Survivable Routing in WDM Networks - Logical Ring in Arbitrary Physical Topology

Arunabha Sen, Bin Hao, Bao Hong Shen and Guohui Lin

***Abstract*— In this paper we consider the problem of routing the lightpaths of a logical topology of a WDM network on an arbitrary physical topology, such that the logical topology remains connected even after the failure of a physical link. We focus our attention on the ring interconnection as the logical topology because it is widely used in many protection schemes. We first establish the necessary and sufficient condition for a ring logical topology to withstand failure of a single physical link. Next we show that the testing of this necessary and sufficient condition is an NP-complete problem. Finally, we give an algorithm for testing the necessary and sufficient condition and demonstrate the execution of the algorithm with the help of an example.**

I. Introduction

Survivabilty of high bandwidth optical networks has become important area of research in recent times due to its tremendous importance as a national and international infrastructure for moving large volumes of data from one part of the globe to another. A significant number of papers addressing various aspects of survivability have appeared in the literature [2, 4-10, 12]. In a recent paper [7], Modiano and Narula-Tam introduced the notion of *survivable routing*, where the objective is to route the lightpaths (corresponding to the links of a logical topology) in the physical topology in such a way, that failure of any single physical (optical fiber) link cannot disconnect the logical network. As defined in [7], a routing is known as *survivable* if the failure of any physical link leaves the (logical) network connected. A somewhat similar problem was studied in [2], [10] where the objective was to support IP networks over WDM networks.

We illustrate the concept of *survivable routing* with the help of an example given in [7]. Suppose that the logical topology is a ring (figure 1a) and the physical topology is the network shown in figure 1b. The logical topology may be embedded in the physical topology in many different ways. For the embedding, the nodes of the logical topology first have to be mapped onto the nodes of the physical topology and then the links of the logical topology have to be mapped onto the

A. Sen is with the Dept. of Computer Science and Engineering, Arizona State University, Tempe, AZ 85287, e-mail: asen@asu.edu.

Bin Hao is with the Dept. of Computer Science and Engineering, Arizona State University, e-mail: binhao@asu.edu.

Bao Hong Shen is with the Dept. of Computer Science and Engineering, Arizona State University, e-mail: bao@asu.edu.

Guohui Lin is with the Dept. of Computing Science, University of Alberta, Edmonton, Canada, T6G 2E8, e-mail: ghlin@cs.ualberta.ca.

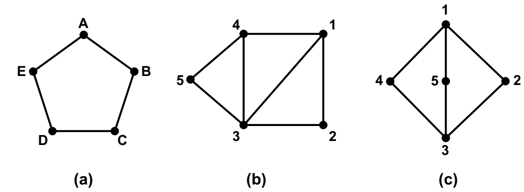

Fig. 1. Logical and physical topologies of WDM networks

paths in the physical topology. Consider the following embedding of the logical topology into the physical topology shown in figures 1a and 1b respectively. The nodes are mapped as $A \to 1, B \to 2, C \to 3, D \to 4, E \to 5$, and the edges are mapped as $(A-B) \to (1-2), (B-C) \to (2-3), (C-D) \to (3-4), (D-E) \to (4-5)$ and $(E-A) \to (5-4-1)$. If the lightpaths are established using this routing, it is clear that the failure of the fiber link between the nodes 4 and 5 will disconnect the logical topology as the node E will lose its connection to both the nodes A and D. However, this situation can be avoided by carrying out the edge mapping (routing of the lightpaths) in a slightly different way: $(A-B) \to (1-2), (B-C) \to (2-3), (C-D) \to (3-4), (D-E) \to (4-5)$ and $(E-A) \to (5-3-1)$. If this mapping (routing) is used, then failure of any single physical link cannot disconnect the logical topology (ring). Thus it is clear that the way the lightpaths are routed has a tremendous impact on the survivability of the logical network. In this paper we examine the issues related to the existence of a survivable routing of a logical ring in a physical network of arbitrary topology.

In the previous example, it was possible to find survivable routing just by changing the edge mapping and without changing the node mapping. However, there exist instances where an "incorrect" node mapping creates an environment where no survivable routes can be found. The following example illustrates the point. We consider mapping the same logical topology (figure 1a) onto the same physical topology (figure 1b), but this time around the nodes are mapped as follows: $A \to 5, B \to 2, C \to 4, D \to 3$, and $E \to 1$. It can be verified that no survivable routes between the nodes can be found in this case. These two examples show that survivable routes may be found if the nodes of the logical topology are mapped "correctly" onto the nodes of the physical topology and may not be found in case they are mapped "incorrectly". However, for certain physical topologies there may not be any "correct"

0-7803-7400-2/02/$17.00 © 2002 IEEE

node mapping and as such survivable routes for a logical ring cannot be found in such physical topologies. Survivable routes for the logical ring in figure 1a cannot be found if the physical topology is as shown in figure 1c.

It is clear from the previous example that survivable routes for a logical ring can be found for some physical topologies and cannot be found for some other topologies. In this paper we investigate the necessary and sufficient condition for the existence of survivable routes for a logical ring in any arbitrary physical topology.

II. Problem Formulation

The physical topology of the network is represented by an *undirected* graph $G_p = (V_p, E_p)$, where V_p is the set of nodes and E_p is the set of *physical* links. Similarly, the logical topology is represented by another undirected graph $G_l = (V_l, E_l)$, where V_l is the set of nodes and E_l is the set of *logical* links. We assume that $|V_p| = |V_l|$. The objective of the survivable routing problem is to find a way to route (map) the logical topology on the physical topology such that the logical topology remains connected inspite of the failure of any one single physical link.

In order to establish a logical link between the nodes s and t of the logical network, a lightpath needs to be established between the nodes $f(s)$ and $f(t)$ in the physical network, where $f(s)$ and $f(t)$, are the images (or mappings) of the nodes s and t in the physical network. Such a lightpath may use a set of physical links and some wavelengths on these links. Since the objective of this paper is to focus on the issues of a survivable design, as in [7], we assume that either a sufficient number of wavelengths or a sufficient number of wavelength converters are available, so that the issues related to wavelength continuity can be ignored.

We use the standard graph theoretical terminologies from [1].

III. Survivable Routing of Ring in Arbitrary Physical Topology

Let (f, F) denote a mapping between two graphs, where f is the node mapping, F is the edge-to-path mapping.

Theorem 1: Let $G_p = (V_p, E_p)$ and $G_l = (V_l, E_l)$ represent the physical and the logical topology of the network respectively. If G_l is a ring network, then there exists a survivable routing for G_l in G_p if and only if G_p contains a closed trail visiting each node at least once.

Proof: Suppose that the logical ring network is the cycle with n nodes represented as $v_0 e_1 v_1 e_2 v_2 \ldots e_{n-1} v_{n-1} e_n v_0$ such that for $1 \leq i \leq n-1$, the ends of e_i are v_{i-1} and v_i and the end of e_n is v_n and v_0. Let (f, F) denote a mapping of G_l in G_p, where $f(v_i)$ is the mapping of the nodes of G_l onto the nodes of G_p and $F(e_i)$ is the mapping of the edges of G_l onto the *paths* of G_p (paths may be of length one).

$\Longrightarrow$ Suppose that there is a survivable routing for G_l in G_p given by (f, F). Then for all $e_i \neq e_j$, $F(e_i) \cap F(e_j) = \emptyset$. This is true because, if there was an edge e of the physical graph in $F(e_i) \cap F(e_j)$ (i.e., $e \in F(e_i) \cap F(e_j), e_i \neq e_j$, then failure of e would disconnect the logical links e_i and e_j. Since G_l is a ring network, failure of the links e_i and e_j would disconnect G_l, contradicting the assumption that (f, F) is a survivable routing for G_l in G_p. Thus all $F(e_i)$'s are pair-wise edge-disjoint. It is not difficult to check that $F(e_1)F(e_2)\ldots F(e_n)$ forms a closed trail visiting each node of G_l at least once.

$\Longleftarrow$ Suppose that G_p has a closed trail, $w = u(0)u(2)...u(m-1)$, visiting each node at least once. Here, each $u(i)$ is a node of G_p. It may be noted that for some i and j, $u(i)$ may be equal to $u(j)$ even though $i \neq j$. Suppose that $P(u(i), u(j))$ denotes the path between the nodes $u(i)$ and $u(j)$ on w.

Now we construct a mapping (f, F).

begin
    for ($i := 0;\ i < m;\ i := i + 1$)
        $mark(u(i)) := 0$;
    $f(v_0) := u(0)$
    $mark(u(0)) := 1$;
    $index := 1$;
    $j := 1$;
    while $(j < n)$ *do* {n is the number of nodes in G_l}
        begin
            while ($mark(u(index)) = 1$) *do*
                $index := index + 1$;
            $f(v_j) := u(index)$
            $mark(u(v_j)) := 1$;
            $index := index + 1$;
            $j := j + 1$;
        end
    for ($i := 1;\ i < n;\ i := i + 1$)
        $F(e_i) := P(f(v_{i-1}), f(v_i))$;
    $F(e_n) := P(f(v_{n-1}), f(v_0))$;
end

We claim that (f, F) is a survivable routing. First, since w visits each node of G_p at least once, $f()$ is a mapping from V_l to V_p. Since w is a trail, the way the function $F()$ is constructed, $F(e_i) \cap F(e_j) = \emptyset$ for all i and j when $i \neq j$. Therefore, the failure of one link in G_p will disconnect at most one edge in G_l, leaving the logical network (ring) connected. Thus (f, F) is survivable.

As noted earlier an *Euler tour* is a tour which traverses each edge *exactly once*. Since such a tour traverses each edge exactly once, it must be traversing each node at least once. The necessary and sufficient condition for a graph to have an Euler tour is given by the following theorem [1].

Theorem 2: A nonempty connected graph is eulerian if and only if it has no nodes of odd degree.

As stated in theorem 1, it is possible to find a survivable routing for a logical ring network in a physical network of arbitrary topology, if and only if the physical network $G_p = (V_p, E_p)$ contains a closed trail visiting each node at least once. Suppose that $G'_p = (V'_p, E'_p)$ is a subgraph of G_p such that (i) $V_p = V'_p$ and $E'_p \subseteq E_p$ and G'_p is eulerian.

It is not difficult to verify that $G_p = (V_p, E_p)$ contains a closed trail visiting each node at least once, if and only if such a subgraph G'_p of G_p exists. From theorem 2 it is known that G'_p will be eulerian if and only if G'_p is connected and has no node with odd degree. It may be noted that the physical network G_p may or may not have such a subgraph G'_p. Thus a survivable routing for a logical ring network in a physical network, G_p, of arbitrary topology exists if and only if G_p contains such a subgraph G'_p.

Survivable Routing of Ring Problem (SRRP)
Instance: A connected undirected graph $G = (V, E)$.
Question: Does G have a subgraph $G' = (V', E')$, such that $V = V'$ and $E' \subseteq E$, such that G' is connected and has no node with an odd degree.

We prove that SRRP is NP-complete by restricting it to cubic graphs (it may be recalled that a graph is called *cubic* if all the nodes in the graph are of degree 3)

Survivable Routing of Ring Problem in Cubic Graph (SRRPC)
Instance: A connected undirected cubic graph $G = (V, E)$.
Question: Does G have a subgraph $G' = (V', E')$, such that $V = V'$ and $E' \subseteq E$, such that G' is connected and has no node with an odd degree.

Hamiltonian Cycle Problem in Planar, Cubic and Triply connected Graph (HCPPCT)
Instance: A connected undirected graph $G = (V, E)$, which is (i) planar, (ii) cubic and (iii) triply connected (i.e., deletion of any two nodes leaves the graph connected).
Question: Does G contain a Hamiltonian Cycle?

It has been proven in [3] that HCPPCT is NP-complete.

Hamiltonian Cycle Problem in Cubic Graph (HCPC)
Instance: A connected undirected cubic graph $G = (V, E)$.
Question: Does G contain a Hamiltonian Cycle?

HCPC is NP-complete because HCPPCT, a restricted version of HCPC is NP-complete.

Theorem 3: Survivable Routing of Ring Problem in Cubic Graph is NP-complete.
Proof: Clearly, SRRPC is in NP as it is fairly simple to check if G' is connected and has no nodes with odd degree.

We will give a transformation from the known NP-complete problem HCPC. We take the instance of SRRPC to be the same as the instance of HCPC. Suppose that this instance is $G = (V, E)$. We claim that G has a Hamiltonian cycle if and only if there is subgraph $G' = (V', E')$ in G that is connected and has no node with odd degree.

Suppose that $G = (V, E)$ contains a Hamiltonian cycle. Construct a subgraph $G' = (V', E')$ as follows: $V' = V$ and E' is the set of edges that make up the Hamiltonian cycle. Clearly, G' is connected and has no node with odd degree (all nodes of G' have degree 2).

Conversely, suppose that $G = (V, E)$ has a subgraph $G' = (V', E')$ such that $V' = V$, $E' \subseteq E$ and G' has no node of odd degree. Since G is cubic graph all the nodes of G are of degree 3. Since G' does not have any node with odd degree, all nodes of G' must be of degree 2. Since all nodes of G' are of even degree (2), G' has a Euler tour. Since all nodes of G' are of degree 2, this Euler tour is also a Hamiltonian cycle of G' and hence G. This proves the theorem.

Theorem 4: Survivable Routing of Ring Problem is NP-complete.
Proof: SRRP is NP-complete because SRRPC, a restricted version of SRRP is NP-complete.

IV. Algorithm for Survivable Routing

In this section, we describe an algorithm for finding survivable routes, if they exist, in the physical network G_p. To this end, we first prove a theorem.

Theorem 5: Suppose that $G_p = (V_p, E_p)$ is the physical topology. Suppose that S, ($S \subseteq E_p$) is an edge-minimal subset of E_p such that $G'_p = (V_p, E_p - S)$ is Eulerian. Let G_S denote the graph formed by the set of edges S together with the corresponding nodes. For all such $S \subseteq E_p$, G_S can be decomposed into paths with both endpoints being odd degree nodes in G_p.

Proof: In G_s, pair (arbitrarily) up odd degree vertices and connect each pair by an artificial edge, get the Euler tour of the resultant graph, and delete the artificial edges from the tour to get a set of paths each of which connect two odd degree vertices of G_s. Please note that each odd degree vertex in G_s must have an odd degree in G_p. Therefore, each of these paths connects two odd degree vertices in G_p.

The above theorem can be utilized to develop an algorithm for determining if a survivable route for the logical ring can be found in the physical topology G_p. From the earlier theorems, we know that if G_p has only nodes of even degree, then survivable routes exist and they can be found fairly easily. If G_p has nodes of both even and odd degree, then existence of survivable routes will depend on the existence of a subset of edges $S \subseteq E_p$, whose removal from G_p, would make the remaining graph eulerian. We now discuss a method to test the existence of such a set S. The algorithm has two phases.

A. Phase I

It is known that in any graph the number of nodes with odd degree is even [1]. Suppose G_p contains $2k$ nodes of odd degree. From the above theorem, we know that if an edge minimal subset S of E_p exists, whose removal makes the remaining graph eulerian, then this set S, together with the corresponding nodes makes a forest. It is also known that this forest can be decomposed into k paths with the endpoints of the paths being the nodes with odd degree in G_p.

To illustrate the execution of our algorithm, we choose a non-planar version of ARPANET with 20 nodes and 32 links [11] shown in figure 2. In this example, the following set of nodes have odd degree, $\{1, 2, 3, 4, 5, 6, 7, 9, 11, 12, 13, 15, 16, 17, 18, 20\}$. We refer to these nodes as *problem* nodes. The "problem" associated with these nodes (i.e., their odd degree) can be resolved by removal of an edge incident on these nodes. Suppose that v is a node of odd degree in G_p and it is adjacent to t other nodes $u_1, u_2, \ldots, u_t$. If one of these nodes $u_i, (1 \leq i \leq t)$ is also of odd degree, then removal of the edges $(v - u_i)$ will resolve the "problem" associated with both the nodes v and u_i. However, if the node u_i is of even degree, then removal of the edge $(v - u_i)$ will resolve the problem associated with node v, but will introduce a new problem at node u_i. In a sense, the removal of this edge merely shifts the problem from node v to node u_i. In this case, we should examine all the adjacent nodes of u_i (except v) and remove one such edge to "fix" the problem at node u_i. The process continues till we find an edge whose removal does not introduce a new "problem" at another node.

In the network of figure 2, node 3 is a "problem" node because it is a node of odd degree. We can "fix" the problem at node 3 by removing any one of the edges e_2, e_9 or e_{10}. The removal of the edge e_9 fixes node 3's problem and does not introduce any new problem. In fact, in addition to solving node 3's problem it also "solves" node 7's problem. Therefore, the removal of the edge e_9 is a "complete solution" for the nodes 3 and 7. However, if instead of e_9, e_{10} is removed then it solves the problem at node 3 but introduces a problem at node 8. Thus removal of the edge e_{10} is only a "partial solution" to the problem at node 3. To obtain a complete solution from this partial solution and to fix the problem at node 8 introduced by the deletion of the edge e_{10}, we need to delete one additional edge at node 8. If e_{13} is deleted, then it solves the problem at node 8 as well as at node 9 and does not introduce any new problem. Thus the removal of the set of edges $\{e_{10}, e_{13}\}$ is a complete solution for the nodes 3 and 9. Similarly, the removal of the set of edges $\{e_{10}, e_{16}\}$ is a complete solution for the nodes 3 and 12. If e_{15} is removed, after the removal of e_{10}, the problem propagates to node 10. A complete solution can be derived by extending the partial solution obtained by the path formed by e_{10} and e_{15}. The set of complete solutions obtained by extending the partial solution $\{e_{10}, e_{15}\}$ are $\{\{e_{10}, e_{15}, e_{11}(7)\}, \{e_{10}, e_{15}, e_{14}(9)\}, \{e_{10}, e_{15}, e_{17}, e_{26}(20)\}, \{e_{10}, e_{15}, e_{17}, e_{31}(17)\}, \{e_{10}, e_{15}, e_{17}, e_{27}(16)\}\}$. The number within () indicates the "other" node (besides node 3), whose problem is "solved" by the removal of the corresponding set of edges. We refer to this node as the *terminal node*. As a first step towards finding the survivable route, we first construct all such "complete solutions". For the graph of figure 2, all such complete solutions are shown in table I. The complete solutions can be obtained by performing *depth first search* on the graph G_p. Please note that to avoid redundancy, we only list the complete solutions, where the index of the problem node is less than the index of the terminal node.

Problem node	Solution Edges	Terminal Node
1	e_1	2
1	e_7	6
1	e_2	3
2	e_3	4
2	e_4	5
3	e_9	7
3	e_{10}, e_{13}	9
3	e_{10}, e_{16}	12
3	e_{10}, e_{15}, e_{11}	7
3	e_{10}, e_{15}, e_{14}	9
3	$e_{10}, e_{15}, e_{17}, e_{26}$	20
3	$e_{10}, e_{15}, e_{17}, e_{31}$	17
3	$e_{10}, e_{15}, e_{17}, e_{27}$	16
4	e_5	5
4	e_{12}	11
5	e_6	6
6	e_8	7
7	e_{11}, e_{14}	9
7	e_{11}, e_{15}, e_{13}	9
7	e_{11}, e_{15}, e_{16}	12
7	e_{11}, e_{17}, e_{31}	17
7	e_{11}, e_{17}, e_{26}	20
7	e_{11}, e_{17}, e_{27}	16
9	e_{30}	11
9	e_{14}, e_{17}, e_{31}	17
9	e_{14}, e_{17}, e_{26}	20
9	e_{14}, e_{17}, e_{27}	16
9	e_{14}, e_{15}, e_{16}	12
11	e_{18}	13
12	e_{20}	18
12	e_{19}, e_{21}	18
12	e_{19}, e_{29}	15
12	e_{19}, e_{32}	20
13	e_{23}	15
13	e_{24}	17
15	e_{28}	16
16	e_{22}	18
16	e_{27}, e_{26}	20
16	e_{27}, e_{31}	17
17	e_{31}	19
17	e_{25}	20

TABLE I

B. Phase II

Once the set of "complete solutions" associated with a pair of nodes is obtained, we try to combine them to obtain complete solution to all the problem nodes. If such a solution can be found (the solution is a set of edges S indicated earlier), then survivable routes for the ring in the given physical topology can be found. Otherwise, survivable routing for the ring in G_p is impossible. To demonstrate the combination process, we again use the example of figure 2. As noted earlier, the problem nodes in this example are $\{1, 2, 3, 4, 5, 6, 7, 9, 11, 12, 13, 15, 16, 17, 18, 20\}$. From the set of complete solutions in table I, we choose one solution after another till all the problem nodes are "fixed". The process constructs the set S (if it exists) and works

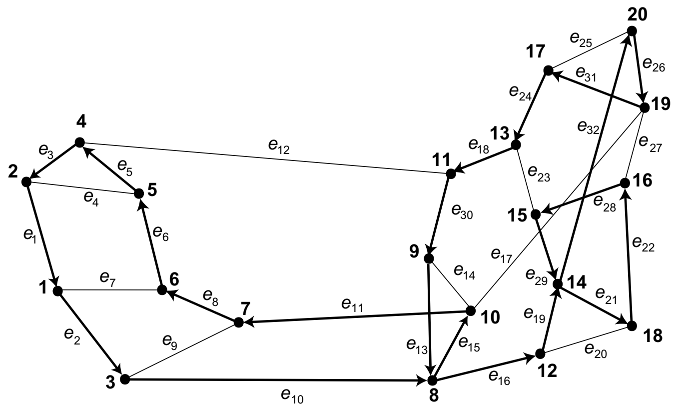

Fig. 2. Euler Tour in $G'_p = (V_p, E_p - S)$

as follows. The set S is empty initially.

Attempt 1: Include the edge e_1 in S. This fixes the problem for nodes 1 and 2. Mark these nodes as fixed. Since the next higher indexed node that needs fixing is node 3, we include the edge e_9 in S. Inclusion of this node fixes the problem for nodes 3 and 7. Mark these nodes as fixed. To fix the next higher indexed problem node (4) we choose e_5 for inclusion in S. This fixes the nodes 4 and 5. Mark these nodes as fixed. Now to fix the next higher indexed problem node (6) we need to choose e_8 which would fix the problem of node 7. But the problem of node 7 was already fixed when we included the edge e_9 in S. As such we cannot choose e_8 now, and these set of edges $\{e_1, e_9, e_5\}$ cannot lead to a complete solution for all the nodes.

Because of this failure to find a solution, we backtrack and make a second attempt.

Attempt 2: This time around, after choosing e_1 and e_9, instead of e_5, we choose e_{12}, which fixes the problem at node 11. Next, to fix the problem at node 5, we choose e_6 which also fixes the problem at node 6. Now, the unfixed problem node with the smallest index is 9 . We cannot choose the edge e_{30} to fix node 9's problem because it would also have solved problem at node 11. However, the problem at node 11 is already solved when we chose the edge e_{12}. Therefore, instead of choosing e_{30}, we choose the set $\{e_{14}, e_{17}, e_{31}\}$, which fixes node 17. To fix the problem at node 12, we choose e_{20} which also fixes the problem at node 18. To fix the problem at node 13, we choose e_{23} which also fixes the problem at node 15. To fix the problem at node 16, we cannot choose e_{22} because the problem with its terminal node (18) was already fixed by e_{20}. Therefore, we choose the edge set $\{e_{27}, e_{26}\}$, which fixes the node 20. If S = $\{e_1, e_9, e_{12}, e_6, (e_{14}, e_{17}, e_{31}), e_{20}, e_{23}, (e_{27}, e_{26})\}$, the set of edges constructed by this process is removed from the graph G_p, no nodes will have an odd degree in the remaining graph. However, if this set of edges are removed, the remaining graph will also be *disconnected*, because node 19 will have all its incident edges removed. Accordingly, this S is also not a feasible solution.

We continue with the backtracking process, till we find a feasible solution or conclude that no feasible solution exists. In the example of figure 2, during the sixth attempt of the backtracking process we find the following feasible solution: $\{e_7, e_4, e_9, e_{12}, (e_{14}, e_{17}, e_{27}), e_{20}, e_{23}, e_{25}\}$. If this set of edges is removed, all the nodes will be of even degree and as such an Euler tour can be constructed in the new graph. The Euler tour is shown in figure 2. The tour is: ($2 \to 1 \to 3 \to 8 \to 12 \to 14 \to 18 \to 16 \to 15 \to 14 \to 20 \to 19 \to 17 \to 13 \to 11 \to 9 \to 8 \to 10 \to 7 \to 6 \to 5 \to 4 \to 2$).

Suppose that the nodes of a 20 node logical ring are labeled from A to T, with the logical edges going from $A - B$, $B - C$, $\ldots$, $S - T$, $T - A$. Then the following mapping of the nodes of the logical graph to the nodes of the physical graph will create *surviable routes*. $A \to 2, B \to 1, C \to 3, D \to 8, E \to 12, F \to 14, G \to 18, H \to 16, I \to 15, J \to 20, K \to 19, L \to 17, M \to 13, N \to 11, O \to 9, P \to 10, Q \to 7, R \to 6, S \to 5, T \to 4$.

V. Conclusion

In this paper we have studied the issues related to survivable routing of a logical ring in a physical network of arbitrary topology. There are three important contributions of paper on this problem. First, we have given a necessary and sufficient condition for the existence of survivable routes. Second, we have shown the problem of determining whether or not the condition is satisfied by an arbitrary physical topology is NP-complete. Third, we have presented an algorithm for finding the survivable routes, if they exist in the physical network. We are currently investigating the issues related to the existence of survivable routes when the logical topology also has an arbitrary structure.

References

[1] J. A. Bondy and U. S. R. Murthy, Graph Theory with Applications, North Holland, 1976.

[2] O. Crochat and J. Y. Le Boudec, "Design Protection for WDM Optical Networks", IEEE JSAC, Vol. 16, No. 7, Sept. 1998.

[3] M.R. Garey, D.S. Johnson and R. E. Tarjan, "The planar Hamiltonian circuit problem is NP-complete", *SIAM Journal onf Computing*, vol. 5, no. 4, pp. 704-714, December 1976.

[4] O. Gerstel and R. Ramaswami, "Optical layer survivability: A services perspective", *IEEE Communications Magazine*, vol. 38, no. 3, pp. 104-113, March 2000.

[5] N. Ghani, S. Dixit and T. S. Wang,"On IP-over-WDM Integration", *IEEE Communications Magazine*, vol. 38, no. 3, pp. 72-84, March 2000.

[6] M. Medard, S. Finn and R. A. Barry, "WDM Loop-back Recovery in Mesh Networks", *Infocom '99, New York*, March, 1999.

[7] E. Modiano and A. Narula-Tam, "Survivable routing of logical topologies in WDM networks", *Proc. IEEE INFOCOM'01*, 2001.

[8] S. Ramamurthy and B. Mukherjee, "Survivable WDM Mesh Netwoks", Proc. of IEEE INFOCOM'99, pp. 744-751, March, 1999.

[9] S. Ramamurthy and B. Mukherjee, "Survivable WDM Mesh Netwoks: Part II -Restoration,", ICC '99, Vancouver, CA, June, 1999.

[10] G. H. Sasaki, C. F. Su and D. Blight,"Simple layout algorithms to maintain network connectivity under faults", *Proc. of 2000 Annual Allerton Conference*, Champaign Il, October 2000.

[11] T. E. Stearn and Krishna Bala, Multiwavelength Optical Networks - A Layered Approach, Addison Wesley, 1999.

[12] H. Zang, J. P. Jue and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *Optical Networks Magazine*, January 2000.