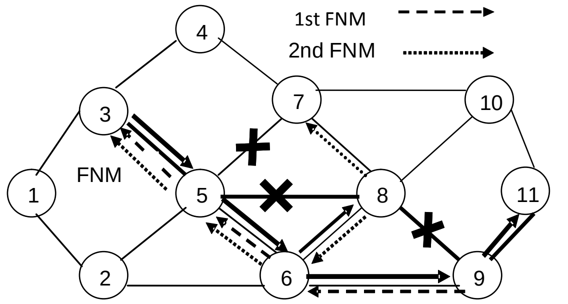


Fig.8 multiple links failure protection

Here cycle 8-9-6-8 is shorter than 8-9-11-10-8, so first cycle is used to restore the backup path. Here node-9 detects the failure and sends the FNM to the source along 9-6-5-3 then the source node-3 retransmits the affected traffics along the new backup path 3-5-6-9-11. Note that when external link on primary path and any other link on its adjacent independent cycle fail then second failure link uses its adjacent active cycle to restore backup path.

4.1 Algorithm for Proposed Method

The algorithm for finding shortest path between two nodes source to destination and the algorithm for finding shortest path are included in this section

In the following algorithm, u: = Extract_Min (Q) searches for the vertex *u* in the vertex set *Q* that has the least *d*[*u*] value. That vertex is removed from the set *Q* and returned to the user [12].

```
1 function Dijkstra (G, w, s)
2   for each vertex v in V [G]
3       d[v]:= infinity
4       previous[v]:= undefined
5   d[s]:= 0
6   S: = empty set
7   Q: = set of all vertices
8   while Q is not an empty set
9       u: = Extract_Min (Q)
10      S: = S union {u}
11      for each edge (u, v) outgoing from u
12          if d[v] > d[u] + w (u, v)
13              d[v]:= d[u] + w (u, v)
14              previous[v]:= u
```

If we are only interested in a shortest path between vertices *s* and *t*, we can terminate the search at line 9 if *u* = *t*.

Now we can read the shortest path from *s* to *t* by iteration:

1 S: = empty sequence

2 u: = t
3 **while** defined u
4 insert u to the beginning of S
5 u: = previous[u]
Now sequence *S* is the list of vertices on the shortest path from *s* to *t*.

At first we have to determine all the independent cycle in the network graph G (i, j). In our approach every node contain all of its adjacent independent cycle. Only one or two cycle can be adjacent with a link. When a link failure occurs in the primary path then we determine the independent cycle (sorter cycle in case of two adjacent independent cycles) adjacent to that failure link. After a link failure occurs all the nodes of the independent cycle adjacent to the failure link update there containing cycle which is adjacent to that failure link. (If the failure link has two adjacent independent cycle then these two cycles create a new cycle otherwise the cycle breaks) Then all other links of that cycle except failure link is used as the backup path of that failure link. The first node next to the failure link just sends an FNM (Failure Notification Message) to the source then the source resends the traffic along with this new path. When two or more links fail simultaneously (in case of multiple link failure) always active cycle works for surviving iteratively.

All the independent cycles in the network are determined and every node has to store its adjacent cycles. By using this cycle backup path nodes adjacent links can be protected.

1. Compute the shortest path, p*(l)* between node pair *(s, d)* using Dijkstra's algorithm for simplicity assume that the primary path consists of a shortest sequence of nodes.
2. If a link failure is detected by p*(i)* node on the primary path
Then set check =0, *Max*= number of adjacent cycles to p*(i) node*
2.1 For *(j=1 to Max)*
If p*(i)*= backuplink*(s)* *and* p*(i)*= backuplink*(e)*
Then replace the failure link by active path between failure link adjacent nodes in the used cycle.
Set *check=1* and break
Set *length*=active backup cycle length
End loop
2.2 if *(check=1)*
Set a new backup path p*(m)* where m=*l+length*
Else No path avail able
End If
3. Then the node by which the failure is detected, send the setup message using the from p*(i+m)* to p*(1)*
4. Then affected traffics are retransmitted along the selected new backup path.
5. Then all the nodes on adjacent cycle to the failure link just update their affected cycle.
6. If another failure is detected again start from step 2 to establish a new backup path.

We have implemented our proposed algorithm by C language. We have tested our implemented algorithm or program for the various graphs. In our implementation at first we determine all the independent cycles of the network using the link adjacent matrix and the cost matrix. In adjacent matrix e_{ij} =1 for i $\neq$ j and e_{ij} belongs to *E*. Where *E* is a set of edges otherwise e_{ij} =0. In cost matrix c_{ij} =weight for i $\neq$ j and c_{ij} belongs to *E*. Where *E* is a set of edges otherwise c_{ij} =0. Then all the cycles are determined from the matrix are as above figure.

$$\begin{pmatrix} e_{11} & e_{12} & e_{13} & \cdots\cdots & e_{1n} \\ e_{21} & e_{22} & e_{23} & \cdots\cdots & e_{2n} \\ e_{n1} & & & & e_{nn} \end{pmatrix}$$

$$\begin{pmatrix} c_{11} & c_{12} & c_{13} \cdots\cdots c_{1n} \\ c_{21} & c_{22} & c_{23} \cdots\cdots c_{2n} \\ & & \\ c_{n1} & c_{n2} & c_{n3} \cdots\cdots c_{nn} \end{pmatrix}$$

Fig.9 adjacent or path matrix and cost matrix

Where first chart represents the cycles before failure and its first column represents the number of cycle's node. Second chart represents the cycles after two or more failures and its first column represents the number of cycle's node. Then by using first chart's cycles we can establish the backup path for the single failure link and rearrange them as second chart, we can establish the backup path for the multiple failure links. In the following figure independent cycles before failure and initial primary path, Independent cycles after one link failure and backup path, and Independent cycles after two links failure and corresponding backup path are shown from implementing output.

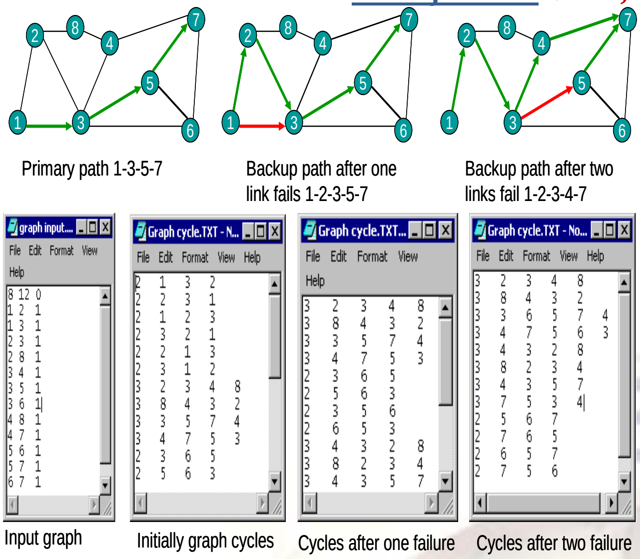


Fig.10 Input graph, Independent cycles before failure, Independent cycles after one link failure, Independent
cycles after two links failure

5. Experimental Result, and Comparison to other Methods

To analysis the experimental results we consider every link weight is equal or may unequal. In time complexity depends on searching strategy. The algorithm which can reduce nodes to recover path or establish path for transmitting data is better. Because every node of transmission line need some time to process data. Similarly the algorithm which can reduce this time complexity can work fast.

5.1 Description of Experimental Result with single Link Failure

To analysis the result, we have considered node 1 as source and node 21 as destination in figure: 6.1, 6.2, 6.3 and 6.4. Then using Dijkstra's algorithm the primary path is 1-6-11-16-21 selected. If the link 11-16 fails, then to survive the single failure link by existing methods and our proposed method are given in the following.

5.1.1 Path based method

As the link 11-16 fails, the node 16 sends a acknowledgement to the destination node 21 and then node 21 sends a set up message to the source node 1 via the path 21-19-15-10-5-2-1 which is link disjoint from the primary path. Then node 1 resends the lost traffic through the path 1-2-5-10-15-19-21. This method needs 6 nodes to recover 11-16 links after fails.

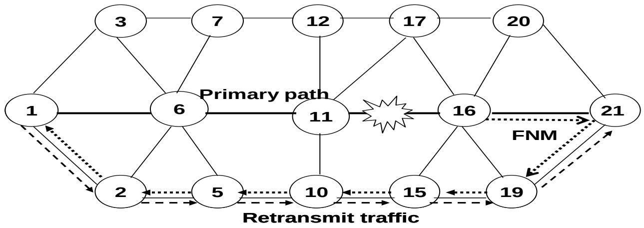


Fig. 11 single link failure protection with path based method.

5.1.2 Multi backup method

When the link failure between node 11 and node 16 occurs along the primary path 1-6-11-16-21 the first node next to the failure, 16, as it is a directly connected to the source with a link disjoint backup path, sends the Setup Message to the source node 1 using the path 16-15-10-5-2-1 to configure the cross connects along the path. Then node 1 resends the lost traffic through the path 1-2-5-10-15-16 and rest of the active link 16-21 in the primary path is used. This method needs 5 nodes to recover 11-16 links after fails.

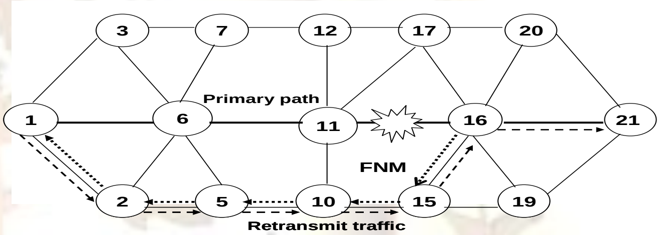


Fig. 12 single link failure protection with multi backup method.

5.1.3 Rearrangement based method

The first node next to the failure detects the failure and it then sends a Setup Message to the source using the shortest path to the source *et al* [11]. During the calculation of the backup path, shortest path from each node to the source node is considered as the backup path. In the figure after link 11-16 fails, node 16 sends the setup message to source node 16 using shortest 16-17-12-7-3-1 then node 1 resends the lost traffic using 1-3-7-12-17-16-21. . This method needs 4 nodes to recover 11-16 links after fails.

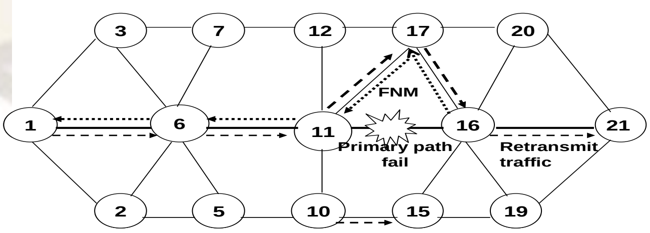


Fig. 13 single link failure protection with rearrangement based method.

5.1.4 Our Proposed Method

According to our proposed method 11-16 link has two adjacent independent cycles 11-16-17-11 and 11-16-15-10-11 here first cycle is sorter than second so for 11-16 link failure 11-16-17-11 is used as protection cycle of 11-16 link. Then 16 send an FNM to A using 16-17-11-6-1 and 1 resends the traffic to 21 using 1-6-11-17-16-21. Then 11, 10, 15, 16 and 17 nodes update their cycle which contains 11-16 links. As 11 update its 11-16-17-11 or 11-16-15-10-10 by 11-10-15-16-17 Cycle. . This method needs 4 nodes to recover 11-16 links after fails.

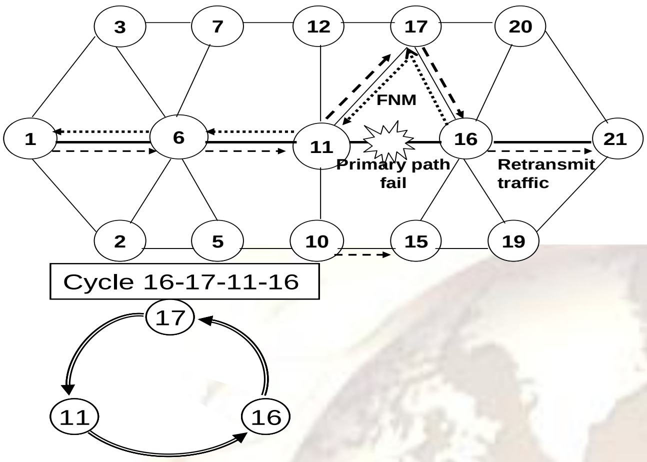
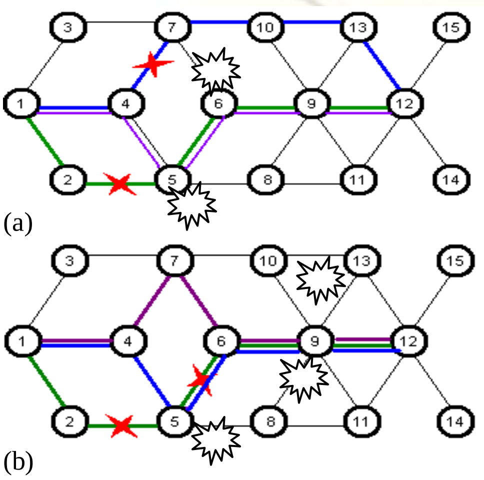


Fig. 14 restoration of backup path for a Single failure link by independent cycle in our proposed method.

5.2 Description of Experimental Result with Multiple Links Failure

To recover multiple failures in the following network, we have considered source 1 and destination 12 then the primary path is 1-2-5-6-9-12 is selected. When first failure occurs at link 2-5 then according to *"Dual-Link Failure Resiliency through Backup Link Mutual Exclusion*" method backup path will be 1-4-7-10-13-12 (in case of *"Link Protection Failure Independent Protection (LP-FIP)"*). In this case backup path is mutually exclusive to the primary path by Chandak *et al* [2]. Now if 4-7 link fails there is no any mutually exclusive path then the network is block for the affected traffics. Because there is no any mutual exclusive backup path for 4-7 link



(a)

(b)

Fig. 15 multiple failure restoration using (a) existing dual-link failure resiliency through backup link mutual exclusion method (b) proposed method

In our proposed method, the primary path is 1-2-5-6-9-12.When the link 2-5 fails, then the back up path 1-4-5-6-9-12 is established by using the cycle 5-4-1-2-5. When second failure occurs at the link 5-6, then the back up path 1-4-7-6-9-12 is established by using the cycle 6-7-4-5-6.

5.2.1 Comparison Result with Other Existing Method

To compare the performance of our proposed, we have compared our proposed method with existing path based method, multi backup method and rearrangement based method in case of single link failure and with *LP-FIP* method in case of multiple links failure.

5.2.2 Comparison Results with existing methods in case of single link failure

In figure: 11, 12, 13 and 14 when link 11-16 fails then the number of nodes required to restore the backup path, selecting path to notify the failure and the new restored backup path are as following Table I. from this table we can show that to restore a backup path to notify the failure to source, path based method needs 8 nodes, Multi Backup Method needs 6 nodes, existing rearrangement based method needs 5 nodes and Proposed Method needs only 4 nodes. So our proposed method reduced the number of nodes to transmit failure notification to the source node from first node next to the failure link.

Table I. Comparison result for failure link recovery from above four figures.

Restoration Scheme	No of nodes to Recover path	Nodes Used to Acknowledge	Nodes Used to Resend Data
Path Based Method	8	16-21-19-15-10-5-2-1	1-2-5-10-15-19-21
Multi Backup Method	6	16-15-10-5-2-1	1-2-5-10-15-16-21
Rearrange ment Based	5	16-17-11-6-1	1-6-11-17-16-21
Proposed Method	4	16-17-11	1-6-11-17-16-21

The experimental results in the following Table II show the required average numbers of nodes to recover a single link failure over the total number of network nodes and links. In fig. 13 we have

ISSN: 2248-9622 www.ijera.com Vol. 3, Issue 1, January -February 2013, pp.1398-1406

compared our proposed method with path based method, multibackup method and existing rearrangement based method in case of single link failure recovery.

Table II Average length of backup paths in various networks (In case single link failure)

Network No	1	2	3	4	5
No of hopes	12	21	27	45	65
No of Links	16	25	50	64	83
Required No of nodes/hopes for restoring backup path					
By path based method	6	7	9	13	17
By multi backup method	5	6	7	11	13
By rearrangement based	3	4	6	8	11
By proposed method	3	4	6	8	11

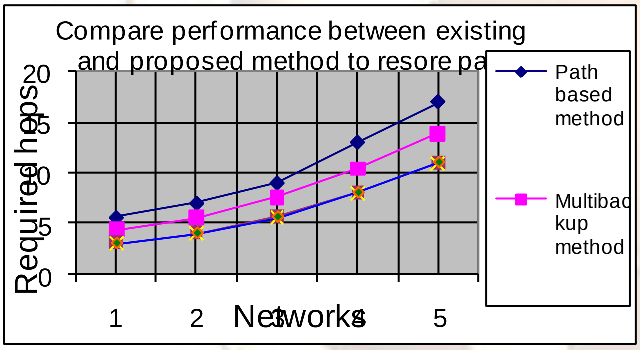


Fig. 16 Average numbers of nodes required to restore backup paths by existing method and proposed method

Table I. shows that the multi backup method can reduce the number of hopes to recover a failure link than path based method. On the other hand the rearrangement based method can show better performance than multi backup method.

5.2.3 Comparison Results with Existing Restoration Scheme in Case of Multiple Failures

The experimental results in the following table show the required number of nodes to recover multiple links failure over the total number of network nodes and network links.

In fig. 17 we have compared our proposed method with LP-FIP in Dual-link failure resiliency along backup link mutual exclusion in case of single link failure and in case of multi links failure.

Table III Average length of backup paths in various networks (With multiple links failure)

Network No		1	2	3	4	5
No of hopes		12	21	27	45	65
No of Links		16	25	50	64	83
Required No of nodes/hopes for restoring backup path						
LP-FIP method	Single fail	3	5	6	9	13
	Multi fails	8	11	15	23	31
Proposed method	Single fail	3	5	6	9	13
	Multi fails	6	9	13	17	23

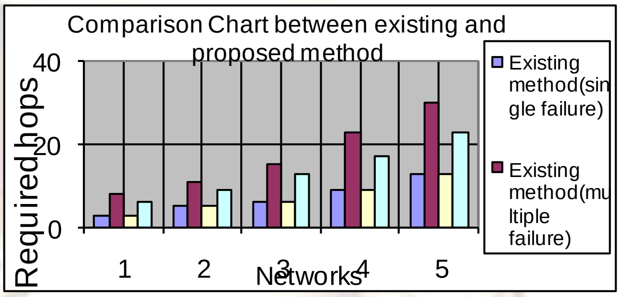


Fig. 17 average numbers of nodes required to restore backup paths by LP-FIP and proposed method

5.3 Complexity Analysis

We can express the running time of Dijkstra's algorithm on a graph with m edges and n vertices as a function of m and n using the Big O notation. Each iteration, need n*(n+1)/2 comparisons so complexity in this algorithm to find shortest path is $O(n^2)$. In our proposed method to select primary path Dijkstra's algorithm is used. So complexity to select primary path is $O(n^2)$. Now if n is the total the number nodes or hops and m is total the number of links or edges aaccording to our proposed method the network has m-(n-1) number of independent cycles in the network where every independent cycle has maximum n nodes. By using these cycles we can determine m (n<m) backup path against every link. So we total comparison for finding backup path m where n=m or n<m. So the time complexity to search a backup cycle is *O (n).*

Table IV. Comparison of Complexity with various existing restoration schemes

No	Restoration method to survive	Searching algorithm to search backup path	Complexity for primary path	Complexity to restore path
1	Path based	Used to search full path	O(n^2)	O(n^2)
2	Multi backup	Used to search full path	O(n^2)	O(n^2)
3	Rearrangement	Used to search full path	O(n^2)	O(n^2)
4	Proposed method	Search only backup cycle	O(n^2)	O(n)

From complexity analysis we show that to select primary path our proposed method uses Dijkstra's algorithm as other method has the complexity *O(n^2)*. But to restore the backup path existing methods use Dijkstra's algorithm so the complexity to restore backup path is *O(n^2)*. But to restore the backup path for the failure link proposed method use only predefined independent cycle adjacent to the failure link which reduces the time complexity. In our proposed method the complexity is *O*(n) to restore a backup path. So our proposed method significantly works fast to restore backup path.

6. Concluding Remarks

The current multi-backup method performs better than the path based method but rearrangement method shows a better performance against the multi-backup method. Which reduces the backup path length, and hence the restoration time is also reduced. But to restore the backup path for the failure link this method uses Dijkstra's algorithm that increases the time complexity of this process. In a large network multiple failures are not uncommon, but this method can not handle the multiple failures. In our proposed method, the required nodes to restore a failure link are same. But to restore the backup path for the failure link proposed method use only shortest predefined independent cycles, adjacent to the failure link which reduces the time complexity of this method. After restore a backup path independent cycles adjacent to the failure link are updated that make this method able to handle multiple failures. So this method shows better performance than existing rearrangement based method. In case of multiple failures our proposed method can solve the blocking problem in LP-FIP.
Our proposed method has some limitations, such as: When the failures of site equipments or node failures create multiple links failure in a network then this method fail to survive.

This method is faster than existing rearrangement based method to survive single link failure. But after survive the first failure, to update the adjacent cycles to the failure link takes some time that may slow to restore backup path of second, third or more other failure links.

To show better performance we have the following recommendation for our proposed method, to reduce the limitations of our proposed method the network components must be reliable, and have to ability to provide backup.

REFERENCES

[1] Sateesh Chandra Shekhar, "Survivable Multicasting in WDM Optical Networks*,"* By Bachelor of Engineering (Computer Engineering) University of Pune, August 2004.

[2] O. Crochat, J. Y. Boudec, and O. Gerstel,"*Protection interoperability for WDM optical networks*," IEEE/ACM Transactions on Networking, June 2000.

[3] Pallab Datta, *"Survivability approaches for multiple failures in WDM optical networks,"* Iowa State University Ames, Iowa 2005.

[3] C.Silva Ram Murthy and Mohan Gurusami,"*WDM Optical Networks Concepts, Design, and Algorithm,*" Pearsonal Education.

[4] Andrew Tanenbaum "*Computer Networks*"-fourth edition Vrije Universiteit, Amsterdam, The Netherlands 2007-2008.

[5] Son Hong Ngo,*" Dynamic Lightpath Protection in WDM optical Networks Using Ant-based Mobile Agents,"* Graduate School of Information Science JAIST, Japan Email: sonhong@jaist.ac.jp November 2005.

[6] M. M. A. Azim and Md. Mamun-ur-Rashid Khandker, "*An Active Lightpath Restoration Method Using Multi-Backup Paths in WDM Networks*", Proceedings of the International Symposium on Towards Peta-Bit Ultra-Networks, Ishikawa, Japan, September 8-9, 2003.

[7] Md.Mahabubur Rahman, Syed Md.Galib and Md.Hasem Uddin,"*a new Rearrangement based Lightpath Restoration Scheme in Optical Networks*", International Conference on Computer and Information Technology (ICCIT)'2005.

[8] Retrieved from http://en.giswiki.net/wiki/Dijkstra%27s_algorithm