Dynamic Dual-Homing Protection in WDM Mesh Networks

Vinod M. Vokkarane[†], Jianping Wang[‡], Raja Jothi[†], Xiangtong Qi[*], Balaji Raghavachari[†], and Jason P. Jue[†]

[†] Department of Computer Science, The University of Texas at Dallas, Richardson, Texas

[‡] Department of Computer Science, Georgia Southern University, Statesboro, Georgia

[*] Department of Industrial Engineering & Engineering Management, HKSTU, Hong Kong

E-mail: [†]{vinod, raja, rbk, jjue}@utdallas.edu, [‡]jpwang@gasou.edu, [*]ieemqi@ust.hk

***Abstract*— A fault-tolerant scheme, called *dual homing*, is generally used in IP-based access networks to increase the availability of the network. In dual homing, a host is connected to two different access routers; therefore, it is unlikely that the host will be denied access to the network as the result of an access line break, a defective power supply in the access router, or congestion of the access router. However, dual homing itself cannot provide survivability due to possible failures in a WDM core network. To provide survivability in the core network, *protection* and *restoration* techniques must be used. The dual homing architecture introduces new issues for protection and restoration design, especially when providing survivability against two independent failures, one in the access network and another in the core network. This paper studies the protection design in the core network given the dual-homing infrastructure in WDM mesh networks. Several solutions are proposed, and the performance of the different solutions are compared. We also prove that one of the proposed algorithms gives a solution that, in the worst case, is at most $\frac{4}{3}$ times the cost of the optimal solution.**

I. Introduction

IP-over-WDM networks are considered as the major components of the next-generation Internet. One important issue in IP-over-WDM networks is survivability. Survivability is the capability of the network to function in the event of node or link failures. *Dual homing* is generally used to increase survivability in the access network. In dual homing, a host in the access network can be connected to two IP routers, which are connected to underlying edge optical cross connects (OXCs) of the core network. Fig. 1 illustrates the IP-over-WDM dual-homing network architecture. The main objective of dual homing is to provide enhanced survivability to protect against access node failures caused by system malfunction, scheduled outage, or an access link failure. Dual-homing architecture design has been widely studied in self-healing ring networks [1], [2], [3].

The survivability in WDM networks is implemented using *protection* and *restoration* techniques. Protection is a static mechanism to protect against failure, where the resource for both the primary and the backup lightpaths are reserved prior to the data communication. Restoration on the other hand, is a dynamic mechanism where the backup lightpath is not set up until the failure occurs. Survivability using these techniques is usually provided to handle single link failures in the core network. Existing literature on protection and restoration in WDM networks can be found in [4], [5].

There have been several efforts for providing survivability for a dual-homed IP-based access network over WDM-based core networks [6], [7]. In all these works, the authors consider providing survivability separately at the IP layer as well as the WDM layer. In [6], the authors discuss how to support dual-homing in passive optical networks; while [7] studies survivability in IP-over-WDM networks and provides different protection types (unprotected, protected, and dual homing) for each IP link in order to keep the networks connected in the event of link failure. The focus of our paper is to provide an integrated solution for providing survivability in an IP-over-WDM mesh network.

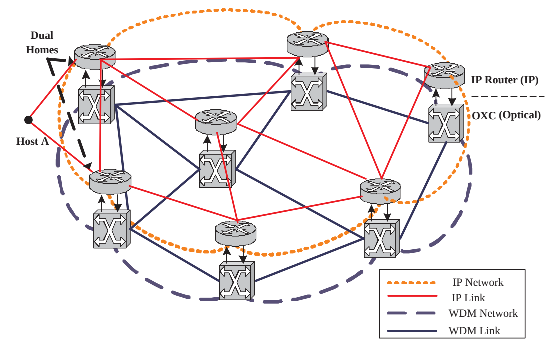


Fig. 1. Dual-homing architecture.

In this paper, we integrate dual homing and protection, using dual homing to protect against a single access node failure, and protection to handle a single link failure in the optical core. We consider the problem of providing protection in the core network, given the dual-homing architecture. We assume that the failures in the access network and the core network are independent, which means that the failure of the access node and the failure of the link in the core network can occur simultaneously. By considering the dual-homed IP-over-WDM architecture (Fig. 1), we observe that, at any given time, each host transmits data to the destination only through one of the dual homes. Based on this observation, we see that only one of the primary paths will be utilized at any given time. Also, this property leads to fewer restrictions on the disjointness constraint between the two primary and two backup paths from the dual homes to each of the destinations. We observe that by providing an integrated solution, we can obtain significant cost benefits as compared to handling survivability separately at each of the layers (IP and WDM). In our study, we focus on the Dynamic Dual-Homing Protection problem, wherein connection requests arrive one at a time and the requests are handled one after the other.

The rest of the paper is organized as follows. The network architecture of Dual-Homing Protection is described in Section II. The detailed problem description is presented in Section III. In Section IV, we propose a number of different heuristics to solve the Dynamic Dual-Homing Protection problem. In Section V, we evaluate the performance of all the proposed algorithms. We also prove that one of the proposed algorithms gives a solution that, in the worst case, is at most $\frac{4}{3}$ times the cost of the optimal solution. Finally, the conclusion and future work are presented in Section VI.

II. Network Architecture

In this paper, we consider an integrated IP-over-WDM network as shown in Fig. 1, where a host in the access network is connected to two IP routers in the IP-based access network. Each IP router is connected to an optical cross connect (OXC), which in turn is connected to other OXCs that constitute the all-optical WDM core network. Dual homing is provided at the access level to provide survivability against a single IP router (node) or access link failure. In dual homing, two link-disjoint paths connect the host to its dual homes. The dual-homed IP routers are connected to the underlying OXCs. The OXCs convert the IP packets into optical signals and transmit packets over the WDM layer to the corresponding destinations.

In the event of an access node failure, we assume that the IP router fails, but the OXC connected to the router continues to carry express optical traffic. Therefore, by using dual homing, the access traffic can be shifted to the other home (access node), which in turn transmits the data traffic to the destination. We also observe that in the event an access link failure, the access network is survivable with dual homing. Hence, dual homing provides survivability against single link or node failure in the access network. In the event of a link failure in the core, we adopt link-disjoint dedicated path protection to provide survivability. In dual homing, we have two source OXCs, with only one source OXC transmitting data to a specific destination OXC at any given time. Therefore, we observe that, in most solutions, the primary paths between the two source OXCs to the destination OXCs need not necessarily be disjoint. Additionally, since we have to protect against a single-link failure in the core, it is best to maximize sharing of links between the two primary paths, so that we have fewer links to protect (while calculating a combined backup path). The detailed description of the problem and the solutions are given in the following sections.

III. Problem Description

A WDM network can be modeled as an undirected graph $G{<}V, E{>}$, where V is the set of OXCs and E is the set of WDM links. Let the wavelength cost of a WDM link $e \in E$ be $c(e)$. Let the maximum number of wavelengths in each link be W. Let each dynamic connection request, denoted by, r, be given by $\{\{s_1, s_2\}, d\}$, where s_1 and s_2 are two OXCs connected to the dual-homed access routers of Host K, and d is the destination OXC that in turn is connected to an IP router that connects to the destination Host D. Algorithms for determining the dual-homes for a Host K are given in [8]. Let the primary lightpath from s_1 to d be denoted by p_a^1 and the link-disjoint backup lightpath from s_1 to d be denoted by p_b^1. Similarly, the primary lightpath from s_2 to d is denoted by p_a^2 and the link-disjoint backup lightpath from s_2 to d is denoted by p_b^2. Let L_r be the set of all links used in the primary and backup lightpaths for the connection request r. L_r is given by $p_a^1 \cup p_b^1 \cup p_a^2 \cup p_b^2$.

If the core network is reliable, p_a^1 and p_a^2 are not necessarily disjoint as shown in Fig. 2(a). Even if p_a^1 and p_a^2 are disjoint, they cannot protect simultaneous failures in the access network and the core network, as shown in Fig. 2(b). If the access node of s_1 is down, and one link in p_a^2 is also down, data cannot be sent to d. In order to provide dual-homing protected service, we need p_b^1 and p_b^2 to protect the lightpaths p_a^1 and p_a^2. We have the following observations:

- p_a^1 and p_a^2 are not necessarily disjoint.
- p_b^1 and p_b^2 are not necessarily disjoint.
- p_a^1 and p_b^2 are not necessarily disjoint.
- p_a^2 and p_b^1 are not necessarily disjoint.
- p_a^1 and p_b^1 must be disjoint.
- p_a^2 and p_b^2 must be disjoint.

Fig. 2(c) illustrates these observations.

In this paper, we study the dynamic dual-homing protection problem in which the objective is to route p_a^1, p_b^1, p_a^2, and p_b^2 for an arriving connection request r such that the wavelength cost is minimized. We assume that each connection request is for a single wavelength on any link, that full-wavelength conversion capability is available at each OXC in the core network, and that the wavelength conversion cost is not significant. We only consider the wavelength cost in the total cost. Therefore, given a new connection request r, the objective of the dynamic dual-homing protection problem to find L_r such that total cost C_r is minimum, where,

$$C_r = \sum_{e \in L_r} c(e). \tag{1}$$

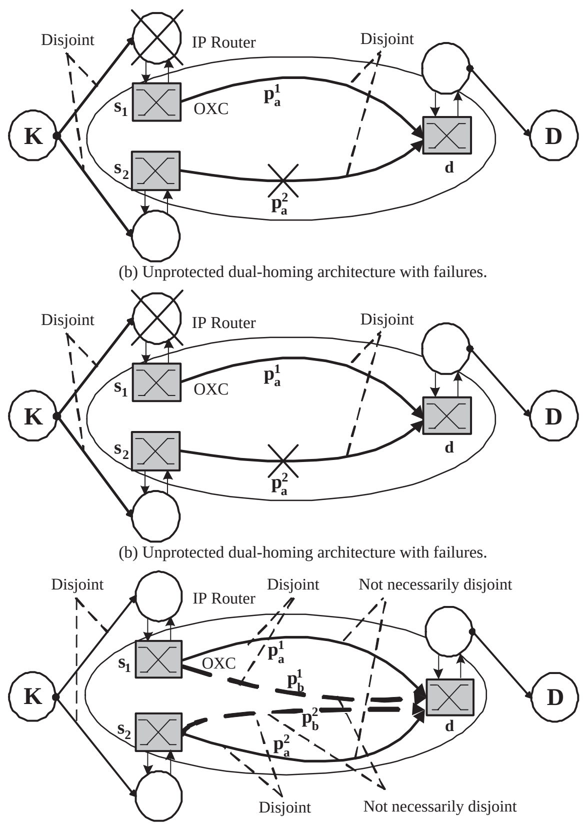
The computational complexity for this problem is still open, and is believed to be NP-hard.

IV. Dynamic Dual-Homing Protection Algorithms

We now propose several heuristics for dynamic dual-homing protection. These heuristics can be classified into two categories: one category is based on a minimum cost network flow model and the other category is based on a minimum Steiner tree model. The minimum cost network flow model computes minimum-cost link-disjoint paths which satisfy the disjointness between the primary path and the backup path [9]. On the other hand, the minimum Steiner tree model considers the sharing among the primary paths and sharing among the backup paths.

The first heuristic is based on the minimum cost network flow model. It finds the optimal link-disjoint primary and backup lightpaths from one of the dual homes to the destination, and then finds the optimal link-disjoint primary and backup lightpaths from the other dual home to the destination. The solution obtained by MCNFH is as shown in Fig. 3(a).

The second heuristic is also based on the minimum cost network flow model and is a generalization of the first heuristic in which we first select a random node known as the *branching node*. From each of the dual homes we compute two minimum-cost link-disjoint paths to the branching point, and from the branching node we compute two minimum cost link-disjoint paths to the destination. This process is repeated, selecting each node as the branching node, and then selecting the minimum cost solution. The first heuristic is a special case of second heuristic in which the destination is chosen as the branch-

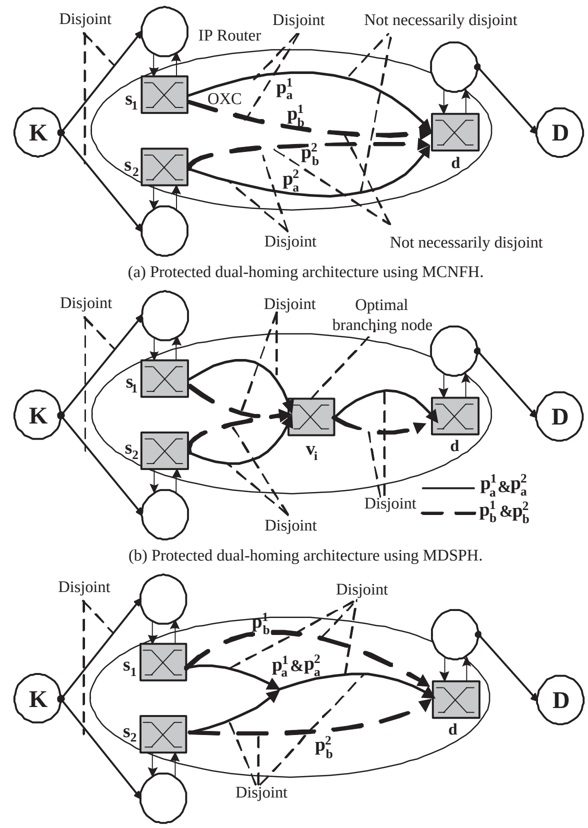


(b) Unprotected dual-homing architecture with failures.

(b) Unprotected dual-homing architecture with failures.

(c) Protected dual-homing architecture.

Fig. 2. Dual homing and protection architectures.



(a) Protected dual-homing architecture using MCNFH.

(b) Protected dual-homing architecture using MDSPH.

(c) Protected dual-homing architecture using MSTH.

Fig. 3. Dynamic dual-homing protection using different heuristics.

ing node. The solution obtained by MDSPH is as shown in Fig. 3(b).

The last heuristic is based on the minimum Steiner tree model. It finds the minimum Steiner tree connecting the dual homes to the destination, then provides path protection from each home to the destination. The solution obtained by MSTH is as shown in Fig. 3(c). We now describe each of the algorithms in detail.

A. Minimum Cost Network Flow Heuristic (MCNFH)

The *minimum cost network flow heuristic* (MCNFH) first finds the minimum cost link-disjoint primary and backup lightpaths from one of the dual homes to the destination, then changes the link cost and finds the minimum cost link-disjoint primary and backup lightpaths from the other dual home to the destination. We can use the minimum cost network flow (MCNF) algorithm to find the minimum-cost link-disjoint primary and backup lightpaths from one home to the destination. Initailly, we assign the capacity of link to be unity in order to force the primary and the backup lightpaths from s_1 to d as well as from s_2 to d to be disjoint. C gives the total cost for the primary and backup lightpaths from s_1 to d as well as from s_2 to d, and S gives the links used for those lightpaths. Note that the order in which the paths are computed has a bearing on the total cost. Hence, we first find the primary and backup lightpaths from one dual home to the destination, and then find the primary and backup lightpaths from the other dual home to the same destination. We repeat the process by finding the primary and backup lightpaths from other dual home to the destination, and then finding the primary and backup lightpaths from the first dual home to the same destination. Of the two resulting solutions, we choose the one that results in the minimum cost. The detailed algorithm is given below.

1) Call the MCNF algorithm to find p_a^1 and p_b^1, of cost $C_1(s_1, d)$, and let $S_1 = p_a^1 \cup p_b^1$.
2) Assign $c(e) = 0 \; \forall \; e \; \in \; S_1$ (encourage sharing).
3) Call the MCNF algorithm to find p_a^2 and p_b^2, of cost $C_1(s_2, d)$. Let $C = C_1(s_1, d) + C_1(s_2, d)$ and $S_1 = S_1 \cup p_a^2 \cup p_b^2$.
4) Hence, we obtain the minimum cost (C) primary and backup paths from s_1 and s_2 to d. Now, we repeat the above step by selecting the other dual home, s_2 as the first node for which the primary and backup paths to the destination is computed.
5) Call the MCNF algorithm to find p_a^2 and p_b^2, of cost $C_2(s_2, d)$, and let $S_2 = p_a^2 \cup p_b^2$.
6) Assign $c(e) = 0 \; \forall \; e \; \in \; S_2$ (encourage sharing).
7) Call the MCNF algorithm to find p_a^1 and p_b^1, of cost $C_2(s_1, d)$. Let $S_2 = S_2 \; \cup \; p_a^1 \cup p_b^1$.
8) If $(C > C_2(s_1, d) + C_2(s_2, d))$, then $C = C_2(s_1, d) + C_2(s_2, d)$ and $S = S_2$. Otherwise, $S = S_1$.

MCNFH gives a solution that, in the worst case, is at most $\frac{4}{3}$ times the cost of the optimal solution (the proof is given in the Appendix).

B. Minimal Disjoint Segment-Pair Heuristic (MDSPH)

The minimal disjoint segment-pair heuristic (MDSPH) is based on the observation that in order to minimize the wavelength cost, the two primary paths are either disjoint or there is a branching node which connects two homes and the destination. If the two primary paths are disjoint, we can take the destination as the branching node. The MDSPH tries to find the branching node such that the total wavelength cost used in both primary paths and backup paths is minimum. Let S_i be the set of links used in the primary paths and backup paths, when node $v_i \in V$ is chosen as the branching node. Let C_i be the total wavelength cost when the links in S_i are used for connection request r_k. Let v_b be the branch node with minimum wavelength cost for the primary paths and backup paths. MDSPH works as follows.

1) Set $C = \infty$, $S_i = \emptyset$
2) For each Node $v_i \in V$, assign v_i to be the branching node and do the following:
   (a) Find two link-disjoint paths using MCNF from s_1 to v_i, let the cost be C_1^k and the links used be $L_1(v_i)$. $S_i = S_i \cup L_1(v_i)$.
   (b) Find two link-disjoint paths using MCNF from s_2 to v_i, let the cost be C_2^k and the links used be $L_2(v_i)$. $S_i = S_i \cup L_2(v_i)$.
   (c) Find two link-disjoint paths using MCNF from v_i to d, let the cost be C_3^k and the links used be$L_3(v_i)$. $S_i = S_i \cup L_3(v_i)$.
   (d) if ($C > C_1^k + C_2^k + C_3^k$) then $C = C_1^k + C_2^k + C_3^k$ and $v_b = v_i$

In the algorithm described above, C gives the total cost of the solution found by MDSPH, and v_b gives the branch node. If $v_b = v_i$, then S_i includes all links used for the primary paths and backup paths.

MDSPH always finds a solution if a feasible solution exists. The solution obtained can be no worse than MCNFH, since MCNFH is a special case of MDSPH where the destination serves as the branching node.

C. Minimum Steiner Tree Heuristic (MSTH)

The minimum Steiner tree heuristic is based on the fact that the minimum Steiner tree is the best approach to connect three nodes with minimum cost. The idea behind the minimum Steiner tree heuristic (MSTH) is to find a minimum cost tree which is designated as the primary tree and which provides path protection to the dual homes.

Although the minimum Steiner tree problem is NP-hard in the general case, it is polynomial-time solvable when there are only three nodes covered. We know that any tree covering three nodes will have at most one branching (or splitting) node. Once the branching node is determined, the minimum cost Steiner tree is obtained by finding the shortest paths from the branching point to each of the end nodes (dual homes and the destination). In order to find the optimal branching node in a network with N nodes, we can consider each Node $v_i \in V$ to be the branching point and then form a tree, T_i, resulting in N different trees. The optimal Steiner tree, T_{opt}, is given by the minimum cost tree of the N different enumerated trees.

1) Find the optimal Steiner tree connecting the dual homes and the destination node. This minimum cost tree is designated as the primary lightpaths, p_a^1 and p_a^2 respectively.
2) Assign the cost of the links in p_a^1 to ∞.
3) Find the shortest path p_b^1 from s_1 to d
4) Assign the cost of the links in p_a^1 and in p_b^1 to 0.
5) Assign the cost of the links in p_a^2 to ∞.
6) Find the shortest path p_b^2 from s_2 to d

TABLE I
TIME COMPLEXITY: DUAL-HOMING PROTECTION ALGORITHMS.

Algorithm	Time Complexity
MCNFH	O(N^2)
MDSPH	O(N^3)
MSTH	O(N^3)

In Table I, we compare the time complexities of the proposed dual-homing protection heuristics. We see that the MCNFH has a worst-case time complexity $O(N^2)$, the generalized MDSPH has a worst-case time complexity $O(N^3)$, and the MSTH has a worst-case time complexity $O(N^3)$.

V. SIMULATION RESULTS

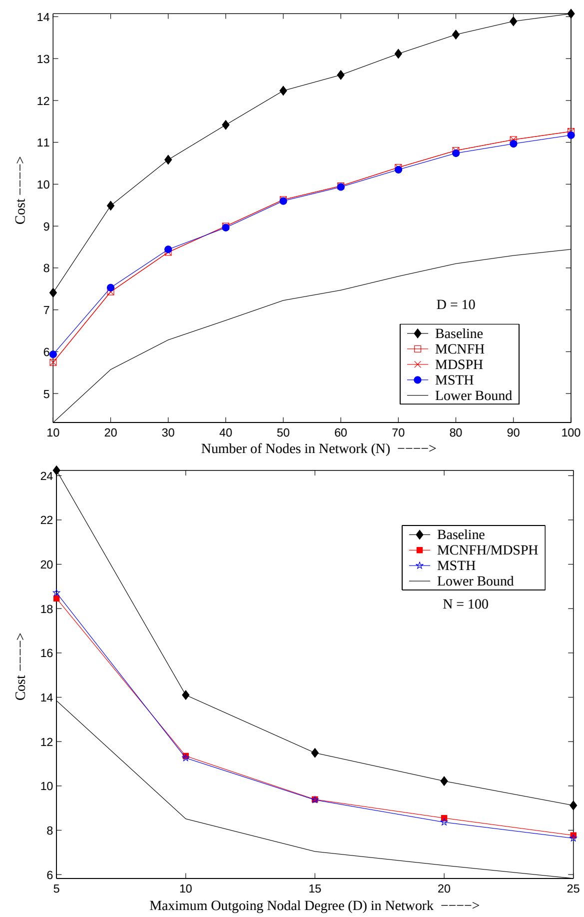


Fig. 4. (a) The average cost versus number of nodes in the network (N), and (b) The average cost versus maximum nodal degree (D) for the proposed algorithms.

In this section, we develop a simulation model in order to analyze the performance of the proposed heuristics for the dynamic dual-homing protection problem. We are interested in comparing the average total cost of the solutions obtained using MCNFH, MDSPH, and MSTH. The important simulation parameters include the network size, N, the maximum outgoing degree at each node, D. Given a group of parameters $<N, D>$, we randomly generate a network with N nodes. The outgoing degree of each Node i, is uniformly distributed in $[1, 2, \ldots, D]$.

The cost of each link is set to unity. We randomly select two nodes to be the dual homes for a new connection request and another node to be the destination.

For each group of parameters, instances are generated until 1000 instances have feasible solutions by using MCNFH. All these instances are also solved by MDSPH and MSTH. Fig. 4(a) plots of average cost for the proposed algorithms versus different values of N, when D is set to 10. In order to show the advantage of the integrated solution, we compare the algorithms with a baseline case wherein sharing between any of the primary and backup paths is not allowed. By considering that a dual-homed IP layer exists above the WDM core network, we can see that the cost of providing protection in the core network using MCNFH, MDSPH, and MSTH is significantly lower than the baseline case. We also observe that MCNFH and MDSPH incur the same cost for the network scenarios considered. The performance of MSTH is also close to that of the network flow-based algorithms. The cost obtained by MSTH is better than MCNFH and MDSPH, for larger networks. By using the 4/3 approximation result, we also give a tight lower-bound for the dynamic dual-homing protection problem.

Fig. 4(b) plots of average cost versus the for the proposed algorithms versus different values of D, when N is set to 100. We observe that MSTH outperforms MCNFH as D increases, since in MSTH, as the network connectivity (D) increases there is a higher possibility of the dual homes sharing common links on their primary tree. We have assumed that the two dual homes are randomly chosen to be any of the nodes in the network. In the case we consider a practical setup where the dual homes are close to each other, there would be a higher possibility of the dual homes sharing common links on their primary tree, so as to reduce the total cost.

VI. Conclusion

We investigate the survivability issue in IP-over-WDM networks when a dual-homing architecture is provided in the access network. Our goal is to provide survivability for such an infrastructure subject to two independent failures, one failure from the access network and one failure from the core network. Three new heuristics, namely MCNFH, MDSPH, and MSTH are proposed. We also derive a tight lower-bound on the optimal solution for the dynamic dual-homing protection problem. We observe that by following an integrated approach that considers the dual-homed IP-over-WDM architecture as compared to an independent solution at each layer (IP and WDM), we can significantly reduce the cost incurred to provide protection in the WDM core network.

Areas of future work include introducing the concept of shared path protection into our integrated approach. In this paper, we consider that the source host is connected to dual homes but the destination is connected to a single home. It would be interesting to see how the proposed algorithms perform in a situation wherein both the source and destination hosts are connected to dual homes. Also, in the simulation model, we have assumed that the dual homes are chosen randomly for each request. We would like to consider a more practical scenario in which the dual homes are close to each other. This scenario may result in the primary and backup paths sharing a higher number of common links, thereby further reducing the cost of the integrated approach.

Appendix
MCNFH: A 4/3-approximation Algorithm

Conjecture 1: We conjecture that the dynamic dual-homing problem is NP-hard.

A. The Minimum Cost Network Flow Heuristic (MCNFH) variation

In what follows, we present a 4/3-approximation algorithm for the dynamic dual-homing problem and show that the cost of the obtained solution is at most 4/3 times the cost of an optimal solution. The minimum cost network flow (MCNF) algorithm is a polynomial time algorithm for finding the minimum cost edge-disjoint paths between any two nodes of a given graph. MCNFH uses the MCNF algorithm as a subroutine to find a feasible solution for dynamic dual-homing problem. The MCNF algorithm, when given a Graph G, Source K, and Destination D, returns the minimum cost edge-disjoint paths from K to D. Given below are the details of MCNFH.

MCNFH(G, s_1, s_2, d)

1) $f_1 \leftarrow$ MCNF(G, s_1, d) + MCNF(G, s_2, d).
2) $f_2 \leftarrow$ MCNF(G, s_2, d) + MCNF(G, s_1, s_2).
3) $f_3 \leftarrow$ MCNF(G, s_1, d) + MCNF(G, s_2, s_1).
4) Return the lowest cost solution among f_1, f_2, and f_3.

From the algorithm above, we can see that each of the three solutions (f_1, f_2, and f_3) makes exactly two calls to MCNF, and each call to the MCNF algorithm satisfies the requirement of finding two edge-disjoint paths from a source to a destination. For f_1, the two calls independently find two edge disjoint paths each from s_1 and s_2 to d. For f_2 (or f_3), while the first call finds two edge-disjoint paths from s_2 to d (s_1 to d resp.), the second call finds two edge-disjoint paths from s_1 to s_2 (s_2 to s_1 resp.), thereby finding two edge-disjoint paths from s_1 to d (s_2 to d resp.) indirectly. The final solution returned by MCNFH is feasible as it contains two edge-disjoint paths each from s_1 and s_2 to d.

B. Approximation Analysis

We now show that any solution returned by MCNFH is at most 4/3 times the cost of an optimal solution.

Lemma 1: In any optimal solution OPT, there exist two nodes u and v (u and v could be the same node, and u and/or v could be s_1, s_2, or d) such that the paths s_1 to u, s_1 to v, s_2 to u, and s_2 to v are all edge-disjoint. If there is no such u and v, then the cost of OPT is at least 3/4 times the cost of the solution obtained from MCNFH.

Proof: In any feasible solution, let p_a^1 and p_b^1 be the two edge-disjoint paths from s_1 to d, and let p_a^2 and p_b^2 be the two edge-disjoint paths from s_2 to d. If d were chosen to be u and v, the paths p_a^1, p_b^1, p_a^2 and p_b^2 might not necessarily be edge-disjoint. If they are edge-disjoint, the proof is complete. But if they are not edge-disjoint, then one of the following must be true:

1) Only paths p_a^1 and p_a^2 are not edge-disjoint.
2) Only paths p_b^1 and p_b^2 are not edge-disjoint.
3) Only paths p_a^1 and p_b^2 are not edge-disjoint.

4) Only paths p_b^1 and p_a^2 are not edge-disjoint.
5) (1) and (2).
6) (3) and (4).
7) (1), (2), and (3).
8) (1), (2), and (4).
9) (3), (4), and (1).
10) (3), (4), and (2).
11) (1), (2), (3), and (4).

For the first 6 cases, we now show (not necessarily in the same order) how to find u and v such that the lemma is true.

Case 5: Intially, set u and v to d. Obviously, p_a^1 and p_a^2 meet at u, and p_b^1 and p_b^2 meet at v. Since p_a^1 and p_a^2 are not edge-disjoint, there exists a common node u' (u' could be s_1 or s_2) where p_a^1 and p_a^2 meet for the first time. Set u to u'. Similarly, since p_b^1 and p_b^2 are not edge-disjoint, there exists a common node v' (v' could be s_1 or s_2) where p_b^1 and p_b^2 meet for the first time. Set v to v'. Since, in any feasible solution, p_a^1 and p_b^1 (and p_a^2 and p_b^2) must be edge-disjoint, the paths s_1 to u, s_2 to u, s_1 to v, and s_2 to v are all edge-disjoint.

Case 6: This case is symmetric to Case 5.

Cases 1-4: Cases 1 and 2 are sub-cases of Case 5, and Cases 3 and 4 are sub-cases of Case 6. Thus, the analysis for Case 5 holds for Cases 1 and 2. Similarly, the analysis for Case 6 holds for Cases 3 and 4.

Now, for Cases 7 to 11, we show that either the cost of OPT is at least 3/4 times the cost of the solution obtained from MCNFH or there exists two points u and v in OPT such that the paths s_1 to u, s_2 to u, s_1 to v, and s_2 to v are all edge-disjoint.

Case 7: The two possible scenarios that one can imagine for this case are considered below.

(a) p_a^1 shares edges with p_b^2 even before p_b^1 meets p_b^2 or p_a^1 meets p_a^2 (see Fig. 5): Let F_1 represent a feasible solution for the problem P_1 which asks for two edge-disjoint paths each from s_1 and s_2 to d. Let F_2 represent a feasible solution for the problem P_2 which asks for two edge-disjoint paths each from s_1 to d and s_2 to s_1, and let F_3 represent a feasible solution for the problem P_3 which asks for two edge-disjoint paths each from s_2 to d and s_1 to s_2. Using the edges in OPT, we can easily construct F_1, F_2, and F_3 by finding two edge-disjoint paths each from the two source nodes to the destination node as shown in Fig. 6(a)-(c). From Figs. 5 and 6(a)-(c), it can be easily verified that the sum of the costs of F_1, F_2, and F_3 is clearly 4 times the cost of OPT. Solutions f_1, f_2, and f_3 returned by lines 1,2 and 3 of MCNFH are feasible for problems P_1, P_2, and P_3, respectively. Since the MCNF algorithm finds minimum cost edge-disjoint paths for a given pair of nodes, the cost of $f_1 \leq F_1$, $f_2 \leq F_2$, and $f_3 \leq F_3$. Also, since MCNFH outputs a lowest cost solution, say f_m, among f_1, f_2, and f_3,

$$3f_m \leq f_1 + f_2 + f_3 \leq F_1 + F_2 + F_3 = 4 \cdot OPT.$$

The above inequality shows that OPT is at least 3/4 times the cost of the solution obtained from MCNFH.

(b) p_a^1 shares edges with p_b^2 after p_b^1 meets p_b^2 (see Fig. 7): Since p_a^1 shares edges with p_b^2 after p_b^1 meets p_b^2, once p_b^1 meets p_b^2 for the first time at a certain node u, both p_b^1 and p_b^2 are made to share the part of the path p_b^1 from u to d. Since p_a^1 and p_b^1 are edge-disjoint, the part of path p_b^1 from u to d is edge-disjoint from p_a^1. Let v be the first node at which p_a^1 meets p_a^2 for the first time. Since p_a^1 meets p_b^2 for the first time after u, and p_b^2 now uses the part of the path p_b^1 from u to d, p_a^1 is edge-disjoint from p_b^2. Thus, the paths s_1 to u, s_2 to u, s_1 to v, and s_2 to v are all edge-disjoint.

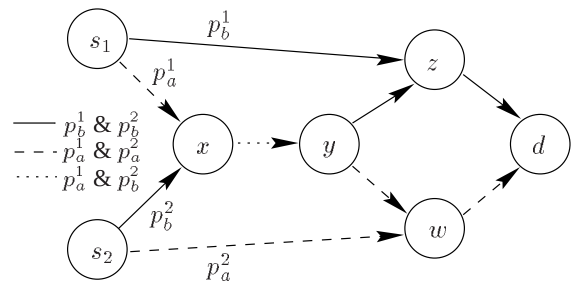


Fig. 5. *p_a^1 shares edges with p_b^2 even before p_b^1 meets p_b^2 or p_a^1 meets p_a^2.*

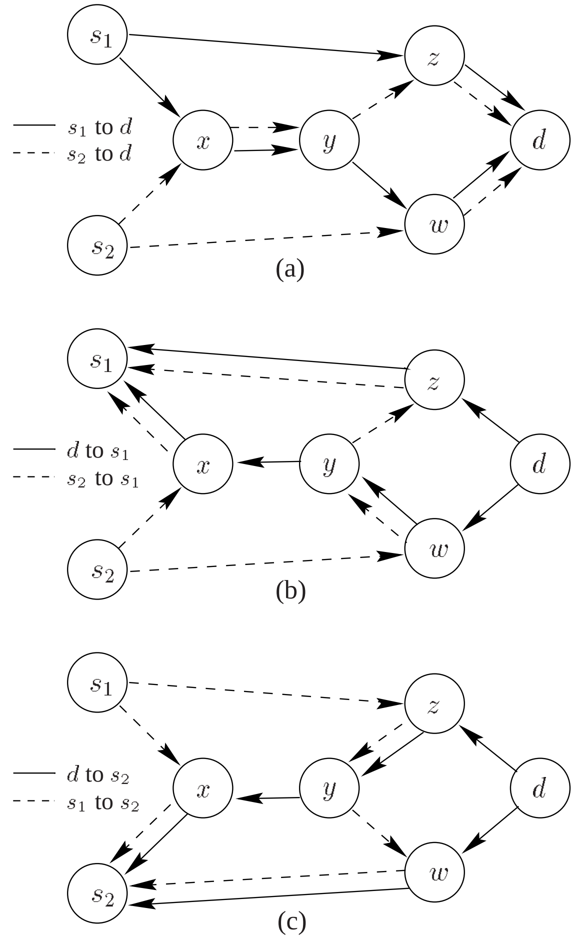


Fig. 6. (a) *f_1 constructed by two edge-disjoint paths each from s_1 and s_2 to d.* (b) *f_2 constructed by two edge-disjoint paths each from d and s_1 to s_2.* (c) *f_3 constructed by two edge-disjoint paths each from d and s_2 to s_1.*

Cases 8-10: Cases 8-10 are symmetric to Case 7, and thus the analysis for Case 7 holds for these cases as well.

Case 11: p_a^1 shares edges with p_a^2 and p_b^2, and p_b^1 shares edges with p_a^2 and p_b^2 (Fig. 8): Without loss of generality, let p_b^2 meet p_a^1 for the first time even before p_a^2 meets p_a^1. Set the node where p_b^2 and p_a^1 meet for the first time to be u. From u, let p_a^1 and p_b^2 use the part of the path p_b^2 from u to d. Since p_a^1 uses the part of the path p_b^2 from u to d, and p_a^2 and p_b^2 are edge-disjoint, p_a^1

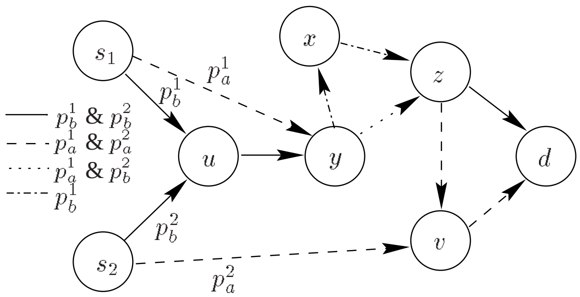


Fig. 7. *p_a^1 shares edges with p_b^2 after p_b^1 meets p_b^2.*

does not share edges with p_a^2 anymore. Let v be the point where p_a^2 meets p_b^1 for the first time. From v, let p_b^1 and p_a^2 use the part of the path p_a^2 from v to d. Since p_b^1 uses the part of the path p_a^2 from v to d, and p_a^1 and p_b^1 are edge-disjoint, p_b^1 does not share edges with p_b^2 anymore. Thus, the paths s_1 to u, s_2 to u, s_1 to v, and s_2 to v are all edge-disjoint.

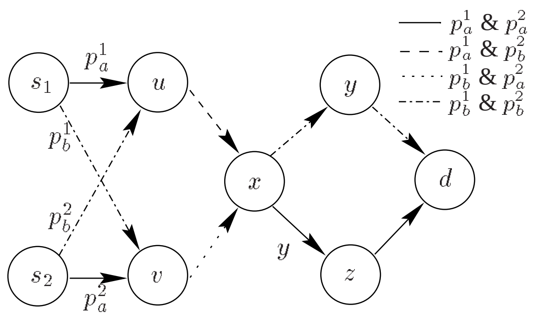


Fig. 8. *p_a^1 shares edges with p_a^2 and p_b^2, and p_b^1 shares edges with p_a^2 and p_b^2*

■

Theorem 1: The final solution returned by MCNFH is at most 4/3 times the cost of an optimal solution.

Proof: If the optimal solution OPT falls under Cases 7(a)-10(a) of Lemma 1, then, by Lemma 1, the cost of the solution returned by MCNFH is at most 4/3 times the cost of OPT.

If OPT falls under Cases 1-6, 7(b)-10(b) or 11 of Lemma 1, we show that the OPT can be converted into the canonical form, shown in Fig. 9. For these cases, by Lemma 1, there always exists two points u and v in OPT, such that the paths s_1 to u, s_1 to v, s_2 to u, and s_2 to v are all edge-disjoint. Let p_a^1 and p_a^2 be the paths from s_1 and s_2 to d, respectively, which pass through u. Let p_b^1 and p_b^2 be the paths from s_1 and s_2 to d, respectively, which pass through v. By definition, paths p_a^1 and p_b^1, and p_a^2 and p_b^2 must be edge-disjoint and thus there exists two edge-disjoint paths from u and v to d. There is a possibility that u and v are d itself, and thus paths from u and v to d are non-existent. In such a scenario, we interpret paths from u and v to d to be of cost zero (paths with no edges). Paths p_a^1 and p_a^2 must share a path from u to d, otherwise OPT is not optimal, which is a contradiction, likewise with paths p_b^1 and p_b^2 sharing a path from v to d. Thus, there exist two edge-disjoint paths from u and v to d, and any optimal solution that falls under Cases 1-6, 7(a)-10(a), and 11 can be converted into the canonical form shown in Fig. 9. We now show that the solution returned by MCNFH is 4/3 times the cost of OPT (represented in canonical form).

Consider Fig. 9. Let the cost of the path from s_2 to v be a, s_2 to u be c, s_1 to u be f, s_1 to v be e, u to d be d, and v to d be b. Since the cost of the path is the same regardless of the

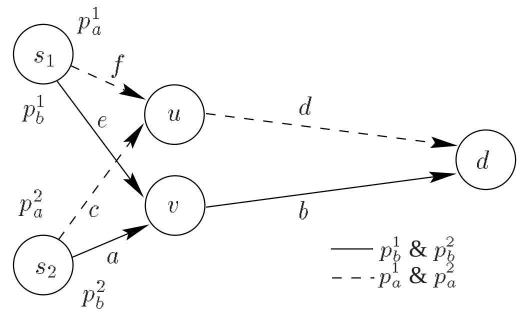


Fig. 9. *Canonical form.*

direction of traversal, without loss of generality, the direction of edges are not considered in the following analysis.

MCNFH considers three solutions and outputs the lowest cost solution, say f_m, among the three possible solutions f_1, f_2, and f_3. Clearly, f_1 is a feasible solution for the problem which asks for two edge-disjoint paths each from s_1 and s_2 to d, f_2 is a feasible solution for the problem which asks for two edge-disjoint paths each from s_1 to d and s_2 to s_1, and f_3 is a feasible solution for the problem which asks for two edge-disjoint paths each from s_2 to d and s_1 to s_2. Since the MCNF algorithm finds the minimum cost edge-disjoint paths from a source to a destination, the cost of any other pair of edge-disjoint paths between that source and destination is at least the cost of the solution obtained by the MCNF algorithm. Thus,

$$\begin{aligned} \text{Cost of } f_1 &\leq a+b+c+d+e+f+b+d \\ \text{Cost of } f_2 &\leq a+b+c+d+e+f+a+c \\ \text{Cost of } f_3 &\leq a+b+c+d+e+f+e+f \end{aligned}$$

Since MCNFH outputs the lowest cost solution f_m among the three possible solutions f_1, f_2, and f_3,

$$\begin{aligned} 3f_m &\leq f_1+f_2+f_3 \\ &\leq 4(a+b+c+d+e+f+b+d) \\ &= 4 \cdot \text{Cost of } OPT \end{aligned}$$

■

REFERENCES

[1] C. Lee and S. Koh, "A Design of the Minimum Cost Ring-Chain Network with Dual-homing Survivability: A Tabu Search Approach," *Computers Operations Research*, vol. 24, no. 9, pp. 883-897, 1997.

[2] M. Willebeek-LeMair and P. Shahabuddin, "Approximating Dependability Measures of Computer Networks: An FDDI Case Study," *IEEE-ACM Transactions on Networking*, vol. 5, no. 2, pp. 311-327, April 1997.

[3] J. Shi and J. Fonseka, "Analysis and Design of Survivable Telecommunications Networks," *IEE Proceedings-Communications*, vol. 144, no. 5, pp. 322-330, Oct. 1997.

[4] E. Modiano and A. Narula, "Survivable Lightpath Routing: A New Approach to the Design of WDM-Based Networks," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 4, pp. 800-809, May 2002.

[5] D. Zhou and S. Subramaniam, "Survivability in Optical Networks," *IEEE Network* , vol. 14, no. 6, pp. 16-23, Nov/Dec. 2000.

[6] A. Phillips, J. Senior, R. Mercinelli, M. Valvo, P. Vetter, C. Martin, M. Deventer, P. Vaes, and X. Qiu, "Redundancy Strategies for a High Splitting Optically Amplified Passive Optical Network," *IEEE Journal of Lightwave Technology*, vol. 19, no. 2, pp. 137-149, Feb. 2001.

[7] G. Sasaki and C. Su, "The Interface between IP and WDM and its Effect on the Cost of Survivability," *IEEE Communications Magazine*, vol. 41, no. 1 , pp. 74-79, Jan. 2003.

[8] A. Orda and R. Rom, "Multihoming in Computer Networks: a Topology-design Approach," *Computer Networks ISDN*, vol. 18, no. 2, pp. 133-141, Feb. 28, 1990.

[9] J. W. Suurballe, "Disjoint Paths in a Network,"*Networks*, vol. 4, pp. 125-145, 1974.