Dual-Homing Protection in IP-Over-WDM Networks

Jianping Wang, *Member, IEEE*, Vinod M. Vokkarane, *Member, IEEE*, Raja Jothi, Xiangtong Qi, Balaji Raghavachari, and Jason P. Jue, *Senior Member, IEEE*

***Abstract*—A fault-tolerant scheme, called dual homing, is generally used in IP-based access networks to increase the availability of the networks. In a dual-homing architecture, 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. This dual-homing architecture in the access network imposes the overhead to provide protection in the core network. Scaling the next-generation IP-over-wavelength-division-multiplexing (WDM) Internet, and being able to support a growing number of such dual-homing connections, as well as protection, demands a scalable mechanism to contain this overhead for protection in the WDM networks. This paper studies the coordinated protection design to reduce the protection cost in the WDM core network, given a dual-homing infrastructure in the access network. The protection problem is considered for both static and dynamic traffic. Several solutions are proposed, and the performances of the solutions are compared. We also prove that one of the proposed algorithms gives a solution that, in the worst case, is at most 4/3 times the cost of the optimal solution.**

***Index Terms*—Dual homing, lightpath, protection, wavelength division multiplexing (WDM).**

I. Introduction

IP-OVER-WDM (wavelength-division-multiplexing) networks are considered to be a major component 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.

In WDM networks, survivability is usually provided to handle single link failures in the core network. A single failure in an optical fiber can dramatically degrade network performance, since a single fiber can carry a large amount of traffic. Therefore, it is critical to support network survivability in WDM networks. Survivability in WDM networks is implemented by using protection and restoration techniques, which provide the survivability by setting up two disjoint lightpaths between the source and the destination. One lightpath is called the primary lightpath, and the other is called the backup lightpath. Protection is a static mechanism, which reserves resources for both primary and backup lightpaths prior to data communication. Restoration, on the other hand, is a dynamic mechanism, where the backup lightpath is not set up until the failure occurs. Existing literature on protection and restoration in WDM networks can be found in [1] and [2].

Dual homing is generally used to increase survivability in the access network. 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 [3]–[7] and wireless networks [8].

As the Internet advances, an enterprise may wish to acquire its Internet connectivity from two Internet service providers (ISPs) for enhanced reliability. Such dually connected enterprises are referred to as being "dual-homed." When connectivity through one of the ISPs fails, connectivity via the other ISP enables the enterprise to preserve its connectivity to the Internet. The enhanced availability, combined with the decreasing cost of Internet connectivity, motivates more and more enterprises to become dual homed. Each enterprise can select two among the available ISPs; the selection criteria, however, is out of the scope of this paper.

Given dual-homing architecture, the routing overhead imposed on the Internet routing system becomes more and more significant if a route has to be maintained from each home to the destination. Scaling the Internet and being able to support a growing number of such enterprises demands scalable mechanism(s) to contain this overhead. Routing and addressing strategies that could reduce the routing overhead due to multihomed enterprises connected to multiple ISPs in the Internet routing system have been addressed in [9] and implemented in Cisco's network-address-translation (NAT) servers.

In an IP-over-WDM dual-homing architecture, a host in the access network is attached 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.

There have been several efforts on providing survivability for a dual-homed IP-based access network over WDM-based core networks [10], [11]. In all this literature, the authors consider providing survivability separately at the IP layer, as well as at the WDM layer. In [10], the authors discuss how to support dual homing in passive optical networks, while [11] studies survivability in IP-over-WDM networks and provides different

Manuscript received December 14, 2004; revised July 19, 2005. This research is partially supported by National Science Foundation (NSF) Grants 0435095 and 0435105.

J. Wang is with the Department of Computer Science, Georgia Southern University, Statesboro, GA 30460 USA (e-mail: jpwang@georgiasouthern.edu).

V. M. Vokkarane is with the Department of Computer and Information Science, University of Massachusetts Dartmouth, North Dartmouth, MA 02747 USA (e-mail: vvokkarane@umassd.edu).

R. Jothi is with the National Center for Biotechnology Information, National Institutes of Health (NIH), Bethesda, MD 20894 USA (e-mail: jothi@ncbi.nlm.nih.gov).

X. Qi is with the Department of Industrial Engineering and Engineering Management, Hong Kong University of Science and Technology, Hong Kong (e-mail: ieemqi@ust.hk).

B. Raghavachari and J. P. Jue are with the Department of Computer Science, The University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: rbk@utdallas.edu; jjue@utdallas.edu).

Digital Object Identifier 10.1109/JLT.2005.856321

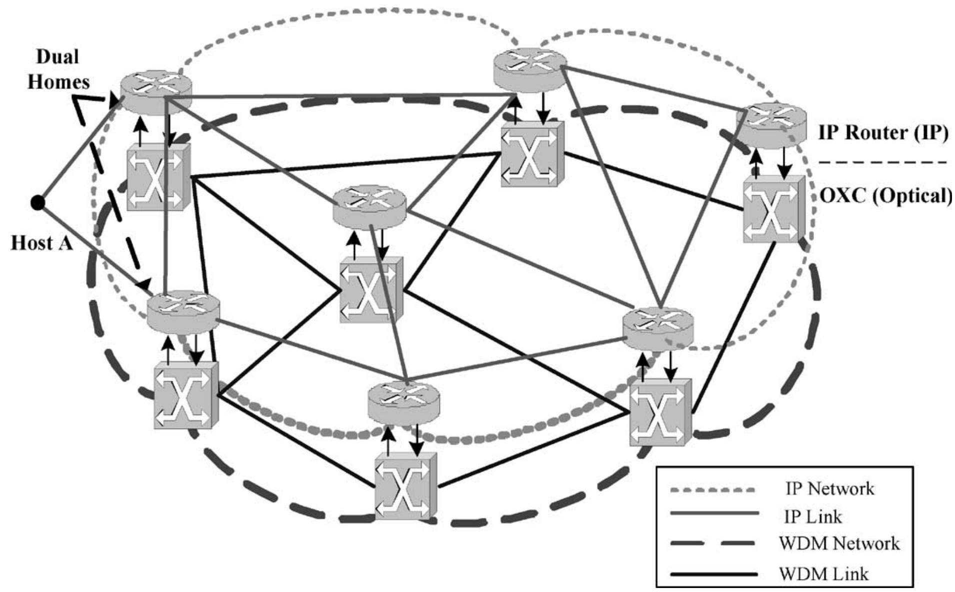

Fig. 1. Dual-homing architecture.

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 present a coordinated solution for providing protection in an IP-over-WDM dual-homing network.

Similar to the observation in [9], given the dual-homing architecture in the access network, it will impose the protection overhead in the core network if protection is provided. There are two possible approaches to provide protection in the core network given the dual-homing architecture in the access network, independent protection and coordinated protection. For independent protection, a host sends an independent request to each of its two dual-homed access routers, while each router attempts to set up two disjoint lightpaths independently, to provide protection against a single failure in the core network. Since each IP router is not aware of the existence of the other home, those two pairs of disjoint paths may not be able to make an effort to share resources. Therefore, it may incur a higher cost. For coordinated protection, a host sends a request to each of its dual homes (access routers) by including the information of the other home. With this additional information, each home calculates two pairs of disjoint paths that maximize the sharing between those two pairs of disjoint paths. Since both homes have the same request information from the host, the independently computed routes at each of the homes will be identical. This paper focuses on developing algorithms to calculate two pairs of disjoint paths with minimum cost. We briefly explain the motivation of coordinated protection as follows. 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 each of the dual homes to the destination. By providing a coordinated solution, we can obtain significant cost benefits, as compared to independent protection.

In this paper, we study the coordinated protection from dual homes to the destination to minimize the protection cost given the dual-homing architecture in the access network, called dual-homing protection (DHP). In this paper, DHP is studied subject to both static and dynamic traffic. The rest of the paper is organized as follows. The network architecture of DHP is described in Section II. The detailed problem description is presented in Section III. An integer linear-programming (ILP) model is given as a solution to the static DHP problem in Section IV. In Section V, we propose a number of different heuristics to solve the dynamic DHP problem. We also prove that one of the proposed algorithms gives a solution that, in the worst case, is at most 4/3 times the cost of the optimal solution. In Section VI, we evaluate the performance of all the proposed algorithms. Finally, the conclusion is presented in Section VII.

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 attached to two IP routers in the IP-based access network. Each IP router is connected to an OXC, which in turn is linked to other OXCs that constitute the all-optical WDM core network. In a dual-homing architecture, two link-disjoint paths connect the host to its dual homes, which provides survivability against a single IP router (node) or access-link failure. The dual-homed IP routers are connected to the underlying OXCs, which 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, 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 of an access-link failure, the access network is survivable with the dual-homing infrastructure.

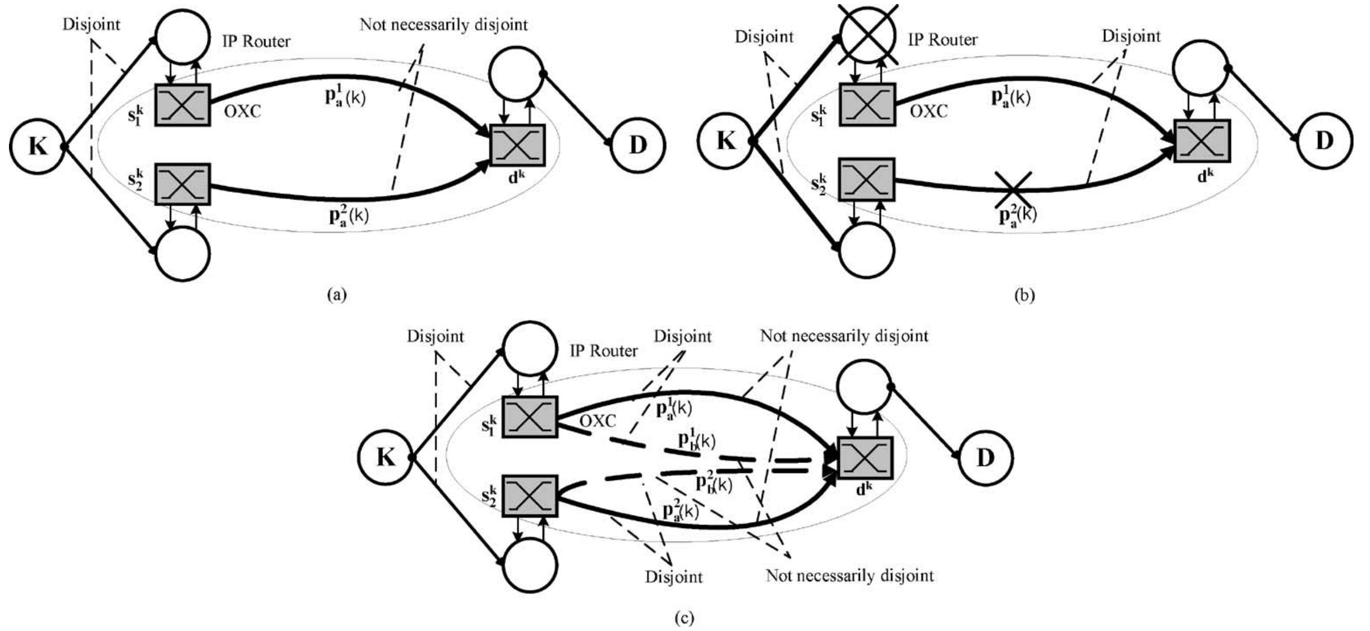

Fig. 2. Dual-homing and protection architectures. (a) Unprotected dual-homing architecture. (b) Unprotected dual-homing architecture with failures. (c) Protected dual-homing architecture.

Hence, dual homing provides survivability against a 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. Therefore, the DHP we study in this paper can provide survivability subject to one link/node failure in the access network, as well as one link failure in the core network simultaneously.

In our model, we assume that when the IP router fails, the OXC connected to the router continues to carry optical traffic from other OXCs in the core network. This assumption is reasonable since WDM layer is a separate layer, and switching functions are provided in the WDM layer.

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. As a matter of fact, we find that having the primary paths share the maximum number of links reduces the amount of resources reserved, which is one of the primary objectives in this paper. On the other hand, the disjointness constraint between the primary paths and the backup paths has to be satisfied. 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 unidirected graph $G\langle V, E\rangle$, where V is the set of OXCs and E is the set of WDM links. Let $N = |V|$ be the number of nodes in the graph. Let the wavelength cost of a WDM link $e \in E$ be $c(e)$, and the maximum number of wavelengths in each link be W. Let R denote the set of connection requests. Suppose that each individual connection request, denoted by k, is given by $\{\{s_1^k, s_2^k\}, d_k\}$, where s_1^k and s_2^k are two OXCs connected to the dual-homed access routers of request k, and d_k is the destination OXC, which in turn is connected to an IP router that connects to the destination host of request k. In this paper, we assume that the two dual homes s_1^k and s_2^k are given, and the problem we study is finding a pair of disjoint paths from s_1^k to d_k, as well as a pair of disjoint paths from s_2^k to d_k, which can provide protection against a single link failure from the access network and a single link failure from the core network simultaneously. Another related problem is the determination of two homes for each host, which will not be discussed in this paper. For interested readers, we refer to [12] and [13] for different models.

Let the primary lightpath from s_1^k to d_k be denoted by $p_a^1(k)$ and the link-disjoint backup lightpath from s_1^k to d_k be denoted by $p_b^1(k)$. Similarly, the primary lightpath from s_2^k to d_k is denoted by $p_a^2(k)$ and the link-disjoint backup lightpath from s_2^k to d_k is denoted by $p_b^2(k)$. Let L_k be the set of all links used in the primary and backup lightpaths for the connection request k. L_k is given by $p_a^1(k) \cup p_b^1(k) \cup p_a^2(k) \cup p_b^2(k)$.

If the core network is reliable, $p_a^1(k)$ and $p_a^2(k)$ are not necessarily disjoint, as shown in Fig. 2(a). However, even if $p_a^1(k)$ and $p_a^2(k)$ 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^k is down and one link in $p_a^2(k)$ is also down, data cannot be sent to d_k. In order to provide dual-homing protected service, we need $p_b^1(k)$ and $p_b^2(k)$ to protect the lightpaths $p_a^1(k)$ and $p_a^2(k)$. We have the following observations with respect to resource sharing:

— $p_a^1(k)$ and $p_a^2(k)$ are not necessarily disjoint.
— $p_b^1(k)$ and $p_b^2(k)$ are not necessarily disjoint.
— $p_a^1(k)$ and $p_b^2(k)$ are not necessarily disjoint.
— $p_a^2(k)$ and $p_b^1(k)$ are not necessarily disjoint.
— $p_a^1(k)$ and $p_b^1(k)$ must be disjoint.
— $p_a^2(k)$ and $p_b^2(k)$ must be disjoint.

Fig. 2(c) illustrates these observations. These observations provide guidance in reducing the protection cost in the core network given the dual-homing architecture in the access network. Our effort in this study focuses on fully investigating the sharing properties among these two pairs of disjoint paths. It is also observed that each request will use no more than one wavelength on any link.

In this paper, we study two traffic models for protection in WDM mesh networks with dual-homing survivability. One problem is to route $p_a^1(k)$, $p_b^1(k)$, $p_a^2(k)$, and $p_b^2(k)$ for all requests in R simultaneously, which is called static DHP. The other problem is to route $p_a^1(k)$, $p_b^1(k)$, $p_a^2(k)$, and $p_b^2(k)$ when request k arrives, which is called dynamic DHP. The difference is that, for the former case, we need to consider all the requests at once, and for the latter case, we only need to consider the newly arriving request.

We assume 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. Therefore, our objective in static DHP is to find L^k for each request k in R, such that the total cost C is minimum, where

$$C = \sum_{k \in R} \sum_{e \in L^k} c(e). \tag{1}$$

The objective of dynamic DHP is to find L^k for request k, such that $\sum_{e \in L^k} c(e)$ is minimum.

IV. Static DHP

Following the spirits of [14], we develop an ILP formulation for the static DHP problem. We have the following notation:

$x_a^1(k,e)$	1 if path $p_a^1(k)$ uses link e, 0 otherwise.
$x_b^1(k,e)$	1 if path $p_b^1(k)$ uses link e, 0 otherwise.
$x_a^2(k,e)$	1 if path $p_a^2(k)$ uses link e, 0 otherwise.
$x_b^2(k,e)$	1 if path $p_b^2(k)$ uses link e, 0 otherwise.
y_e^k	1 if any path for request k uses link e, 0 otherwise.
y_e	total number of wavelengths used in link e.
$\mathrm{In}(v)$	set of links that end at node v.
$\mathrm{Out}(v)$	set of links that start from node v.

The objective is to minimize

$$\sum_{e \in E} y_e c(e) \tag{2}$$

subject to

$$\sum_{e \in \mathrm{Out}(s_1^k)} x_a^1(k,e) = 1 \qquad \forall k \tag{3}$$

$$\sum_{e \in \mathrm{In}(d^k)} x_a^1(k,e) = 1 \qquad \forall k \tag{4}$$

$$\sum_{e \in \mathrm{Out}(s_1^k)} x_b^1(k,e) = 1 \qquad \forall k \tag{5}$$

$$\sum_{e \in \mathrm{In}(d^k)} x_b^1(k,e) = 1 \qquad \forall k \tag{6}$$

$$\sum_{e \in \mathrm{Out}(s_2^k)} x_a^2(k,e) = 1 \qquad \forall k \tag{7}$$

$$\sum_{e \in \mathrm{In}(d^k)} x_a^2(k,e) = 1 \qquad \forall k \tag{8}$$

$$\sum_{e \in \mathrm{Out}(s_2^k)} x_b^2(k,e) = 1 \qquad \forall k \tag{9}$$

$$\sum_{e \in \mathrm{In}(d^k)} x_b^2(k,e) = 1 \qquad \forall k \tag{10}$$

$$\sum_{e \in \mathrm{In}(v)} x_a^1(k,e) = \sum_{e \in \mathrm{Out}(v)} x_a^1(k,e) \qquad \forall k, v, v \neq s_1^k \tag{11}$$

$$\sum_{e \in \mathrm{In}(v)} x_b^1(k,e) = \sum_{e \in \mathrm{Out}(v)} x_b^1(k,e) \qquad \forall k, v,\ v \neq s_1^k \tag{12}$$

$$\sum_{e \in \mathrm{In}(v)} x_a^2(k,e) = \sum_{e \in \mathrm{Out}(v)} x_a^2(k,e) \qquad \forall k, v, v \neq s_2^k \tag{13}$$

$$\sum_{e \in \mathrm{In}(v)} x_b^2(k,e) = \sum_{e \in \mathrm{Out}(v)} x_b^2(k,e) \qquad \forall k, v, v \neq s_2^k \tag{14}$$

$$x_a^1(k,e) + x_b^1(k,e) \leq 1 \qquad \forall k, e \tag{15}$$

$$x_a^2(k,e) + x_b^2(k,e) \leq 1 \qquad \forall k, e \tag{16}$$

$$y_e^k \geq \frac{1}{4}\left(x_a^1(k,e) + x_b^1(k,e) + x_a^2(k,e) + x_b^2(k,e)\right) \qquad \forall k, e \tag{17}$$

$$y_e = \sum_k y_e^k \qquad \forall e \tag{18}$$

$$y_e \leq W \qquad \forall e \tag{19}$$

$$y_e^k \in \{0,1\} \qquad \forall k, e \tag{20}$$

$$x_a^1(k,e) \in \{0,1\} \qquad \forall k, e \tag{21}$$

$$x_b^1(k,e) \in \{0,1\} \qquad \forall k, e \tag{22}$$

$$x_a^2(k,e) \in \{0,1\} \qquad \forall k, e \tag{23}$$

$$x_b^2(k,e) \in \{0,1\} \qquad \forall k, e. \tag{24}$$

Constraints (3)–(14) are the network-flow conservation constraints. Constraint (15) forces $p_a^1(k)$ and $p_b^1(k)$ to be disjoint, and constraint (16) forces $p_a^2(k)$ and $p_b^2(k)$ to be disjoint. Constraint (17) indicates that no more than one wavelength is reserved in any link e for a request r_k. Constraints (18) and (19) indicate the maximum requests a link can support. Constraints (20)–(24) indicate the integer constraint on the variables.

We analyze the size of the abovementioned ILP models as follows. Let $|E|$ be the number of links in the network, $|V|$ the number of nodes, and $|R|$ the number of connections. Then we can see that the total number of decision variables in the model is $5|K||E| + |E|$, and the number of constraints is $4|K||V| + 3|K||E| + 2|E|$. The ILP model can be used to find the optimal solution to a problem, though it may take a long time to find a solution when the problem size becomes large. However, the static routing model is usually regarded as a planning problem that will not be solved very frequently. For these planning problems, finding a high-quality solution, i.e., the global optimum, is more important than obtaining a solution quickly. Thus, an ILP model is the correct choice for approaching the static routing model. On the other hand, the dynamic routing model studied below is usually regarded as a real-time scheduling problem, where a solution is needed immediately when a connection request is received. Thus, quick heuristic algorithms need to be developed to handle the dynamic routing problem.

To evaluate the cost of static DHP, we compare the DHP model with four other models, namely, single-homing without protection (SH), single-homing with protection (SHP), dual-homing without protection (DH), and independent DHP (IDHP). The main difference among SH, SHP, and DH models is that they only need one or two paths. The difference between the IDHP solution and the DHP solution is that the IDHP solution is not aware of the existence of the other home; therefore, sharing of edges in the optical layer is not considered. The four other models can be formulated and solved by slightly modifying the developed DHP ILP model. We can modify the flow-conservation constraints to indicate the requirements for the paths of p_a^1, p_a^2, p_b^1, and p_b^2 in each specific model.

For the SH model, we set the right-hand side of (5)–(10) to 0, indicating that only one path p_a^1 is needed. In this case, no protection is provided against any failures.

For the SHP model, we set the right-hand side of (7)–(10) to 0, indicating that two disjoint paths p_a^1 and p_b^1 are needed. In this case, protection is provided against a single link failure of the core network, but not against any single failure of the access network.

For the DH model, we can set the right-hand side of (5), (6), (9) and (10) to 0, indicating that two disjoint paths p_a^1 and p_a^2 are needed. In this case, protection is provided against a single failure of the access network, but not against any single failure of the core network.

For the IDHP model, variable y_e^k is redefined as a nonnegative integer variable for the wavelength usage of link e by request k, and (17) is changed to

$$y_e^k = x_a^1(k,e) + x_b^1(k,e) + x_a^2(k,e) + x_b^2(k,e) \qquad \forall k, e.$$

In this case, protection is provided against a single failure of the access network and a single failure of the core network simultaneously, the same effect as our coordinated DHP solution, but with higher cost.

V. Dynamic DHP Algorithms

We now propose several heuristics for dynamic DHP. These heuristics can be classified into two categories: One category is based on a minimum-cost network-flow (MCNF) model and the other category is based on a minimum Steiner-tree model. The MCNF model computes minimum-cost link-disjoint paths that satisfy the disjointness between the primary path and the backup path [15]. On the other hand, the minimum Steiner-tree model considers the sharing among the primary paths and the sharing among the backup paths.

Since we only consider the current arrival request, for simplicity, we will omit the index variable k from the previous notations. Instead, let s_1, s_2, and d be the first home router, the second home router, and the destination router of the current request, respectively. Correspondingly, let p_a^1 be the primary lightpath from s_1 to d, p_a^2 be the primary lightpath from s_2 to d, p_b^1 be the backup lightpath from s_1 to d, and p_b^2 be the backup lightpath from s_2 to d.

The first heuristic is based on MCNFs. The heuristic 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 approach to obtain the solution by this heuristic algorithm is illustrated in Fig. 3(a).

The second heuristic is also based on MCNFs and is a generalization of the first heuristic, in which we first select a new 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 the second heuristic, in which the destination is chosen as the branching node. Fig. 3(b) illustrates the steps to obtain the solution by this heuristic algorithm.

The third heuristic algorithm is also based on the MCNF model and is motivated by the fact that the two dual homes are usually located close to each other. Here, we find the shortest link-disjoint paths from each of the dual homes to the destination, and also two minimum-cost link-disjoint paths between the dual homes. These four paths make up the primary and backup lightpaths. The solution obtained by this heuristic algorithm is illustrated in Fig. 3(c).

The last heuristic algorithm is based on the minimum Steiner tree. The algorithm finds a low-cost Steiner tree that connects the two homes to the destination; the primary paths are covered by the minimum Steiner tree. The algorithm then provides path protection from each home to the destination. This approach explicitly exploits the sharing of links between the primary lightpaths and is demonstrated in Fig. 3(d).

We now describe each of the algorithms in detail and compare their relative performance.

A. Minimum-Cost Network-Flow Heuristic (MCNFH)

The MCNFH first finds the minimum-cost link-disjoint primary and backup lightpaths from one of the dual homes to the destination, then changes the cost of the these links to zero (in order to encourage sharing), and finds the minimum-cost link-disjoint primary and backup lightpaths from the other dual home to the destination.

We can use the MCNF algorithm to find the minimum-cost link-disjoint primary and backup lightpaths from one home to the destination. Initially, we set the capacity of the link to be unity, in order to force the primary and backup lightpaths from s_1 to d, as well as from s_2 to d, to be disjoint. 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

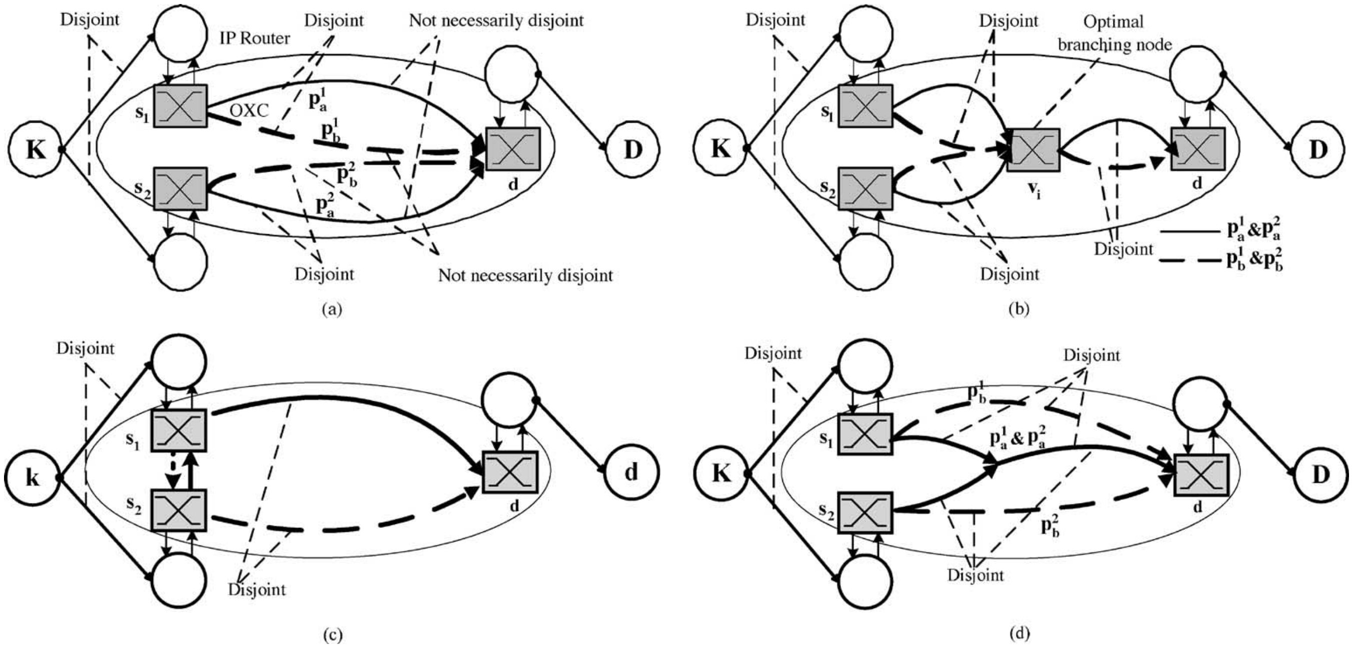

Fig. 3. Dynamic DHP using different heuristics. (a) Protected dual-homing architecture using MCNFH. (b) Protected dual-homing architecture using MDSPH. (c) Protected dual-homing architecture using MCSPH. (d) Protected dual-homing architecture using MSTH.

one dual home to the destination, and then find the primary and backup lightpaths from the other dual home to the same destination. Then, we exchange the order and repeat the same process. Finally, we select the solution having the minimum cost.

The detailed algorithm is given in Fig. 4. In Fig. 4, 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.

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 either exists 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 or the cost of OPT is at least 3/4 times the cost of the solution obtained from MCNFH.

Proof: In any optimal 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) Paths p_a^1 and p_a^2 are not edge disjoint.
2) Paths p_b^1 and p_b^2 are not edge disjoint.
3) Paths p_a^1 and p_b^2 are not edge disjoint.
4) 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 six cases, we now show (not necessarily in the same order) how to find u and v such that the lemma is true.

Case 5: Initially, set u and v to d. Obviously, p_a^1 and p_a^2 meet at u, 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 subcases of case 5, and cases 3 and 4 are subcases 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 7–11, 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 discussed 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 problem P_1, which asks for two edge-disjoint paths from each of 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 from s_1 to d, two edge-disjoint paths from 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 from s_2 to d and two edge-disjoint paths from s_1 to s_2. Using the edges in OPT, we can easily construct F_1, F_2, and

$C_1(s_1, d) = 0$; $S_1 = \emptyset$;
Call MCNF(s_1, d) to find p_a^1 and p_b^1
for ($e \in p_a^1 \cup p_b^1$) {
    $C_1(s_1, d) += e$; $S_1 = S_1 \cup \{e\}$;
    $BackupCost(e) = c(e)$; $c(e) = 0$;
}
$C_2(s_2, d) = 0$; $S_2 = \emptyset$;
Call MCNF(s_2, d) to find p_a^2 and p_b^2
for ($e \in p_a^2 \cup p_b^2$) {
    $C_1(s_2, d) += e$; $BackupCost(e) = c(e)$;
}
$C = C_1(s_1, d) + C_1(s_2, d)$; $S_1 = S_1 \cup p_a^2 \cup p_b^2$;
for ($e \in S_1$)
    $c(e) = BackupCost(e)$;
$C_2(s_2, d) = 0$; $S_2 = \emptyset$; $C_2(s_1, d) = 0$;
Call MCNF(s_2, d) algorithm to find p_a^2 and p_b^2
for ($e \in p_a^2 \cup p_b^2$) {
    $C_2(s_2, d) += e$; $S_2 = S_1 \cup \{e\}$;
    $backup(e) = c(e)$; $c(e) = 0$;
}
Call MCNF(s_1, d) to find p_a^1 and p_b^1
for ($e \in p_a^1 \cup p_b^1$)
    $C_2(s_1, d) += e$;
$S_2 = S_2 \cup p_a^1 \cup p_b^1$;
if ($C > C_2(s_1, d) + C_2(s_2, d)$) {
    $C = C_2(s_1, d) + C_2(s_2, d)$;
    $S = S_2$;
} else
    $S = S_1$;

Fig. 4. MCNFH algorithm description.

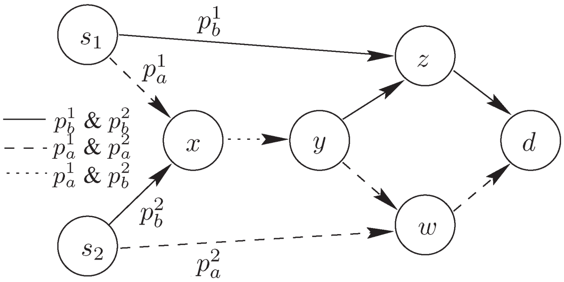

Fig. 5. p_a^1 shares edges with p_b^2 even before p_b^1 meets p_b^2 or p_a^1 meets p_a^2.

F_3 by finding two edge-disjoint paths from each of the two source nodes to the destination nodes, as shown in Fig. 6(a)–(c).

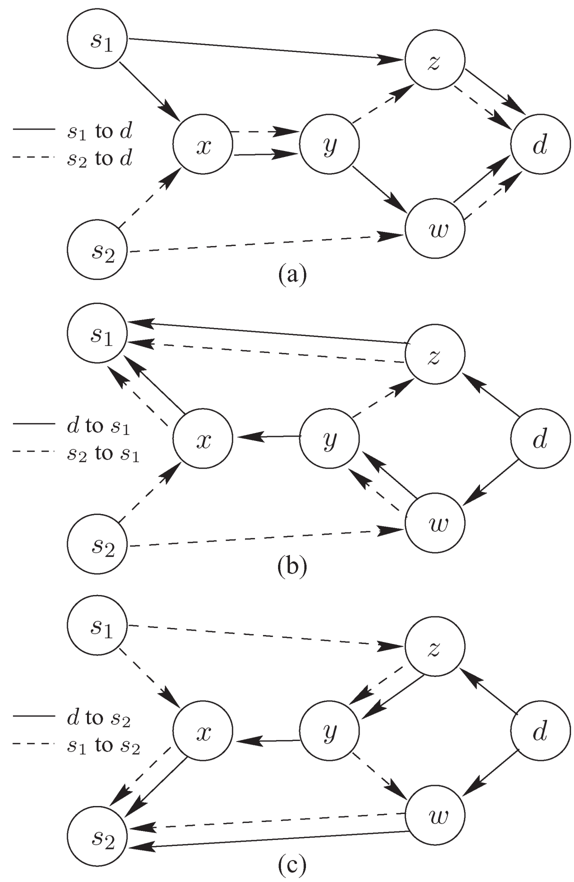

Fig. 6. (a) F_1 constructed by two edge-disjoint paths from each of s_1 and s_2 to t. (b) F_2 constructed by two edge-disjoint paths from each of t and s_1 to s_2. (c) F_3 constructed by two edge-disjoint paths from each of t and s_2 to s_1.

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 four times the cost of OPT, as each edge in OPT appears four times F_1, F_2, and F_3 overall. 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

$$3f_m \leq f_1 + f_2 + f_3 \leq F_1 + F_2 + F_3 = 4 * \text{OPT}.$$

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 is 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_a^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.

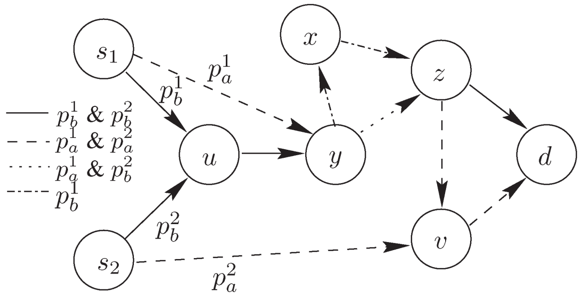

Fig. 7. p_a^1 shares edges with p_b^2 after p_b^1 meets p_b^2.

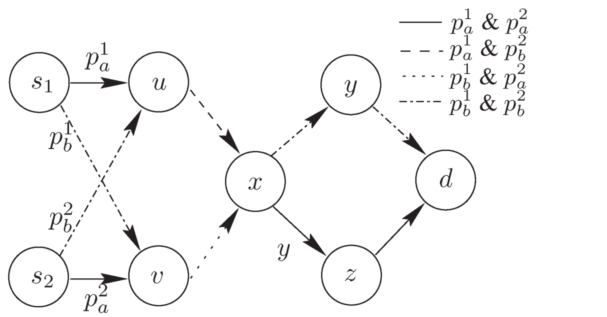

Fig. 8. p_a^1 shares edges with p_a^2 and p_b^2, and p_b^1 shares edges with p_a^2 and p_b^2.

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 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. ■

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. And 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 nonexistent. In such a scenario, we interpret paths

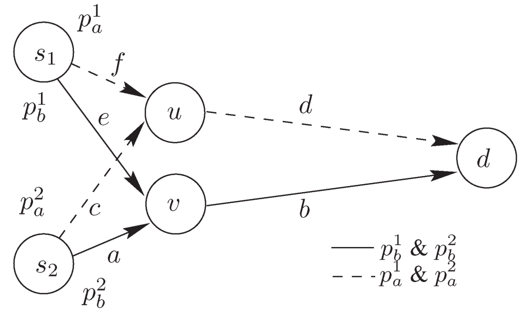

Fig. 9. Canonical form.

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. This is similar to paths p_b^1 and p_b^2 sharing a path from v to d. Thus, there exists 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 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 that asks for two edge-disjoint paths from each of s_1 and s_2 to d, f_2 is a feasible solution for the problem that asks for two edge-disjoint paths from each of s_1 to d and s_2 to s_1, and f_3 is a feasible solution for the problem that asks for two edge-disjoint paths from each of 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,

$$\text{Cost of } f_1 \le a+b+c+d+e+f+b+d$$
$$\text{Cost of } f_2 \le a+b+c+d+e+f+a+c$$
$$\text{Cost of } f_3 \le a+b+c+d+e+f+e+f.$$

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

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

■

B. Minimal Disjoint Segment-Pair Heuristic (MDSPH)

The MDSPH is based on the observation that the two primary paths are either disjoint or there is a branching node that

```
C = ∞
for(v_i ∈ V) {
    C_i = 0;
    Call MCNF(s_1, v_i) to find two link-disjoint paths p_a^1 and p_b^1;
    L_1(v_i) = p_a^1 ∪ p_b^1; S_i = L_1(v_i);
    for (e ∈ L_1(v_i)) {
        C_i+ = c(e); BackCost(e) = c(e); c(e) = 0;
    }
    Call MCNF(s_2, v_i) to find two link-disjoint paths p_a^2 and p_b^2;
    L_2(v_i) = p_a^2 ∪ p_b^2; S_i = S_i ∪ L_2(v_i);
    for (e ∈ L_2(v_i)) {
        C_i+ = c(e); BackupCost(e) = c(e); c(e) = 0;
    };
    Call MCNF(v_i, d) to find two link-disjoint paths p_a^3 and p_b^3;
    L_3(v_i) = p_a^3 ∪ p_b^3; S_i = S_i ∪ L_3(v_i);
    for (e ∈ L_3(v_i)) {
        C_i+ = c(e); BackupCost(e) = c(e); c(e) = 0;
    };      if (C > C_i) {
        C = C_i; v_b = V_i;
    }
    for (e ∈ S_i)
    c(e) = BackupCost(e);
}
```

Fig. 10. Description of the MDSPH algorithm.

connects the two homes and the destination. As a matter of fact, if two primary paths are disjoint, it can still be considered as if there is a branching node that is the destination. Obviously, the position of the branching node will effect the total cost of the primary and backup lightpaths.

The MDSPH tries to find the right branching node, such that the total wavelength cost used in both the primary 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. MDSPH makes efforts on finding v_b, such that $S_b = \min_{v_i \in V}\{S_i\}$. MDSPH works as shown in Fig. 10.

In the algorithm described in Fig. 10, 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 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.

Here, we give an example to show that MDSPH can find a better solution than MCNFH. In Fig. 11, for MCNFH, when MCNF is called to find two disjoint paths from s_1 to d, there are three feasible solutions that give a cost of 8. If we choose paths $s_1-1-2-3-d$ and $s_2-4-5-6-d$, when MCNF is called to find two disjoint paths from s_2 to d, there are three feasible solutions that give a cost of 8. Therefore, the total cost is 16. However, MDSPH will choose paths $s_1-7-9-10-d$, $s_1-8-9-10-d$, $s_2-12-9-10-d$, and $s_2-13-9-10-d$, for which the total cost is 12. The difference between MDSPH and MCNFH in this example is that it is hard for MCNFH to make the right decision when there are several feasible solutions, since it finds the two link-disjoint paths from one home to the destination, then from the other home to the destination.

C. Minimum-Cost Shortest Path Heuristic (MCSPH)

In the MCSPH, we obtain link-disjoint shortest paths from the dual homes to the destination, and then compute two link-disjoint paths with minimum cost between the dual homes themselves. The details of the algorithms are as found in Fig. 12.

The solution obtained is composed of two minimum-cost link-disjoint primary paths from the dual homes OXCs to the destination, p_a^1 and p_a^2. The backup path for the first home is composed of the path from the first home to the second home and the path from the second home to the destination. The backup path for the second home is composed of the path from the second home to the first home and the path from the first home to the destination. Since the backup paths from a dual home to the destination go through the other dual home, in the case of an access-node failure, we assume that the underlying OXC can continue to forward all-optical traffic seamlessly.

D. Minimum Steiner-Tree Heuristic (MSTH)

The MSTH uses the fact that the minimum Steiner tree is the best approach to connect three nodes with minimum cost. The idea behind MSTH is to find a minimum-cost tree that is designated as the primary tree and then 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 terminal nodes. We observe that a tree with only three terminal 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 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 T_i, which consists of the shortest paths from s_1 to v_i, from s_2 to v_i, and from v_i to d, resulting in N different trees. The optimal Steiner tree T_{opt} is given by the minimum-cost tree of the N different enumerated trees. Two primary lightpaths are provided in T_{opt}. Then, a link-disjoint backup lightpath is constructed from each source to the destination. The algorithm description is given in Fig. 13.

E. Heuristic Algorithms Comparison

Let us consider the network topology given in Fig. 14(a). We assume unit-link cost for all links. The primary and backup

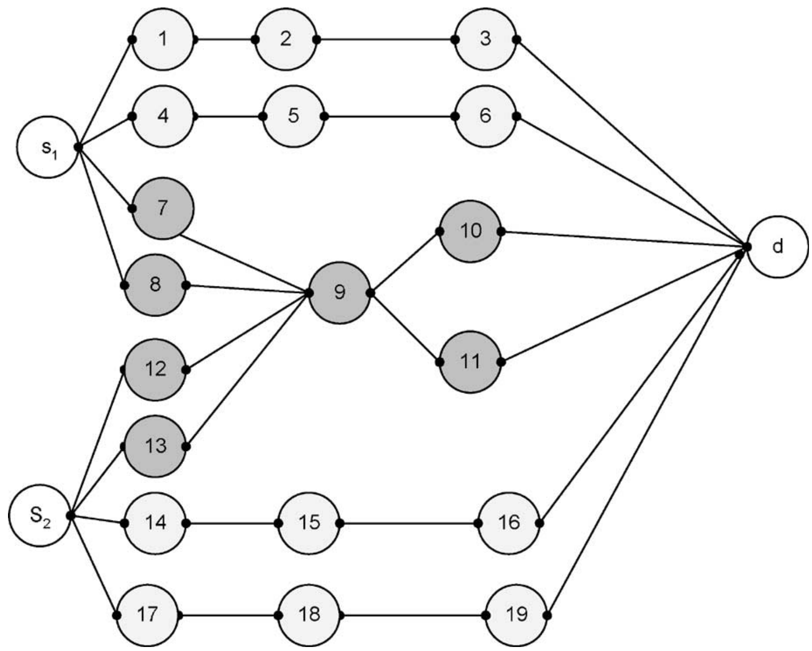

Fig. 11. One example to show MDSPH can find a better solution than MCNFH.

$C_1 = 0; C_2 = 0;$

Call Dijkstra(s_1, d) to find shortest path p_a^1;

for ($e \in p_a^1$) {

$C_1 += c(e); c(e) = \infty;$

}

Call Dijkstra(s_2, d) to find shortest path p_a^2;

for ($e \in p_a^2$) {

$C_2 += c(e); c(e) = \infty;$

}

Call Dijkstra(s_1, s_2) to find shortest path p_b^1

for ($e \in p_b^1$) {

$C_2 += c(e); c(e) = \infty;$

}

$p_b^1 = p_b^1 \cup p_a^2;$

Call Dijkstra(s_2, s_1) to find shortest path p_b^2

for ($e \in p_b^2$) {

$C_1 += c(e); c(e) = \infty;$

}

$p_b^2 = p_b^2 \cup p_a^1;$

Fig. 12. MCSPH algorithm description.

lightpaths found by MCNFH are given in Fig. 14(b) and have a cost of 12. The primary and backup lightpaths found by MDSPH are also given in Fig. 14(b) and have a cost of 12. The primary and backup lightpaths found by MCSPH are given in Fig. 14(c) and have a cost of 10. The primary and backup lightpaths found by MSTH are given in Fig. 14(d) and have a cost of 11. We see that for the given network topology, the MSTH performs the best.

We observe that MCNFH and MDSPH can always find a solution, if one exists, since finding a disjoint pair of paths from one home router to the destination does not interfere with the choice of the disjoint pair of paths from the other home router to the destination. However, MSTH may not be able to find such a feasible solution, even if there is such a solution, since there may not be link-disjoint backup lightpaths (or a disjoint tree) after the primary lightpaths (or tree) are computed. For MCSPH, it is also possible that the algorithm cannot find the feasible solution, even if such a solution exists, since there may not be link-disjoint paths between the dual homes.

In Table I, we compare the time complexities of the proposed DHP heuristics. We see that MCNFH and MCSPH have 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 $O(N^3)$.

VI. Simulation Results

In this section, we analyze the performance of proposed algorithms for DHP. For static DHP, the DHP service provides the customers with a very high level of survivability at the additional cost of consuming more wavelengths than other less-reliable services. Therefore, we are interested in the cost increase caused by the DHP service. For the dynamic DHP model, we are interested in comparing the performance of MCNFH, MDSPH, MCSPH, and MSTH algorithms. In this section, we first analyze the simulation results for static DHP, then discuss the simulation results for dynamic DHP heuristic algorithms. For both the simulation for the static and dynamic DHPs, the network topologies are randomly generated as unidirectional graphics given network size N and the maximum outgoing degree D.

```
C_min = ∞; C = 0;
for (v_i ∈ V) {
    Call Dijkstra(s_1, v_i) to find shortest path X^1;
    for (e ∈ X^1)
        C+=c(e);
    Call Dijkstra(s_2, v_i) to find shortest path X^2;
    for (e ∈ X^2)
        C+=c(e);
    Call Dijkstra(v_i, d) to find shortest path X^3;
    for (e ∈ X^3)
        C+=c(e);
    if (C < C_min) {
        p_a^1 = X^1 ∪ X^3;
        p_a^2 = X^2 ∪ X^3;
    }
}
for(e ∈ p_a^1)
    c(e) = ∞;
Call Dijkstra(s_1, d) to find shortest path p_b^1;
for(e ∈ p_a^1 − p_a^2)
    c(e) = 0;
for(e ∈ p_b^1 − p_a^2)
    c(e) = 0;
for(e ∈ p_a^2)
    c(e) = ∞;
Call Dijkstra(s_2, d) to find shortest path p_b^2;
```

Fig. 13. Description of the MSTH algorithm.

A. Static DHP

We compare the DHP model with four other models, SH, SHP, DH, and IDHP. For a given instance of the problem with randomly generated network topology and a set of connection requests, we compare the cost for each of the five different types of service categories. All problems are solved by CPLEX.

The important simulation parameters in generating problem instances include the network size N and the maximum outgoing degree at each node D. Given a group of parameters N and 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.

In the first experiment, we test the cost involving the number of connections. We set the number of nodes in the network, $N = 50$, the maximum outgoing degree of a node, $D = 20$, the number of wavelengths on every link, $W = 32$. Let the number of connection requests be $K = 8, 16, 24,$ and 32. For each K, we randomly generate 50 instances, calculate the cost for each type of service, and report the average cost.

Fig. 15(a) plots the average cost versus the number of connection requests. Not surprisingly, we observe that the cost increases with the number of connection requests for each type of service. The IDHP solution has the highest cost, followed by DHP, SHP, DH, and SH. An interesting observation is that the ratio of the costs between any two types of services is almost constant across the entire network-load range. In particular, we see that the cost of the IDHP solution is 30% higher than DHP cost, the cost of DHP is about 60% higher than that of SHP, which implies that providing DHP is about 60% more expensive than providing single-homing protection. Similarly, DHP is about 150% more expensive than dual homing without protection; and about 300% more expensive than SH. Such results are very useful for service providers in helping determine the pricing for each level of service.

We also observe that SHP has a higher cost than DH. Both SHP and DH require two paths. However, the two paths in SHP are disjoint for the purpose of link protection, while the two paths in DH are shared to reduce cost. This implies that protecting link failures (protection) is more expensive than protecting access-node failures (dual homing).

In the second experiment, we study how the number of wavelengths can affect service costs. In a WDM network, each link has a limited number of wavelengths, each of which can be assigned to an individual connection request. Hence, there is a limit on the number of connections that can share a common link. Accordingly, when there are fewer wavelengths available on every link, some connection requests have to choose longer paths, and thus, will have a higher cost.

Fig. 15(b) shows the results of our experiments, wherein we set the number of nodes, $V = 50$, the maximum outgoing degree of a node, $D = 20$, the number of randomly generated connection requests, $K = 32$, and vary the number of available wavelengths on each link.

From Fig. 15(b), we see that the number of wavelengths has little impact on the cost of SH and DH, since there are relatively fewer links required for these two schemes. On the other hand, the cost of DHP increases when there are fewer wavelengths per link. This verifies our previous argument that fewer wavelengths will force some connections to choose longer paths with higher costs. Such a result is helpful for comparing the tradeoff between providing more wavelengths to reduce the communication costs and providing fewer wavelengths to increase the communication costs.

B. Dynamic DHP

Now, we use simulation to analyze the performance of the proposed heuristics for the dynamic DHP problem. We are interested in comparing the average total cost of the solutions obtained using MCNFH, MDSPH, MCSPH, and MSTH algorithms. We randomly generate problem instances in the same way as the static case, which is controlled by two parameters,

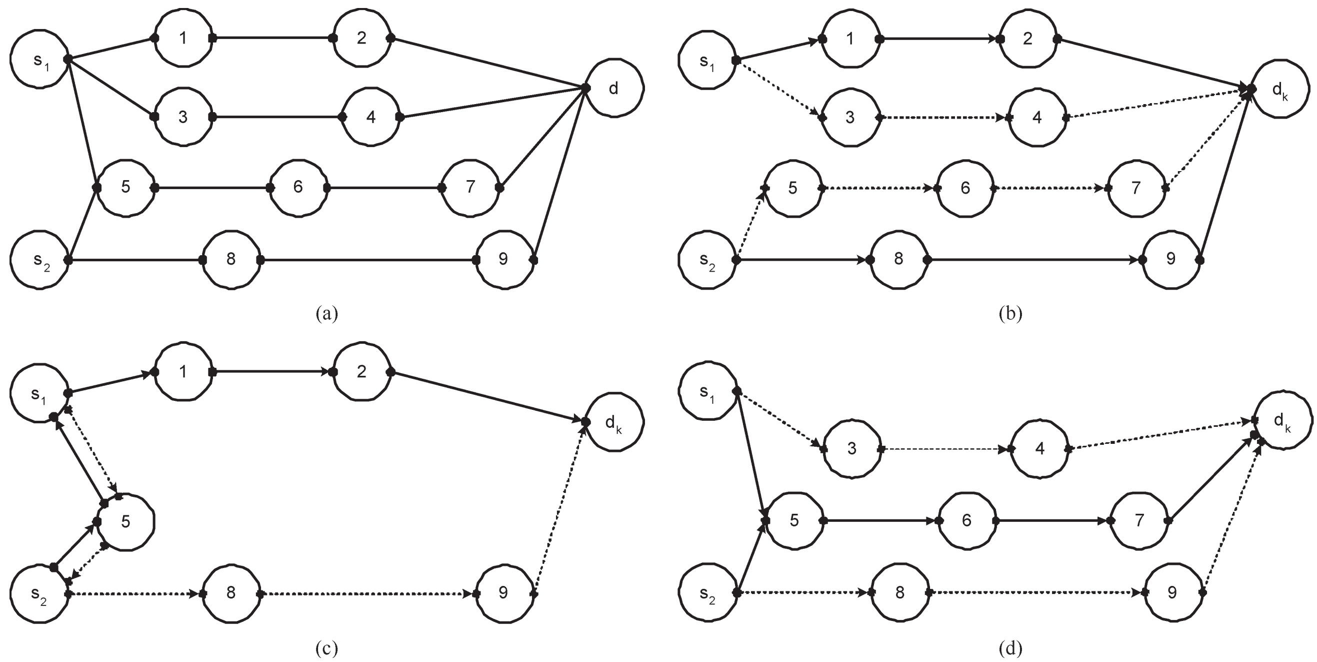

Fig. 14. Example: Comparison of different solutions for a given network topology. (a) Network topology. (b) Primary and backup lightpaths found by MCNFH and MDSPH, total cost = 13. (c) Primary and backup lightpaths found by MCSPH, total cost = 10. (d) Primary and backup lightpaths found by MSTH, total cost = 11.

TABLE I
TIME COMPLEXITY: DHP ALGORITHMS

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

the network size N and the maximum outgoing degree D. For the selection of dual homes, we consider both randomly selected dual homes, as well as the closest dual homes, the letter of which is to randomly select a pair of dual homes with a direct link. Once the dual homes are selected, we randomly select a destination and assume the current connection request is from the selected dual homes to the selected destination.

For each group of parameters, problem instances are generated until 1000 instances have feasible solutions by using MCNFH. All these instances are simultaneously solved by MDSPH, MCSPH, and MSTH algorithms, as well as an IDHP solution that is obtained by solving the static IDHP model with only a single connection request. By using the 4/3 approximation result of the MCNFH algorithm, we can also get a lower bound for the minimum cost, i.e., lower bound of the minimum cost $= 3/4$ MCNFH cost.

Fig. 16(a)–(b) plots the average cost for the proposed algorithms versus different values of N, when D is set to 10 with closest/random dual homes. In order to show the advantage of the integrated solution, we compare the algorithms with an IDHP 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, MCSPH, and MSTH is significantly lower than the IDHP solution. We also observe that MCNFH and MDSPH incur the same cost for the network scenarios considered. The performance of MSTH is slightly better than that of the network flow-based algorithms.

For both cases in Fig. 16(a)–(b), the performance of MCSPH is worse than the network flow-based algorithms. As shown in Fig. 16(a), when the dual homes are selected as the closest homes, the performance of MCSPH is closer to the network flow-based algorithms. However, when the dual homes are randomly selected, the performance difference between MCSPH and the other algorithms is bigger as shown in Fig. 16(b).

We also observe that if the paths from the closest dual homes to the destination for the current request is long, MCSPH works better than the other heuristic algorithms, since only two long paths need to be found. In other words, the average cost of MCSPH is lower than the other heuristic algorithms in a large sparse network where the dual homes are close to each other and the paths from the dual homes to the destination are long. Our expectation is validated by the simulation results shown in Fig. 17, where D is set to 4 and the network size ranges from 50, 100, 150, and 200 to 250. We specifically simulate the instances in which the total cost for MCNFH is larger than 25. By setting such a constraint, the advantage of MCSPH can surface. In Fig. 17, the performance of MCNFH, MDSPH, and MSTH are the same, and the performance of MCSPH is better than the other heuristic algorithms. Considering the simplicity of MCSPH, it is a good candidate for solving the dynamic DHP problem in large sparse networks.

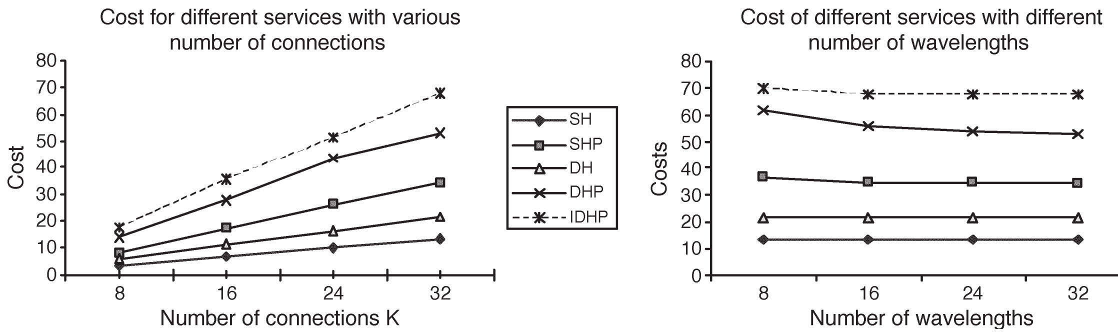

Fig. 15. Computational results for the static case: (a) cost for different services versus various number of connection requests and (b) cost for different services versus various number of wavelengths.

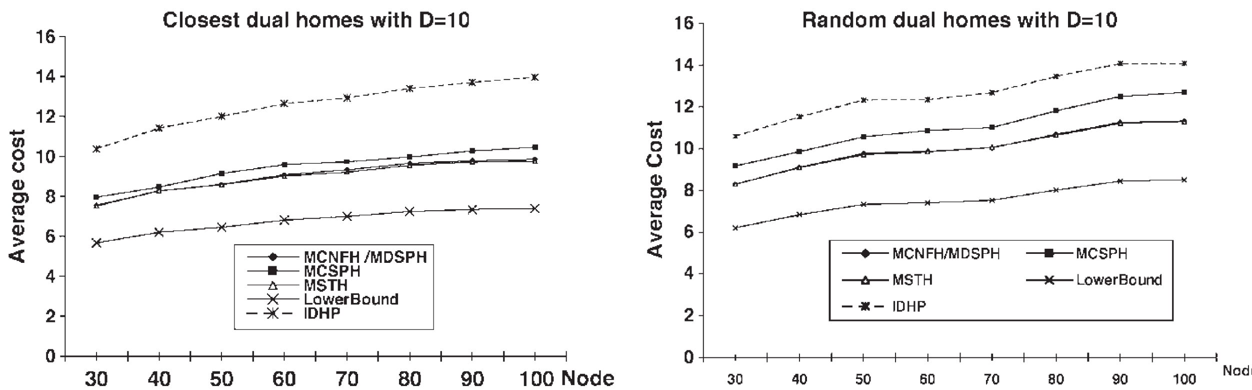

Fig. 16. Computational results for the dynamic case: (a) cost versus number of nodes (N) with closest dual homes and (b) cost versus number of nodes (N) with random dual homes.

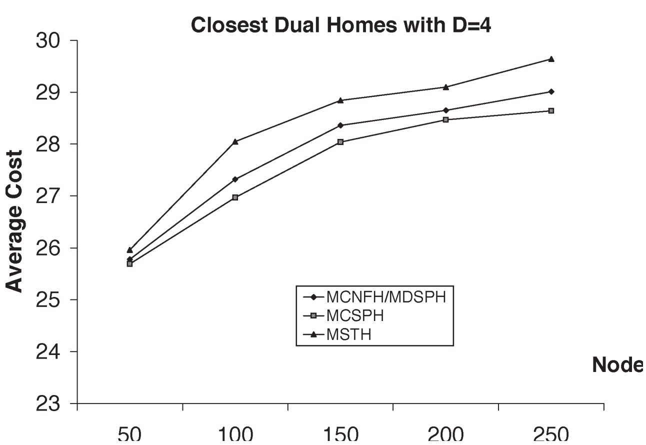

Fig. 17. Average cost versus the number of nodes in large sparse networks.

VII. Conclusion

We investigate the survivability issue in IP-over-wavelength-division-multiplexing (WDM) networks when a dual-homing architecture is provided in the access network. Our goal is to reduce the protection cost in the core network imposed by the dual-homing architecture in the access network. We study both the static case with a set of known connection requests and the dynamic case with a single connection-request arrival. The basic idea that motivates this research is that multihoming in the access network can reduce the protection cost in the core network if the protection is conducted in a coordinated way, which is supported by our computational results. Our research will help network design in terms of reliability and costs.

References

[1] E. Modiano and A. Narula, "Survivable lightpath routing: A new approach to the design of WDM-based networks," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 4, pp. 800–809, May 2002.

[2] D. Zhou and S. Subramaniam, "Survivability in optical networks," *IEEE Netw.*, vol. 14, no. 6, pp. 16–23, Nov./Dec. 2000.

[3] C. Lee and S. Koh, "A design of the minimum-cost ring-chain network with dual-homing survivability: A tabu search approach," *Comput. Oper. Res.*, vol. 24, no. 9, pp. 883–897, Sep. 1997.

[4] M. Willebeek-LeMair and P. Shahabuddin, "Approximating dependability measures of computer networks: An FDDI case study," *IEEE-ACM Trans. Netw.*, vol. 5, no. 2, pp. 311–327, Apr. 1997.

[5] F. Farahmand, A. F. Fumagalli, and M. Tacca, "Near-optimal design of WDM dual-ring with dual-crossconnect architecture," in *Proc. SPIE Optical Networking Communication Conf. (OptiComm)*, Boston, MA, Jul. 2002, vol. 4874, pp. 286–297.

[6] A. Proestaki and M. Sinclair, "Design and dimensioning of dual-homing hierarchical multi-ring networks," *Proc. Inst. Elect. Eng., Commun.*, vol. 147, no. 2, pp. 96–104, Apr. 2000.

[7] J. Shi and J. Fonseka, "Analysis and design of survivable telecommunications networks," *IEE Proc., Commun.*, vol. 144, no. 5, pp. 322–330, Oct. 1997.

[8] D. Din and S. Tseng, "A genetic algorithm for solving dual-homing cell assignment problem of the two-level wireless ATM network," *Comput. Commun.*, vol. 25, no. 17, pp. 1536–1547, Nov. 2002.

[9] T. Bates and Y. Rekhter, *Scalable Support for Multi-homed Multi-provider Connectivity*, 1998. RFC2260.

[10] 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," *J. Lightw. Technol.*, vol. 19, no. 2, pp. 137–149, Feb. 2001.

[11] G. Sasaki and C. Su, "The interface between IP and WDM and its effect on the cost of survivability," *IEEE Commun. Mag.*, vol. 41, no. 1, pp. 74–79, Jan. 2003.

[12] A. Orda and R. Rom, "Multihoming in computer networks: A topology-design approach," *Comput. Netw. ISDN Syst.*, vol. 18, no. 2, pp. 133–141, Feb. 1990.

[13] J. Wang, M. Yang, X. Qi, and R. P. Cook, "Dual-homing multicast protection," in *Proc. IEEE Global Telecommunications (GLOBECOM)*, Dallas, TX, 2004, vol. 2, pp. 1123–1127.

[14] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[15] J. W. Suurballe, "Disjoint paths in a network," *Networks*, vol. 4, no. 2, pp. 125–145, 1974.

Jianping Wang (M'03) received the B.Sc. and the M.Sc. degrees in computer science from Nankai University, Tianjin, China, in 1996 and 1999, respectively, and the Ph.D. degree in computer science from the University of Texas at Dallas in 2003.

She is currently an Assistant Professor at the Department of Computer Science, Georgia Southern University. Her research interests include optical networks and wireless networks.

Vinod M. Vokkarane (S'02–M'04) received the B.Eng. degree with honors in computer science and engineering from the University of Mysore, India, in 1999, and the M.S. and Ph.D. degrees in computer science from the University of Texas, Dallas, in 2001 and 2004, respectively.

He is currently an Assistant Professor at the Department of Computer and Information Science, University of Massachusetts at Dartmouth. His primary areas of research are in optical and wireless networking, with a special focus on the design of architectures and protocols for optical networks.

Dr. Vokkarane is a recipient of the University of Texas at Dallas Best Computer Science Ph.D. Dissertation Award (2003–2004).

Raja Jothi, photograph and biography not available at the time of publication.

Xiangtong Qi received the Ph.D. degree from the Department of Management Science and Information Systems, Red McCombs School of Business, University of Texas at Austin, in August 2003.

He is an Assistant Professor at the Department of Industrial Engineering and Logistics Management, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong. His main research interest is applied optimization with applications in scheduling, logistics, and networking.

Balaji Raghavachari received the Ph.D. degree from the Pennsylvania State University in 1992.

He is a Professor of Computer Science at the Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas. His research interests include the design and analysis of algorithms, database design, approximation algorithms, combinatorial optimization, network design, telecommunication networks, and vehicle routing and traversal problems.

Jason P. Jue (S'95–A'99–M'00–SM'04) received the B.S. degree in electrical engineering from the University of California, Berkeley, in 1990, the M.S. degree in electrical engineering from the University of California, Los Angeles, in 1991, and the Ph.D. degree in computer engineering from the University of California, Davis, in 1999.

He is currently an Associate Professor at the Department of Computer Science, University of Texas at Dallas. His research interests include optical networks, network control and management, and network survivability.