Reliability in Layered Networks With Random Link Failures

Kayi Lee, *Member, IEEE*, Hyang-Won Lee, *Member, IEEE*, and Eytan Modiano, *Senior Member, IEEE*

***Abstract*—We consider network reliability in layered networks where the lower layer experiences random link failures. In layered networks, each failure at the lower layer may lead to multiple failures at the upper layer. We generalize the classical polynomial expression for network reliability to the multilayer setting. Using random sampling techniques, we develop polynomial-time approximation algorithms for the failure polynomial. Our approach gives an approximate expression for reliability as a function of the link failure probability, eliminating the need to resample for different values of the failure probability. Furthermore, it gives insight on how the routings of the logical topology on the physical topology impact network reliability. We show that maximizing the min cut of the (layered) network maximizes reliability in the low-failure-probability regime. Based on this observation, we develop algorithms for routing the logical topology to maximize reliability.**

***Index Terms*—Lightpath routing, multilayer network, network reliability, random failures, random sampling, reliability approximation.**

I. Introduction

MODERN communication networks are constructed using a layered approach, with one or more electronic layers (e.g., IP, ATM, SONET) built on top of an optical fiber network. One important aspect that is unique in layered network design is the embedding of the logical electronic topology onto the physical fiber topology, called *lightpath routing*. The choice of the lightpath routing determines the way physical failures affect the logical links and, therefore, plays an important role in the survivability of layered network. In this paper, we investigate the survivability of layered networks assuming that physical links experience random failures.

One important connectivity requirement for a layered network is to keep the logical topology connected under the failures. For example, many common networking protocols (such as IP) that run on top of the logical topology rely on restoration as the recovery mechanism [1]. In the case of failures, the routers can rebuild the routing tables based on the new logical topology, such that the packet traffic continues to be delivered as long as the logical topology stays connected. Therefore, a natural survivability metric in this context is, given a lightpath routing, the probability that the logical topology remains connected. We call this probability the *cross-layer (network) reliability*.

The cross-layer reliability reflects the survivability performance achieved by the lightpath routing. In this paper, we address two important problems of cross-layer reliability.

1) *Reliability evaluation:* Given a lightpath routing of a layered network, how can the cross-layer reliability be efficiently computed?
2) *Reliability maximization:* Given the physical and logical topologies, how can we compute a lightpath routing with high reliability?

The reliability evaluation problem has been extensively studied for single-layer networks (defined as the probability that a single-layer network remains connected under random link failures). In particular, it was shown that it is $\#P$-complete to compute reliability in a single-layer setting exactly [2] or up to ϵ relative accuracy [3]. Therefore, computing cross-layer reliability is also $\#P$-complete as it is at least as difficult as the single-layer problem. We address this issue by developing an efficient method that can accurately estimate the cross-layer reliability with high probability. Although the cross-layer reliability can be exactly computed by enumerating all the failure states, such a brute-force enumeration takes 2000 min even on a small network such as the NSFNET, whereas our algorithm takes less than 1 h (this will be discussed in Section VI).

Clearly, the reliability of a layered network depends on both the physical and logical topologies. In particular, the single-layer reliability of the physical topology imposes an upper bound on the cross-layer reliability. However, even if both the physical and logical topologies are highly connected, the layered network may still have low reliability if the lightpath routing is poorly chosen. Hence, the lightpath routing design is equally important. Therefore, in this paper we study the reliability maximization problem in order to understand desirable properties of reliable lightpath routings. One important observation developed in this paper is that the optimal lightpath routing varies with the link failure probability in general. Therefore, under different values of link failure probability, the optimal lightpath routings may exhibit different properties. In modern high-speed networks such as the IP-over-WDM network, the probability of a physical link failure is typically very small. Therefore, in this paper we study optimal lightpath routings, assuming the link failure probability is small, and define the reliability maximization problem as finding the optimal lightpath routing under this low-failure-probability regime.

Manuscript received August 06, 2010; revised February 15, 2011; accepted April 02, 2011; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor C.-N. Chuah. Date of publication April 29, 2011; date of current version December 16, 2011. This work was supported by the National Science Foundation (NSF) under Grants CNS-0626781 and CNS-0830961 and the Defense Threat Reduction Agency (DTRA) under Grants HDTRA1-07-1-0004 and HDTRA-09-1-005. This paper was presented in part at the IEEE Conference on Computer Communications (INFOCOM), San Diego, CA, March 15–19, 2010.

K. Lee was with the Massachusetts Institute of Technology, Cambridge, MA 02139 USA. He is now with Google, Inc., Cambridge, MA 02142 USA (e-mail: kylee@mit.edu).

H.-W. Lee and E. Modiano are with the Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: hwlee@mit.edu; modiano@mit.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNET.2011.2143425

To the best of our knowledge, there is no known work dealing with cross-layer reliability. Our basic approach to address these two problems is to extend the polynomial expression for single-layer network reliability to the layered setting. This polynomial [cf. Equation (1)] provides a formula for network reliability as a function of the link failure probability. Hence, the cross-layer reliability can be estimated by approximating the coefficients of the polynomial.

One important aspect of this approach is that it is not tailored to a particular probability of link failure, and consequently, once we have established this polynomial, it can be immediately used for the reliability evaluation problem for *any* value of link failure probability. In addition, the coefficients in the polynomial expression contain important structural information of the lightpath routing, which provides important insights into the reliability maximization problem.

Our contributions can be summarized as follows.

- We develop polynomial-time approximation algorithms for cross-layer reliability computation using random sampling.
- We characterize the properties of optimal lightpath routings in the low-failure-probability regime and develop lightpath routing algorithms for reliability maximization in the low-failure-probability regime.

This paper is organized as follows. We first discuss some previous work in Section II, followed by a discussion of the model and background in Section III. In Sections IV–VII, we develop methods based on Monte Carlo simulation to approximate cross-layer reliability with provable accuracy and present our simulation results. In Section VIII, we address the reliability maximization problem by presenting several lightpath routing algorithms to maximize the reliability under the low-failure-probability regime. We present our simulation results for these algorithms in Section VIII-B. Finally, in Section IX, we briefly discuss how our reliability estimation algorithm can be extended to more general random failure models.

II. Previous Work

The network reliability estimation problem has been extensively studied in the single-layer setting. Since it is a difficult problem, most of the previous works in this context focused on approximating the actual reliability. Although there are some works aimed at exact computation of reliability through graph transformation and reduction [4]–[11], the applications of such methods are highly limited since they are targeted to particular topologies. Furthermore, those methods cannot be used for estimating cross-layer reliability because they assume independence between link failures, while failures are often correlated in multilayer networks.

Monte Carlo simulation was also used for estimating the single-layer reliability for some fixed link failure probability [12]–[14]. Using simulation, the reliability can be approximated to an arbitrary accuracy, but the number of required iterations tends to be very large especially when the failure probability is small. Moreover, the simulation must be repeated for different values of the failure probability.

Another approach is to use a polynomial expression for reliability [15] and estimate every coefficient appearing in the polynomial, where the reliability can be approximated using the estimated coefficients. The advantage of this approach over simulation is that once every coefficient is estimated, they can be used for any value of failure probability. Most of the works in this context have focused on bounding the coefficients by applying subgraph counting techniques and results from combinatorics [16]–[20]. This approach is computationally attractive, but its estimation accuracy is not guaranteed. Some previous works studied the regime of low failure probability by focusing on small cut sets [21], [22]. In [23], a random sampling technique is used to enhance those bounding results. In particular, [23] considers another form of the polynomial used in [24] and estimates some of the coefficients by enumerating spanning trees in the graph. These estimates are used to improve the algebraic bound in [24]. This approach is similar to our work in that it tries to approximate the coefficients in the polynomial through random sampling. However, the algorithm cannot be used to estimate cross-layer reliability because of the inherent structural difference between single-layer and multilayer networks. Specifically, one important step in the algorithm presented in [23] involves uniformly sampling spanning trees in the network. While this is easy to do in a single-layer network because all spanning trees have the same size, this special property is no longer true in the multilayer setting due to the new notion of lightpath routing [25]. Because of the structural differences between the single-layer and multilayer networks, estimating reliability in a multilayer network becomes a fundamentally different problem from its single-layer counterparts.

Our paper is the first attempt to address reliability estimation in multilayer networks. The most relevant work in multilayer setting is our work in [26], where a new survivability metric called the Min Cross Layer Cut (MCLC) is introduced by generalizing the traditional min cut. The MCLC is defined as the minimum number of physical link failures that are needed to disconnect the logical topology, and it is closely related to cross-layer reliability. As MCLC grows, it becomes harder to disconnect the logical topology, which in turn implies high reliability. Consequently, a lightpath routing that maximizes the MCLC may achieve good reliability performance. In fact, we will evaluate the reliability performance of the lightpath routing algorithms from [26] under the random failure model considered in this paper. Thus, our work in this paper can be viewed as an extension of [26] to the case of random link failures.

III. Model and Background

We consider a layered network $\mathcal{G}$ that consists of a logical topology $G_{\mathrm{L}} = (V_{\mathrm{L}}, E_{\mathrm{L}})$ built on top of a physical topology $G_{\mathrm{P}} = (V_{\mathrm{P}}, E_{\mathrm{P}})$ through a lightpath routing, where V and E are the set of nodes and links, respectively. In the context of WDM network, a logical link is called a *lightpath*, and each lightpath is routed over the physical topology. This *lightpath routing* is denoted by $f = [f_{ij}^{st}, (i,j) \in E_{\mathrm{P}}, (s,t) \in E_{\mathrm{L}}]$, where f_{ij}^{st} takes the value 1 if logical link (s,t) is routed over physical link (i,j), and 0 otherwise.

Each physical link fails independently with probability p, and if a physical link (i,j) fails, all the logical links (s,t) carried

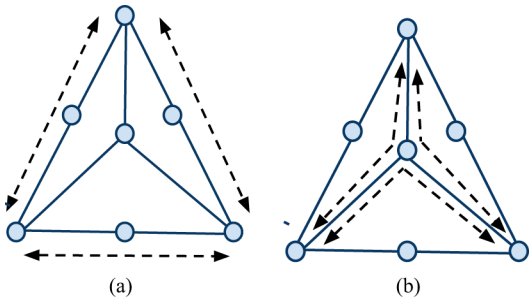


Fig. 1. Optimal lightpath routing may depend on the link failure probability. The logical topologies are connected by solid and dashed lines, respectively. (a) Disjoint routing. (b) Nondisjoint routing.

over (i, j) [i.e., (s, t) such that $f_{ij}^{st} = 1$] also fail. A set S of physical links is called a *cross-layer cut* if and only if the failure of the links in S causes some logical nodes to be disconnected from one another in the logical topology. We also define the *network state* as the subset of physical links S that failed. Hence, if S is a cross-layer cut, the network state S represents a *disconnected* network state. Otherwise, it is a *connected* state. The *cross-layer reliability*, or reliability for simplicity, of a multilayer network is the probability that the network state S is a connected state.

A. Importance of Lightpath Routing

It is important to note that the reliability depends on the underlying lightpath routing. For example, in Fig. 1, the logical topology consists of three logical nodes connected in a cycle. Suppose every physical link fails independently with probability p. The first lightpath routing routes the logical links over the physically disjoint and shortest paths. Under this routing, the logical network will be disconnected with probability $1-3(1-p)^4+(1-p)^6$. In the second lightpath routing, the lightpaths are routed over the nondisjoint paths (with the same number of physical hops). Under this routing, the logical network will be disconnected with probability $1-(1-p)^3$. While disjoint path routing is generally considered more reliable, it is only true in this example for small values of p. For large p (e.g., $p > 0.7$), the second lightpath routing is actually more reliable. Therefore, whether one lightpath routing is better than another depends on the value of p.

B. Cross-Layer Failure Polynomial

In general, it is difficult to compare lightpath routings since computing the cross-layer reliability is $\#P$-complete. Hence, we first need to develop an algorithm that can accurately estimate the reliability. However, even if an estimation algorithm is capable of comparing lightpath routings for a certain value of p, it does not mean that the comparison is valid at other values (as shown in the above example), and this may require rerunning the algorithm when the probability of interest p has changed. Hence, for this purpose, it is useful to develop an estimation method such that once an estimation is made, the result can be used for every value of p. Having a failure polynomial makes it possible to compare reliability among different lightpath routings at all values of p. Furthermore, as will be discussed, the failure polynomial gives insights to the design of lightpath routings for better reliability.

We introduce a polynomial expression for reliability that is a natural extension of the single-layer polynomial [15] to the cross-layer setting. Assume that there are m physical links, i.e., $|E_{\mathrm{P}}| = m$. The probability associated with a network state S with exactly i physical link failures (i.e., $|S| = i$) is $p^i(1-p)^{m-i}$. Let N_i be the number of disconnected network states S with $|S| = i$, then the probability that the network gets disconnected is simply the sum of the probabilities over all disconnected states, i.e.,

$$F(p) = \sum_{i=0}^{m} N_i p^i (1-p)^{m-i}. \tag{1}$$

Therefore, the failure probability of a multilayer network can be expressed as a polynomial in p. The function $F(p)$ will be called *cross-layer failure polynomial* or simply the *failure polynomial*. The vector $[N_0, \ldots, N_m]$ plays an important role in assessing the reliability of a network. In particular, one can simply plug the value of p in the above failure polynomial to compute the reliability if the values of N_i are known.

Intuitively, each N_i represents the number of cross-layer cuts of size i in the network, and its value can be obtained by exhaustively enumerating all possible fiber sets with size i and counting the number of fiber sets that are cross-layer cuts. However, since the number of fiber sets with size i is $\binom{m}{i}$, this enumeration approach will enumerate a total of 2^m fiber sets in order to compute all the values of N_i. Therefore, this approach is feasible only for small networks. It is important to devise a more efficient way to estimate the values of N_i's.

The coefficient of the failure polynomial also contains important structural information about the lightpath routing. Clearly, if $N_i > 0$, then $N_j > 0, \forall j > i$ (because any cut of size i will still be a cut with the addition of more failed links). The smallest i such that $N_i > 0$ is of special importance because it represents the MCLC [26] of the network, i.e., it is the minimum number of physical link failures needed to disconnect the logical network. Although computing the MCLC is NP-hard [26], for practical purposes, the MCLC of a network is typically upper-bounded by some constant, such as the minimum node degree of the logical network. Therefore, for the rest of the paper, we denote the MCLC value of the network by d and assume that it is a constant independent of the physical network size. It is important to note that $N_i = 0, \forall i < d$, and the term $N_d p^d (1-p)^{m-d}$ in the failure polynomial dominates for small values of p. Consequently, if a lightpath routing tries to maximize MCLC, i.e., make d as large as possible, it will achieve good reliability in the low-failure-probability regime. On the other hand, its reliability performance is not guaranteed in other regimes. This will be further discussed in Section VIII, where we present some lightpath routing algorithms for achieving maximum reliability. A similar observation was made for single-layer networks in [27].

Although the failure polynomial in (1) takes the same form as its single-layer counterpart [15], computing the failure polynomials in the single-layer and multilayer settings are fundamentally different problems due to the inherent structural differences between the two types of networks. For example, it was

shown in [26] that computing the MCLC for a multilayer network is NP-hard. On the other hand, computing the min-cut for a single-layer network can be done in polynomial time. Therefore, although the general approach of computing the failure polynomial applies to both single-layer and multilayer networks, the actual procedure to estimate the value of each N_i can be vastly different.

In this paper, we focus on approximating the cross-layer failure polynomial. We will use the following notions of approximation.

Definition 1 (Relative Approximation): A function $\hat{F}(p)$ is an ϵ-approximation for the failure polynomial $F(p)$ if

$$|F(p) - \hat{F}(p)| \leq \epsilon F(p), \qquad \text{for all } p \in [0,1].$$

This relative error is typically the measure of interest in the literature of reliability estimation. However, as mentioned above, it is also $\#P$-complete to approximate the reliability to ϵ accuracy [3]. Hence, it is not likely that there exists a *deterministic* ϵ-approximation algorithm requiring reasonably low computation. For this reason, our estimation focuses on the following probabilistic approximation.

Definition 2 [(ϵ,δ)-Approximation]: A function $\hat{F}(p)$ is an (ϵ,δ)-approximation for the failure polynomial $F(p)$ if

$$\Pr\left[|F(p) - \hat{F}(p)| \leq \epsilon F(p)\right] \geq (1-\delta), \qquad \text{for all } p \in [0,1].$$

In other words, an (ϵ,δ)-approximation algorithm approximates the polynomial to ϵ relative accuracy with high probability. In Sections IV and V, we will present randomized (ϵ,δ)-approximation algorithms for the failure polynomial.

C. Monte Carlo Simulation

Our estimation algorithm is based on Monte Carlo simulation techniques. The central theme of such Monte Carlo techniques is based on the *Estimator Theorem*, presented below. Let U be a ground set defined as the set of all possible events (e.g., all network states), and G be a subset of U (e.g., disconnected network states). Suppose that we want to estimate $|G|$. To do this, the Monte Carlo method samples an element e from U uniformly at random for T times and counts the number of times N such that $e \in G$. The Monte Carlo method returns the quantity $\frac{N}{T}|U|$ as an estimator of $|G|$. The Estimator Theorem states the following.

Theorem 1 (Estimator Theorem [28]): Let $\rho = \frac{|G|}{|U|}$. Then, the Monte Carlo method yields an ϵ-approximation to $|G|$ with probability at least 1-δ, provided that

$$T \geq \frac{4}{\epsilon^2 \rho} \ln \frac{2}{\delta}.$$

In other words, if we sample from the ground set U frequently enough, we can estimate $|G|$ accurately with high probability. According to Theorem 1, the ratio ρ, called the *density* of the set G, is inversely proportional to the required number of iterations T. In the following sections, we will define the sets G and U in various ways to ensure high ρ value and propose polynomial-time Monte Carlo methods to compute approximations of the failure polynomial.

IV. Estimating Cross-Layer Reliability

Our approach to approximating the cross-layer failure polynomial is to estimate the values of N_i in Equation (1). If we can estimate each N_i with sufficient accuracy, we will obtain an approximate failure polynomial for the multilayer network. The idea is formalized in the following theorem.

Theorem 2: Let $\hat{N}_i$ be an ϵ-approximation of N_i for all $i \in \{1,\ldots,m\}$. Then, the function $\hat{F}(p) = \sum_{i=0}^{m} \hat{N}_i p^i (1-p)^{m-i}$ is an ϵ-approximation for the failure polynomial.

Proof: For all $0 \leq p \leq 1$

$$\begin{aligned}|\hat{F}(p) - F(p)| &\leq \sum_{i=0}^{m} |(\hat{N}_i - N_i)| p^i (1-p)^{m-i} \\ &\leq \sum_{i=0}^{m} \epsilon N_i p^i (1-p)^{m-i} = \epsilon F(p).\end{aligned}$$

■

Corollary 1: Let A be an algorithm that computes an $\left(\epsilon, \frac{\delta}{m+1}\right)$-approximation for each N_i. Then, A gives an (ϵ,δ)-approximation algorithm for the failure polynomial.

Proof: By the union bound, the probability that all the N_i estimates are ϵ-approximate is at least $1 - \sum_{i=0}^{m} \frac{\delta}{m+1} = 1-\delta$. By Theorem 2, A gives an (ϵ,δ)-approximation algorithm for the failure polynomial. ■

Therefore, for the rest of the section, our goal is to estimate each N_i with ϵ accuracy and probability at least $1 - \frac{\delta}{m+1}$.

A. Estimating N_i

Let $\mathcal{H}_i$ be the family of all subsets of E_{P} with exactly i physical links. Clearly, N_i is the number of subsets in $\mathcal{H}_i$ that are cross-layer cuts. Hence, one can compute the exact value of N_i by enumerating all subsets in $\mathcal{H}_i$ and counting the number of cross-layer cuts. However, the number of subsets to enumerate is $\binom{m}{i}$, which can be prohibitively large.

An alternative approach to estimating N_i is to carry out Monte Carlo simulation on $\mathcal{H}_i$. Suppose we sample uniformly at random from $\mathcal{H}_i$ for T times and count the number of cross-layer cuts W in the sample. The Estimator Theorem guarantees that $\binom{m}{i}\frac{W}{T}$ is an $\left(\epsilon, \frac{\delta}{m+1}\right)$-approximation, provided that

$$T \geq \frac{4}{\epsilon^2 \rho_i} \ln \frac{2(m+1)}{\delta} \tag{2}$$

where $\rho_i = \frac{N_i}{\binom{m}{i}}$ is the density of cross-layer cuts in $\mathcal{H}_i$. The main issue here is that the exact value for ρ_i, which depends on N_i, is unknown to us. However, if we substitute ρ_i in (2) with a lower bound of ρ_i, the sample size will be guaranteed to be no less than the required value. Therefore, it is important to establish a good lower bound for ρ_i in order to keep the sample size small while achieving the desired accuracy.

B. Lower-Bounding ρ_i

Given a layered network, suppose its Min Cross Layer Cut value d is known, Theorem 3 gives a lower bound on ρ_i.

Theorem 3: For $i \geq d, \rho_i \geq \frac{\binom{m-d}{i-d}}{\binom{m}{i}}$.

Proof: Since d is the Min Cross Layer Cut value, there exists a cross-layer cut S with size d. Any superset of S with i physical links is therefore also a cross-layer cut. Since there are a total of $\binom{m-d}{i-d}$ such supersets, we have $N_i \geq \binom{m-d}{i-d}$, and the theorem follows immediately. ∎

Therefore, we can use $\tilde{\rho}_i = \frac{\binom{m-d}{i-d}}{\binom{m}{i}}$ as the lower bound for ρ_i in (2) to estimate N_i, with the following observations.

1) The MCLC value d needs to be known in advance.
2) The sample size can be very large for small values of i. For example, when $i = d$, the sample size T required is $\frac{4\binom{m}{d}}{\epsilon^2} \ln \frac{2(m+1)}{\delta}$, which is no better than enumerating all sets in $\mathcal{H}^d$ by brute force.
3) The lower bound $\tilde{\rho}_i$ increases with i. In particular, $\frac{\tilde{\rho}_{i+1}}{\tilde{\rho}_i} = 1 + \frac{d}{i+1-d}$. Therefore, the sample size required to estimate N_i decreases with i.

In Section IV-D, we will present an algorithm that combines the enumeration and Monte Carlo methods to take advantage of their different strengths. In Section V, we will present an enhanced version of the algorithm that significantly reduces the number of iterations by establishing a much tighter lower bound on ρ_i. The final outcome is an (ϵ, δ)-approximation algorithm for the failure polynomial $F(p)$ that requires only a polynomial number of iterations.

C. Combined Enumeration and Monte Carlo Approach

Recall that N_i can be estimated with two different approaches, brute-force enumeration and Monte Carlo. The two approaches can be combined to design an efficient (ϵ, δ)-approximation algorithm for the failure polynomial.

The key observation for the combined approach is that brute-force enumeration works well when i is small, and the Monte Carlo method works well when i is large. Therefore, it makes sense to use the enumeration method to find the Min Cross Layer Cut value d as well as the associated value N_d. Once we obtain the value of d, we can decide on the fly whether to use the enumeration method or the Monte Carlo method to estimate each N_i by comparing the number of fiber sets of size i that needs to be generated by each method. The total number of fiber sets generated by this combined approach will be

$$\sum_{i=1}^{d} \binom{m}{i} + \sum_{i=d+1}^{m} \min\left\{ \binom{m}{i}, \frac{4\binom{m}{i}}{\epsilon^2 \binom{m-d}{i-d}} \ln \frac{2(m+1)}{\delta} \right\}$$

where the terms inside the min operator are the number of fiber sets generated by enumeration and Monte Carlo methods, respectively. The total number of iterations can be upper-bounded as follows:

$$\begin{aligned} &\sum_{i=0}^{d} \binom{m}{i} + \sum_{i=d+1}^{m} \min\left\{ \binom{m}{i}, \frac{4\binom{m}{i}}{\epsilon^2 \binom{m-d}{i-d}} \ln \frac{2(m+1)}{\delta} \right\} \\ &\quad \leq (m+1)^d + \frac{4}{\epsilon^2} \ln \frac{2(m+1)}{\delta} \sum_{i=d+1}^{m} \frac{\binom{m}{i}}{\binom{m-d}{i-d}} \\ &\quad = (m+1)^d + \frac{4}{\epsilon^2} \ln \frac{2(m+1)}{\delta} \sum_{i=d+1}^{m} \frac{\binom{m}{d}}{\binom{i}{d}} \\ &\quad \leq (m+1)^d + \frac{4\binom{m}{d} d!}{\epsilon^2} \ln \frac{2(m+1)}{\delta} \sum_{i=1}^{m-d} \frac{1}{i^d} \\ &\quad = \begin{cases} O(m \log^2 m), & \text{if } d = 1 \\ O(m^d \log m), & \text{if } d \geq 2. \end{cases} \end{aligned}$$

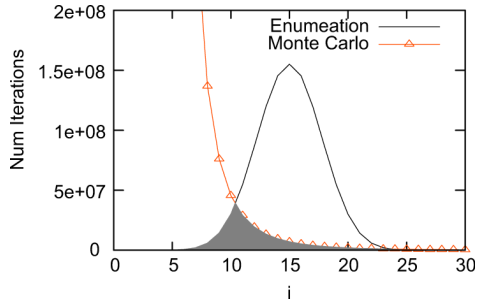


Fig. 2. Monte Carlo versus enumeration: Number of iterations for estimating N_i, for a network with 30 physical links, $\epsilon = 0.01$, $\delta = \frac{0.01}{31}$, $d = 4$. The shaded region represents the required iterations for the combined approach.

Each iteration involves testing whether the generated fiber set is a cross-layer cut, which can be done in linear time using Breadth-First Search over the logical topology with the links using the failed physical fibers removed. This gives us a polynomial-time approximation algorithm to the reliability estimation problem.

The improvement in running time of this combined approach is illustrated by Fig. 2.

V. Improved ρ_i Lower Bounds for Reliability Estimation

The running time performance of the algorithm introduced in Section IV hinges on the tightness of the lower bounds ρ_i used for the algorithm. In this section, we discuss ways to establish tighter lower bounds.

The idea behind these improved bounds is based on the observation that any superset of a cross-layer cut is also a cross-layer cut. Let $\mathcal{F} = \{C_1, \ldots, C_n\}$ be a collection of cross-layer cuts. For each $C_j \in \mathcal{F}$, let $\partial_i(C_j) \subseteq \mathcal{H}_i$ be the family of supersets of C_j with i physical links. Similarly, let $\partial_i(\mathcal{F}) = \cup_{C_j \in \mathcal{F}} \partial_i(C_j)$ be the union over all $\partial_i(C_j)$. Using the terminology in [29], the family of subsets $\partial_i(\mathcal{F})$ is called the ith *upper shadow* for $\mathcal{F}$. The following theorem provides a lower bound on ρ_i in terms of ∂_i.

Theorem 4: Let $\mathcal{F}$ be a collection of cross-layer cuts with size less than i, then $\rho_i \geq \frac{|\partial_i(\mathcal{F})|}{\binom{m}{i}}$.

Proof: Every set $S \in \partial_i(\mathcal{F})$ is a superset of the some cross-layer cut in $\mathcal{F}$ and is therefore a cross-layer cut with size i. Thus, $\partial_i(\mathcal{F})$ is a collection of cross-layer cuts with size i, which implies $|\partial_i(\mathcal{F})| \leq N_i$. It follows that $\frac{|\partial_i(\mathcal{F})|}{\binom{m}{i}} \leq \frac{N_i}{\binom{m}{i}} = \rho_i$. ∎

Therefore, if we know the value of $|\partial_i(\mathcal{F})|$, we can use $\frac{|\partial_i(\mathcal{F})|}{\binom{m}{i}}$ as the lower bound for ρ_i in the Monte Carlo method to estimate N_i. Note that if $\mathcal{F}$ contains only a Min Cross Layer Cut of the network, the value of $\frac{|\partial_i(\mathcal{F}|}{\binom{m}{i}}$ is equal to the bound given by Theorem 3. Therefore, Theorem 4 generalizes the lower bound result in Section IV-B.

Although the value of each $|\partial_i(C_j)| = \binom{m-|C_j|}{i-|C_j|}$ can be computed easily, finding the size of the union $\partial_i(\mathcal{F}) = \cup_{C_j \in \mathcal{F}} \partial_i(C_j)$ can be difficult because the sets $\partial_i(C_j)$ are not disjoint. Instead of computing $|\partial_i(\mathcal{F})|$ precisely, we introduce techniques for lower-bounding $|\partial_i(\mathcal{F})|$. The first technique, introduced in Section V-A, is based on importance sampling. The second technique, introduced in Section V-B, is a combinatorial result based on the Kruskal–Katona Theorem [29].

A. Lower Bound Based on Importance Sampling

Given a set of cross-layer cuts $\mathcal{F}$, the problem of estimating the size of its upper shadow $\partial_i(\mathcal{F})$ can be formulated as the Union of Sets Problem [30], for which a Monte-Carlo-based approach exists using the technique of importance sampling. We summarize the result in this section and leave the detailed proofs in Appendix A.

Theorem 5: Let $\mathcal{F} = \{C_1, \ldots, C_n\}$ be a collection of cross-layer cuts of the layered network. For each $C_j \in \mathcal{F}$, let $\partial_i(C_j)$ be the ith upper shadow of C_j. There exists a Monte Carlo method that produces an $(\epsilon_{lb}, \delta_{lb})$-approximation, $\hat{L}^i$, for $L_i = |\cup_{C_j \in \mathcal{F}} \partial_i(C_j)|$, provided that the number of samples is at least

$$T_{lb} = \frac{4|\mathcal{F}|}{\epsilon_{lb}^2} \ln \frac{2}{\delta_{lb}}. \tag{3}$$

Proof: Let $U = \{(S, j) : j \in \{1, \ldots, |\mathcal{F}|\}, S \in \partial_i(C_j)\}$ be the ground set for the Monte Carlo algorithm, and let $G = \{(S, j) : S \in \partial_i(\mathcal{F}), j = \min\{k : S \in \partial_i(C_k)\}\}$ be the events of interest. We show in Appendix A that the ground set U can be sampled uniformly at random. Since $|G| = |\partial_i(\mathcal{F})|$ and $\frac{|G|}{|U|} \geq \frac{1}{|\mathcal{F}|}$, Theorem 5 follows immediately from the Estimator Theorem. ∎

Theorem 5 implies $\hat{\rho}_i = \frac{\hat{L}^i}{(1+\epsilon_{lb})\binom{m}{i}}$ is a lower bound on ρ_i with probability at least $1-\delta_{lb}$. The following theorem describes how such a probabilistic lower bound can be used to estimate N_i.

Theorem 6: Let $\hat{L}_i$ be an $(\epsilon_{lb}, \delta_{lb})$-approximation for $|\cup_{C_j \in \mathcal{F}} \partial_i(C_j)|$. Then, the Monte Carlo method described in Section IV-A yields an $(\epsilon_{mc}, \delta_{lb} + \delta_{mc})$-approximation for N_i, provided that the number of samples is at least

$$T_{mc} = \frac{4(1+\epsilon_{lb})\binom{m}{i}}{\epsilon_{mc}^2 \hat{L}_i} \ln \frac{2}{\delta_{mc}}. \tag{4}$$

Proof: By definition of $\hat{L}_i$, the probability that $\hat{\rho}_i = \frac{\hat{L}_i}{(1+\epsilon_{lb})\binom{m}{i}}$ is *not* a lower bound on ρ_i is at most δ_{lb}. Given that $\hat{\rho}_i$ is a lower bound for ρ_i, by the Estimator Theorem, the probability that $\hat{N}_i$ is *not* an ϵ_{mc}-approximation for N_i is at most δ_{mc}. Hence, by the union bound, the probability that none of these "bad" events happens is at least $1 - (\delta_{lb} + \delta_{mc})$, and the theorem follows. ∎

To apply this result to reliability estimation, we can modify our algorithm presented in Section IV-C to also maintain the collection $\mathcal{F}$ of cross-layer cuts as we carry out the enumeration or Monte Carlo methods. Specifically, as we discover a cross-layer cut C_j with size i when estimating N_i, we will add the cut C_j to our collection $\mathcal{F}$. When we move on to estimate N_{i+1}, we will have a collection $\mathcal{F}$ of cross-layer cuts with size i or smaller. We can therefore apply Theorem 4 to obtain a lower bound for N_{i+1}. Note that the size of $\partial_i(\mathcal{F})$ is monotonic in $\mathcal{F}$. Therefore, the more cross-layer cuts that are included in $\mathcal{F}$, the better the lower bound is.

B. Lower Bound Based on Kruskal–Katona Theorem

We can also derive a lower bound on ρ_i based on the values of N_j for $j < i$ using the Kruskal–Katona Theorem. Let $[m] = \{1, \ldots, m\}$, i.e., $[m]$ is the enumeration of physical links. Let $\mathcal{H}_i^m = \{S \subseteq [m] : |S| = i\}$ be a family of subsets of $[m]$ with size i. For any $\mathcal{F} \in \mathcal{H}_i^m$, we denote $\partial_i^m(\mathcal{F})$ to be the ith upper shadow over $[m]$ for $\mathcal{F}$.

We define the *lexicographic ordering* on $\mathcal{H}_i^m$ as follows. Given any two subsets S_1 and S_2 in $\mathcal{H}_i^m$, S_1 is *lexicographically smaller than* S_2 if and only if $\min\{i : i \in S_1 \Delta S_2\} \in S_1$, where Δ denotes the symmetric difference between the two sets, i.e., $S_1 \Delta S_2 = S_1 \cup S_2 - S_1 \cap S_2$. For example, the set {1,2,4} is lexicographically smaller than {1,3,4} because the smallest element where the two sets differ, 2, is in the first set.

Given $\mathcal{H}_i^m$, the family of all subsets with size i, let $\mathcal{H}_i^m(k) \subseteq \mathcal{H}_i^m$ be the first k elements of $\mathcal{H}_i^m$ under the lexicographical ordering. The Kruskal–Katona Theorem states that $\mathcal{H}_i^m(k)$ yields the smallest upper shadow among all k-subsets of $\mathcal{H}_i^m$.

Theorem 7 [29]: For any $i < j$ and $\mathcal{F} \subseteq \mathcal{H}_i^m$

$$|\partial_j^m(\mathcal{H}_i^m(|\mathcal{F}|))| \leq |\partial_j^m(\mathcal{F})|. \tag{5}$$

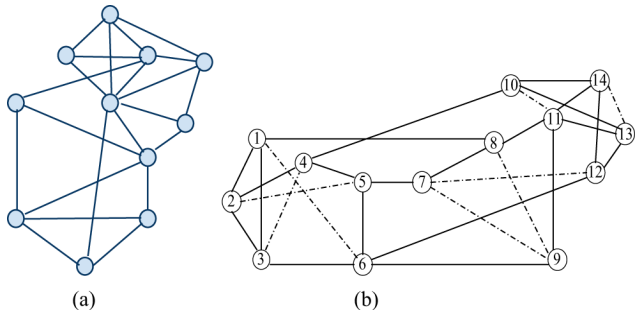
In other words, for a fixed value of k, the upper shadow for $\mathcal{F}$ with $|\mathcal{F}| = k$ is minimized if $\mathcal{F}$ consists of the first k subsets of $\mathcal{H}_i^m$ in lexicographical order. Therefore, suppose a multilayer network has a N_i cross-layer cuts with size i, Theorem 7 implies that $N_j \geq |\partial_j^m(\mathcal{H}_i^m(N_i))|$ for all $j > i$. We prove the following recursive formula for $|\partial_j^m(\mathcal{H}_i^m(N_i))|$.

Theorem 8: For $i < j \leq m$ and $1 \leq k \leq \binom{m}{i}$, let $w = \max\{0 \leq r < i : \binom{m-r}{i-r} \geq k\}$. Also, let $t = m - (w+1)$, $u = j - (w+1)$, and $v = i - (w+1)$. Then

$$|\partial_j^m(\mathcal{H}_i^m(k))| = \begin{cases} \binom{m-i}{j-i}, & \text{if } k = 1 \\ \binom{t}{u} + |\partial_{u+1}^t(\mathcal{H}_{v+1}^t(k - \binom{t}{v}))|, & \text{otherwise.} \end{cases}$$

Proof: See Appendix B. ∎

When estimating N_j in the jth round, the algorithm has already discovered a collection of cross-layer cuts with size i for each $i < j$ either by sampling or exhaustive enumeration. Let $\bar{N}_i$ be the number of cross-layer cuts with size i seen by the algorithm. Then, N_j is lower-bounded by $\max_{1 \leq i < j} |\partial_j^m(\mathcal{H}_i^m(\bar{N}_i))|$, where each term $|\partial_j^m(\mathcal{H}_i^m(\bar{N}_i))|$ can be computed easily using the recursive formula in Theorem 8. Notice that the original lower bound in Theorem 3 is a special case where a single MCLC is assumed and (according to Theorem 8) N_j is lower-bounded by $|\partial_j^m(\mathcal{H}_d^m(\bar{N}_d = 1))| = \binom{m-d}{j-d}$ for each $j > d$. Theorem 8 improves this bound by accounting for more cross-layer cuts, and therefore, it can be used to further reduce the number of iterations required by the Monte Carlo algorithm. We note, however, that the enhanced lower bounds obtained by Theorems 5 and 8 may still result in the same order of $O(m^{d+1} \log m)$ iterations. Nevertheless, simulation studies in



(a) (b)

Fig. 3. Physical topologies. (a) NJLATA. (b) Augmented NSFNET. The dashed lines are the new links.

Section VI show that these enhanced bounds can substantially reduce the number of iterations.

We also note that a probabilistic lower bound for N_j can be established by using the estimated value $\hat{N}_i$ instead of $\bar{N}_i$ and adjusting the parameters δ and ϵ in a way similar to Theorem 6. We omit the details here for brevity.

VI. Empirical Studies

We present some empirical results about the reliability estimation algorithms described in Sections IV and V. We first evaluate the efficiency of the algorithms, both in terms of the total number of fiber sets sampled and the running time, with parameters $\epsilon = \delta = 0.01$. We then compare the actual relative error of the failure polynomials computed by the algorithms with the theoretical guarantee given by the parameter ϵ.

We consider two different physical topologies (Fig. 3): the NJLATA network with 22 links and connectivity 2, and the augmented NSFNET with 29 links with connectivity 4. For the NJLATA network, 250 random logical topologies with 6–10 nodes were generated, and each logical link is routed over the physical route with the minimum number of hops. For the NSFNET, 350 random logical topologies with 6–12 nodes were generated, and the lightpath routings are created using the MCF (multicommodity flow) algorithms described in Section VIII. These two sets of layered networks represent topologies with different sizes as well as different levels of connectivity. Note that each of the logical topologies laid on top of the physical topology one at a time, and its reliability is estimated.

For each lightpath routing, we ran the following variants of reliability estimation algorithms to compute the failure polynomials. Each algorithm estimates the value of each N_i in the failure polynomial by generating a sample of fiber sets with i physical links and counting the number of cross-layer cuts in the sample. The fraction of cross-layer cuts in the sample can then be used to estimate the value of N_i. The algorithms only differ in how the sample of fiber sets is generated, which is described as follow.

1) ENUM: Each N_i is computed by exhaustively enumerating all possible $\binom{m}{i}$ sets with i physical links.
2) $\text{MIXED}_{\text{original}}$: The sets of i physical links are generated based on the combined enumeration and Monte Carlo methods introduced in Section IV-C, where the sample size in the Monte Carlo method is determined by the lower bound given by Theorem 3.

TABLE I
Number of Iterations for Each Algorithm

Algorithm	NJLATA		NSFNET	
	Sample Size	Time	Sample Size	Time
ENUM	4,194,304	15m25s	536,870,912	2206m33s
$\text{MIXED}_{\text{original}}$	4,162,237	15m14s	46,900,857	279m37s
MIXED_{KK}	3,050,462	11m22s	15,467,815	40m32s
$\text{MIXED}_{\text{sample}}$	3,186,712	13m3s	11,968,535	58m17s

3) MIXED_{KK}: The sets of i physical links are generated based on the combined enumeration and Monte Carlo method, where the sample size in the Monte Carlo method is determined by the enhanced lower bound given by Theorem 8.
4) $\text{MIXED}_{\text{sample}}$: The sets of i physical links are generated based on the combined enumeration and Monte Carlo method, where the sample size in the Monte Carlo method is determined by the enhanced lower bound given by the importance sampling technique in Section V-A, with $\epsilon_{mc} = 0.01$, $\epsilon_{lb} = 0.1$, $\delta_{mc} = \delta_{lb} = \frac{0.005}{30}$.

Table I shows the average number of samples as well as the running time required for each algorithm to compute the failure polynomial. Compared to the brute-force approach ENUM that enumerates all possible 2^m sets of physical links, the polynomial-time algorithms based on the Monte Carlo method have significantly reduced the running time to estimate the reliability with the desired accuracy, especially for the larger NSFNET. A comparison between the algorithm $\text{MIXED}_{\text{original}}$ and its enhanced counterparts MIXED_{KK} and $\text{MIXED}_{\text{sample}}$ reflects the impact of the tightness of the N_i lower bound. The two enhanced algorithms, which strive to establish a tighter lower bound, are able to estimate the reliability with the desired accuracy with a much smaller sample size. Finally, between the two enhanced algorithms, the lower bound given by MIXED_{KK} is combinatorial in nature and is thus easier to compute than the one by $\text{MIXED}_{\text{sample}}$, which is based on importance sampling. The difference is reflected in the overall shorter running time by the algorithm MIXED_{KK}.

Finally, we compare the failure polynomials estimated by algorithm $\text{MIXED}_{\text{sample}}$ with the exact failure polynomials computed by the enumeration method ENUM. Fig. 4 shows the accuracy results on two sets of failure polynomials, with theoretical relative error bounds $\epsilon = 0.01$ and 0.05. For each set of failure polynomials, we compute the maximum relative error (from the actual polynomial) among them for various values of p. Therefore, each curve shows the upper envelope of relative errors by the failure polynomials. In both cases, the relative error is much smaller than the theoretical guarantee. This is because by using a lower bound for ρ_i, the algorithm oversamples in each Monte Carlo approximation for N_i. In addition, the errors for the N_i estimates are independent and may cancel out each other. Therefore, in practice, the algorithm would provide much better estimates than theoretically guaranteed.

VII. Estimating Cross-Layer Reliability With Absolute Error

We have considered computing *relative* approximation for the failure polynomial $F(p)$. However, in certain contexts, it may

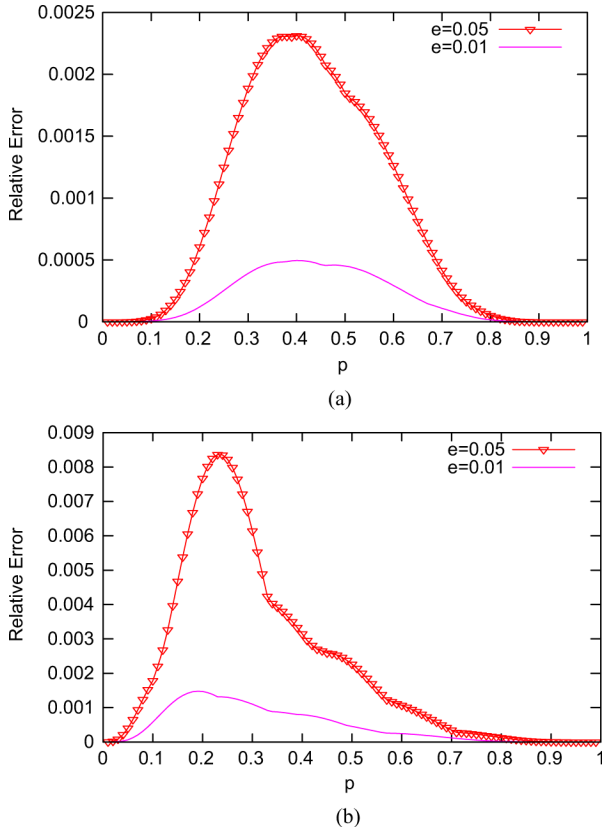


Fig. 4. Relative error of the failure polynomial approximation. (a) NJLATA. (b) NSFNET.

make sense to describe the error in absolute terms. A function $\hat{F}(p)$ is ϵ-absolute-approximate to $F(p)$ if

$$|\hat{F}(p) - F(p)| \leq \epsilon.$$

For example, if our goal is to design a network with a certain reliability target (say five 9's), it is sufficient to present a network whose associated failure polynomial has absolute error in the order of 10^{-6}. Constructing a failure polynomial with such relative error, however, may be overly stringent.

A function that is ϵ-approximate to $F(p)$ immediately implies that it is ϵ-absolute-approximate. As it turns out, using a similar approach of probabilistically estimating each N_i requires a much smaller number of samples to achieve ϵ-absolute accuracy. The total number of iterations required to compute an ϵ-absolute-approximation for $F(p)$ with high probability is $O(m \log m)$, in contrast to $O(m^{d+1} \log m)$ in the case of ϵ-approximation.

The intuition behind the difference is that computing an ϵ-approximation for N_i is difficult when the density ρ_i is small. However, in that case, the absolute contribution of the term $N_i p^i (1-p)^{m-i} = \rho_i \binom{m}{i} p^i (1-p)^{m-i}$ will be small as well. Therefore, in this case, even a large relative error for N_i will only account for a small absolute error.

More precisely, by the Estimator Theorem, the Monte Carlo method yields an $(\frac{\epsilon}{\sqrt{\rho_i}}, \frac{\delta}{m+1})$-approximation for N_i with $\frac{4}{\epsilon^2} \ln \frac{2(m+1)}{\delta}$ samples. In other words, if we run the Monte Carlo method with $O(\log m)$ samples to estimate each N_i, we can obtain $\frac{\epsilon}{\sqrt{\rho_i}}$-approximations $\hat{N}_i$ for all N_i with probability at least $1 - \delta$. This implies

$$\left| \sum_{i=0}^{m} (\hat{N}_i - N_i) p^i (1-p)^{m-i} \right| \leq \sum_{i=0}^{m} \frac{\epsilon}{\sqrt{\rho_i}} N_i p^i (1-p)^{m-i}$$
$$\leq \epsilon \sum_{i=0}^{m} \binom{m}{i} p^i (1-p)^{m-i} = \epsilon.$$

This means that we can compute ϵ-absolute-approximation for the failure polynomial $F(p)$ with high probability with a total of $O(m \log m)$ iterations. Unlike the case for ϵ-approximation, the number of iterations is independent of the Min Cross Layer Cut value d. This makes the method efficient even in the settings where d can be large.

VIII. Lightpath Routing Algorithms for Reliability Maximization

As illustrated in Section III-A, lightpath routing in a layered network plays an important role in the reliability. Therefore, an important question in reliable layered network design is to find the lightpath routing that provides maximum reliability given physical and logical topologies. However, as we have seen in Fig. 1, different values of link failure probability p often lead to different optimal lightpath routings. Hence, the reliability maximization problem is well defined only when the failure probability p is specified.

An important scenario to consider is the case where the physical link failure probability p is very small, as is the case for fiber networks. Theorem 9 describes an important property of optimal lightpath routings for this scenario. Specifically, when p is sufficiently small, the lightpath routing with the maximum MCLC value achieves maximum reliability over this low-failure-probability regime.

Theorem 9: Given two lightpath routings 1 and 2 for a given pair of physical and logical topologies, let

$$F_1(p) = \sum_{i=d}^{m} N_i p^i (1-p)^{m-i}$$

and

$$F_2(p) = \sum_{i=c}^{m} M_i p^i (1-p)^{m-i}$$

be the failure polynomials for the two lightpath routings, respectively. In addition, let c and d be their respective MCLC values. Assume $d > c$. Then, there exists a positive number p_0 such that $F_1(p) < F_2(p)$ for $p < p_0$. In particular

$$p_0 = \frac{M_c}{\binom{m}{c+1} + \frac{c}{c+1} M_c}.$$

Proof: See Appendix C. ∎

Therefore, under the lower failure regime, maximizing the reliability of a layered network is equivalent to maximizing its MCLC. As a result, we can formulate the reliability maximization problem using the MCLC as the objective function. This optimization criteria for layered network design is consistent with the deterministic failure setting studied in [26], which discussed

the intractability of this optimization problem and proposed several heuristics based on multicommodity flows that are shown to archive much better reliability performance than existing algorithms. It is therefore interesting to evaluate the performance of these algorithms under the random failure setting. A brief description of the three lightpath routing algorithms we will consider follows.

A. Description of Lightpath Routing Algorithms

1) Multicommodity Flow ($\mathsf{MCF}_{\mathsf{MinCut}}$)*:* The multicommodity flow lightpath routing algorithm $\mathsf{MCF}_{\mathsf{MinCut}}$ can be formulated as an integer linear program (ILP)

$$\begin{aligned}
&\mathsf{MCF}_{\mathsf{MinCut}}: \quad \text{Minimize } \rho, \quad \text{subject to:}\\
&\rho \geq \sum_{(s,t)\in E_{\mathrm{L}}} w(s,t) f_{ij}^{st} \qquad \forall (i,j) \in E_{\mathrm{P}}\\
&f_{ij}^{st} \in \{0,1\}\\
&\{(i,j): f_{ij}^{st} = 1\} \text{ forms an } (s,t) \text{ path in } G_{\mathrm{P}} \qquad \forall (s,t) \in E_{\mathrm{L}}
\end{aligned}$$

where $w(s,t)$ is the weight assigned to logical link (s,t). The optimal lightpath routing under this algorithm is determined by the weights $w(s,t)$. For example, if $w(s,t)$ is set to 1 for all logical links, the above formulation will minimize the number of logical links that traverse the same fiber. In other words, this uniform weight function treats each logical link equally and seeks to minimize the impact of a single physical link failure on the number of disconnected logical links. However, the connectivity is not well captured under this function since the logical network may remain connected even when a large number of logical links fail. In order to better account for the connectivity, we use the weight function $w(s,t) = \frac{1}{\mathrm{MinCut}_{\mathrm{L}}(s,t)}$, where $\mathrm{MinCut}_{\mathrm{L}}(s,t)$ is the size of the min-cut between nodes s and t in the logical topology. Intuitively, this weight function attempts to minimize the *impact* of a single-fiber failure to the logical connectivity, where impact is defined to be the total sum of weight of the logical links that traverse the fiber. Since the weight is defined to be $\frac{1}{\mathrm{MinCut}_{\mathrm{L}}(s,t)}$, a logical link that belongs to a small cut will contribute more weight than a logical link in a large cut. Presumably, if a lightpath routing ensures that any single-fiber failure has a small impact to the logical connectivity, a better MCLC will be achieved. In [26], it was shown that using the weight function $w(s,t) = \frac{1}{\mathrm{MinCut}_{\mathrm{L}}(s,t)}$ achieves better MCLC than using the uniform weight function $w(s,t) = 1$.

2) Enhanced Multicommodity Flow ($\mathsf{MCF}_{\mathsf{LF}}$)*:* The enhanced version of the multicommodity flow algorithm $\mathsf{MCF}_{\mathsf{LF}}$ tries to capture the impact of a single-fiber failure in much greater details. For each fiber failure, the algorithm measures its impact on each logical cut. As a result, the algorithm tries to minimize the maximum impact of a fiber failure among all possible logical cuts. It can be formulated by a similar ILP as follows:

$$\begin{aligned}
&\mathsf{MCF}_{\mathsf{LF}}: \quad \text{Minimize } \gamma, \quad \text{subject to:}\\
&\gamma |\delta(S)| \geq \sum_{(s,t)\in\delta(S)} f_{ij}^{st} \qquad \forall (i,j) \in E_{\mathrm{P}}, S \subset V_{\mathrm{L}}\\
&f_{ij}^{st} \in \{0,1\}\\
&\{f_{ij}^{st} : (i,j) \in E_{\mathrm{P}}\} \text{ forms an } (s,t) \text{ path} \qquad \forall (s,t) \in E_{\mathrm{L}}
\end{aligned}$$

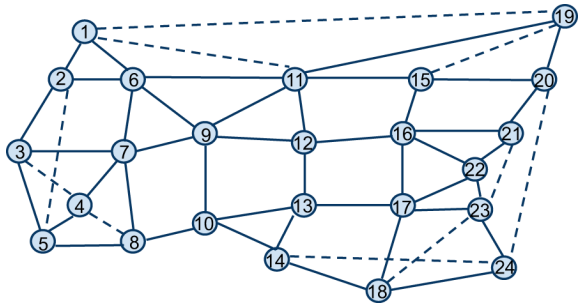


Fig. 5. Augmented USIP network (the dashed lines are the new links).

where $\delta(S)$ is the cut set of S, i.e., the set of logical links that have exactly one endpoint in S. Therefore, each constraint in $\mathsf{MCF}_{\mathsf{LF}}$ captures the fraction of logical links in the cut $\delta(S)$ that will be disconnected by the failure of fiber (i,j), and the formulation will try to minimize the maximum fraction disconnected among all possible fiber failures. In order to circumvent the issue of solving an integer program with a large number of constraints, the lightpath routing is obtained based on randomized rounding over the optimal solution to the linear relaxation of the formulation, as discussed in greater details in [26]. This enhanced algorithm is shown to achieve better MCLC performance compared to $\mathsf{MCF}_{\mathsf{MinCut}}$ [26].

3) Survivable Lightpath Routing (SURVIVE)*:* The survivable lightpath routing algorithm SURVIVE, proposed in [31], was used as the benchmark to evaluate the reliability performance of the two lightpath routing algorithms based on multicommodity flows. The algorithm tries to find a lightpath routing that keeps the logical topology connected under any single physical link failure. In other words, the objective is to find a lightpath routing with MCLC at least two. However, it makes no attempt to maximize the MCLC value as the previous two algorithms do.

B. Simulation Results

We study the reliability performance of the lightpath routing algorithms in this section. We used the NSFNET [Fig. 3(b)] and USIP networks [Fig. 5] as the physical topologies, both augmented to connectivity 4 so that a wider range of MCLC values is possible. For the NSFNET, a set of 350 random logical topologies with size from 6 to 12 nodes is generated, whereas for the USIP, a set of 500 random logical topologies with size from 6 to 15 nodes is generated. For each pair of physical and logical topologies, we compute the approximate failure polynomials for the lightpath routings generated by the three lightpath routing algorithms, using the reliability estimation algorithm proposed in Section V.

Fig. 6 shows the cumulative distributions of reliability for the lightpath routings generated by the three algorithms, with $p = 0.1$. For both physical topologies, the multicommodity flow algorithms $\mathsf{MCF}_{\mathsf{MinCut}}$ and $\mathsf{MCF}_{\mathsf{LF}}$, which try to maximize the MCLC value, are able to generate more lightpath routings with higher reliability than the algorithm SURVIVE. In particular, the enhanced version $\mathsf{MCF}_{\mathsf{LF}}$, which is shown to archive the best MCLC performance, also achieves the best reliability in this randomized setting. For small p, the term $N_d p^d (1-p)^{m-d}$,

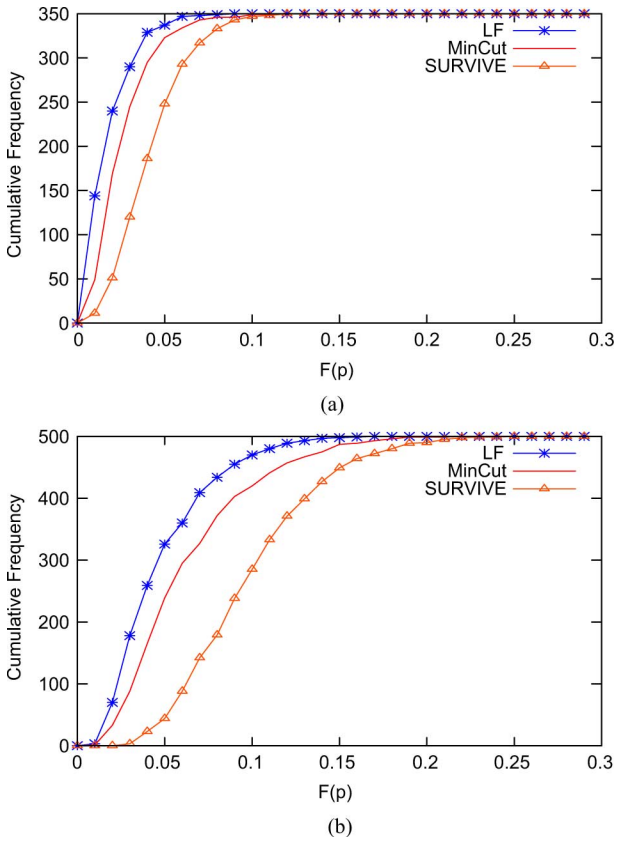


Fig. 6. Reliability cumulative distribution function (cdf) for the lightpath routing algorithms: $p = 0.1$. (a) NSFNET. (b) USIP.

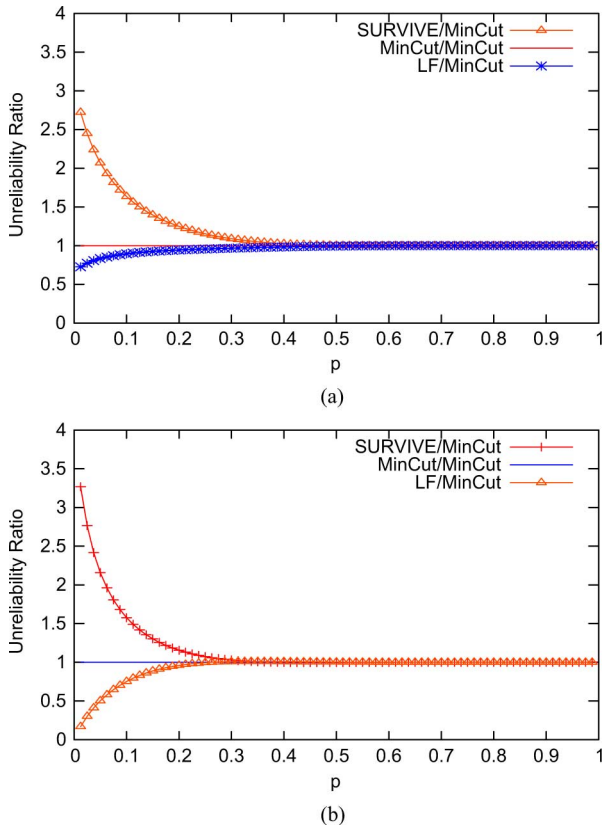


Fig. 7. Ratio of average failure probabilities among different algorithms. (a) NSFNET. (b) USIP.

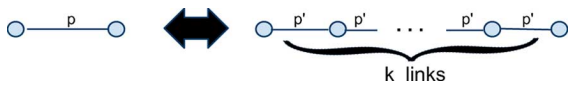


Fig. 8. Physical link with failure probability p is equivalent to $k = \log(1-p)/\log(1-p')$ physical links in series with failure probability p'.

where d is the Min Cross Layer Cut, dominates the failure polynomial. Therefore, maximizing d has the effect of maximizing the cross-layer reliability.

The effect of the lightpath routing and link failure probability p on the cross-layer reliability is further illustrated by Fig. 7, which plots the ratio of average failure probabilities of the lightpath routings generated by the three algorithms, using $\mathsf{MCF}_{\mathsf{MinCut}}$ as the baseline. When p is small, the lightpath routing algorithms with better MCLC performance also achieve better reliability performance. However, as p increases, the distinction becomes much less clear, primarily because the unreliability for any lightpath routing becomes close to 1 for large p.

IX. Extensions to the Failure Model

In this section, we present a few extensions to the failure model and discuss the application of the reliability estimation method to these extensions.

A. Nonuniform Failure Probabilities

In the nonuniform physical link failure model, each physical link (i,j) fails with probability p_{ij}. The physical topology can be approximated by replacing each physical link (i,j) by $k = \text{round}\left(\frac{\log(1-p_{ij})}{\log(1-p')}\right)$ physical links in series, where round() is the rounding function and p' is a constant that represents the link failure probability of the transformed network (Fig. 8). In this case, the probability that none of the replacements for (i,j) fails equals

$$(1-p')^k = (1-p')^{\frac{\log(1-p_{ij})}{\log(1-p')}}(1-p')^{\epsilon} = (1-p_{ij})(1-p')^{\epsilon}$$

where $|\epsilon| = \left|\text{round}\left(\frac{\log(1-p_{ij})}{\log(1-p')}\right) - \left(\frac{\log(1-p_{ij})}{\log(1-p')}\right)\right| \leq 0.5$. Therefore, this probability can be made arbitrarily close to $1-p_{ij}$ by choosing a sufficiently small p', with the tradeoff being a larger number of new links. In this case, the lightpath routing can then be modified such that a logical link originally using (i,j) is now routed over its replacements. This gives us an equivalent layered network where every physical link fails independently with probability p'.

B. Random Node Failures

The reliability estimation method can be extended to a model where each physical link fails with probability p and each physical node fails with probability q. We can model a network state as the set of failed physical nodes and links, and a logical link will fail if any of the physical nodes and links it uses fail. In this case, a cross-layer cut is a set of physical nodes and links whose

failures would cause the logical topology to be disconnected. The reliability of the layered network can then be expressed as follows:

$$\sum_{i=0}^{m}\sum_{j=0}^{n} N_{ij} p^i (1-p)^{m-i} q^j (1-q)^{n-j}$$

where m, n are the numbers of physical links and nodes, respectively, and N_{ij} is the number of cross-layer cuts with i failed physical links and j failed physical nodes. Then, we can estimate the reliability in a similar fashion by approximating each N_{ij} separately via the Monte Carlo method. To estimate N_{ij}, network states with i fibers and j nodes will be uniformly sampled. The methods in Sections IV-B and V-A to establish lower bounds on N_i can be extended to establish lower bounds on N_{ij} based on a similar observation in this setting that any network state that contains a cross-layer cut is also a cross-layer cut.

X. Conclusion

We considered network reliability in multilayer networks. In this setting, logical link failures can be correlated even if physical links fail independently. Hence, conventional estimation methods that assume particular topologies, independent failures, and network parameters could not be used for our problem. To that end, we developed a Monte Carlo simulation-based estimation algorithm that approximates cross-layer reliability with high probability. We first extended the classical polynomial expression for reliability to multilayer networks. Our algorithm approximates the failure polynomial by estimating the values of its coefficients. The advantages of our approach are twofold. First, it does not require resampling for different values of link failure probability p. Second, with a polynomial number of iterations, it guarantees the accuracy of estimation with high probability. We also observed through the polynomial expression that lightpath routings that maximize the MCLC can perform very well in terms of reliability. This observation led to the development of lightpath routing algorithms that attempt to maximize reliability.

While sampling failure states, our estimation algorithm naturally reveals the vulnerable parts of the network or lightpath routing. This information could be used to enhance the current lightpath routing. Therefore, future directions include the use our estimation algorithm for improving the lightpath routing, and the development of a new lightpath routing algorithm that maximizes the cross-layer reliability.

Appendix A
Estimating Lower Bound for ρ_i With Importance Sampling

As seen in Section V, given a set of cross-layer cuts $\mathcal{F}$, the value $\frac{|\partial_i(\mathcal{F})|}{\binom{m}{i}}$ gives a lower bound for ρ_i. We will discuss how to estimate the size of $\partial_i(\mathcal{F}) = \cup_{C_j \in \mathcal{F}} \partial_i(C_j)$ probabilistically.

Computing the value of $|\partial_i(\mathcal{F})|$ can be formulated as the Union of Sets Problem [30], where Monte Carlo method exists to estimate the size of $|\partial_i(\mathcal{F})|$ using the technique of importance sampling. Here, we define the ground set U to be $\{(S,j) : j \in \{1,\ldots,|\mathcal{F}|\}, S \in \partial_i(C_j)\}$, and the events of interest G to be $\{(S,j) : S \in \partial_i(\mathcal{F}), j = \min\{k : S \in \partial_i(C_k)\}\}$.

In other words, the ground set U represents a multiset where each set S in $\partial_i(\mathcal{F})$ is represented k times in U, where k is the number of elements in $\mathcal{F}$ that are subsets of S. On the other hand, each set S in $\partial_i(\mathcal{F})$ is represented by exactly one element (S,j) in G, where C_j is the first element in $\mathcal{F}$ that is a subset of S. As a result, for each $S \in \partial_i(\mathcal{F})$, $|\{(T,j) \in U : T = S\}| \le |\mathcal{F}|$, and $|\{(T,j) \in G : T = S\}| = 1$. It immediately follows that

$$|G| = |\partial_i(\mathcal{F})|$$

and

$$\frac{|G|}{|U|} \ge \frac{1}{|\mathcal{F}|}.$$

Therefore, by the Estimator Theorem, if we sample from U uniformly at random for T times, where

$$T = \frac{4|\mathcal{F}|}{\epsilon_{lb}^2} \ln\frac{2}{\delta_{lb}} \ge \frac{4}{\epsilon_{lb}^2 \frac{|G|}{|U|}} \ln\frac{2}{\delta_{lb}}$$

the Monte Carlo method will yield an ϵ_{lb}-approximation for $|G|$, which is equal to $|\partial_i(\mathcal{F})|$, with probability at least $1 - \delta_{lb}$.

Finally, the sample space U can be sampled uniformly at random as follows.

1) Select an element j from $\{1,\ldots,|\mathcal{F}|\}$, where the probability of selecting j is $\frac{|\partial_i(C_j)|}{\sum_{C_k \in \mathcal{F}} |\partial_i(C_k)|}$. Note that $|\partial_i(C_j)| = \binom{m-|C_j|}{i-|C_j|}$, which can be computed easily.
2) Given the selected value j, pick a set $S \in \partial_i(C_j)$ uniformly at random.

The probability of selecting each element $(S,j) \in U$ is therefore

$$\frac{|\partial_i(C_j)|}{\sum_{C_k \in \mathcal{F}} |\partial_i(C_k)|} \cdot \frac{1}{|\partial_i(C_j)|} = \frac{1}{\sum_{C_k \in \mathcal{F}} |\partial_k(C_k)|} = \frac{1}{|U|}.$$

This gives us a method to establish a probabilistic lower bound $\hat{\rho}_i$ for ρ_i.

Appendix B
Proof of Theorem 8

Let $[m] = \{1,\ldots,m\}$. Let $\mathcal{H}_i^m = \{S \subseteq [m] : |S| = i\}$ be a family of subsets of $[m]$ with size i, and let $\mathcal{H}_i^m(k)$ be the first k subsets in $\mathcal{H}_i^m$ under the lexicographical ordering. In addition, for any family $\mathcal{F}$ of subsets of $[m]$ and for any $j > i$, let $\partial_j^m(\mathcal{F})$ be the jth upper shadow of $\mathcal{F}$ over $[m]$. Theorem 8 states the following.

Theorem 8: For $i < j \le m$ and $1 \le k \le \binom{m}{i}$, let $w = \max\{0 \le r < i : \binom{m-r}{i-r} \ge k\}$. Also, let $t = m-(w+1)$, $u = j-(w+1)$ and $v = i-(w+1)$. Then

$$|\partial_j^m(\mathcal{H}_i^m(k))| = \begin{cases} \binom{m-i}{j-i}, & \text{if } k = 1 \\ \binom{t}{u} + |\partial_{u+1}^t(\mathcal{H}_{v+1}^t(k - \binom{t}{v}))|, & \text{otherwise.} \end{cases}$$

The case for $k = 1$ follows from the fact that for a set with size i, it has $\binom{m-i}{j-i}$ supersets with size j. We will prove the case where $k > 1$.

Let S be the lexicographically largest element in $\mathcal{H}_i^m(k)$. We first prove the following lemma.

Lemma 1: $[w] \subseteq S$ and $w+1 \notin S$.

Proof: Suppose the lemma is not true. We have the following two cases.

1) S does not contain some element $e \in [w]$. In this case, all subsets of $\mathcal{H}_i^m$ that contains $[w]$ are lexicographically smaller than S and thus belong to $\mathcal{H}_i^m(k)$. Therefore, $k = |\mathcal{H}_i^m(k)| > \binom{m-w}{i-w}$. This contradicts with the fact that $\binom{m-w}{i-w} \geq k$.
2) S contains $[w+1]$. Thus, any set $T \in \mathcal{H}_i^m$ that does not contain $[w+1]$ is lexicographically greater than S and, therefore, cannot be in $\mathcal{H}_i^m(k)$. As a result, $k = |\mathcal{H}_i^m(k)| \leq \binom{m-(w+1)}{i-(w+1)}$. However, by definition of w, we have $\binom{m-(w+1)}{i-(w+1)} < k$, which is a contradiction.

■

Corollary 2: All elements in $\mathcal{H}_i^m(k)$ must contain $[w]$.

Proof: Any element in $\mathcal{H}_i^m(k)$ must be lexicographically at most S and, therefore, must contain $[w]$. ■

Corollary 3: All elements in $\mathcal{H}_i^m$ that contain $[w+1]$ are in $\mathcal{H}_i^m(k)$.

Proof: Any element that contains $[w+1]$ is lexicographically smaller than S and, therefore, belongs to $\mathcal{H}_i^m(k)$. ■

We now partition the family $\mathcal{H}_i^m(k)$ into two subfamilies:

- $\mathcal{H}_i^m(k)^+ := \{T \in \mathcal{H}_i^m(k) : w+1 \in T\}$;
- $\mathcal{H}_i^m(k)^- := \{T \in \mathcal{H}_i^m(k) : w+1 \notin T\}$.

As a result of Corollaries 2 and 3, $\mathcal{H}_i^m(k)^+$ consists of all $\binom{m-(w+1)}{i-(w+1)}$ elements in $\mathcal{H}_i^m$ that contain $[w+1]$, and $\mathcal{H}_i^m(k)^-$ consists of the next $k - \binom{m-(w+1)}{i-(w+1)}$ elements in the lexicographical order. We define a bijection g_w on $\mathcal{H}_i^m(k)^-$ as follows:

$$g_w(T) := \{e-(w+1) : e \in T-[w]\} \qquad \forall T \in \mathcal{H}_i^m(k)^-. \quad (6)$$

In other words, for any $T \in \mathcal{H}_i^m(k)^-$, we construct $g_w(T)$ by first removing the common subset $[w]$ from T and then subtracting each remaining element by $w+1$. As a result, each $g_w(T)$ is a subset of $[m-(w+1)]$. The image $g_w(\mathcal{H}_i^m(k)^-)$ consists of the first $k - \binom{m-(w+1)}{i-(w+1)}$ subsets of $[m-(w+1)]$ size $i-w$ in lexicographical order. In other words, we have

$$g_w(\mathcal{H}_i^m(k)^-) = \mathcal{H}_{i-w}^{m-(w+1)}\left(k - \binom{m-(w+1)}{i-(w+1)}\right). \quad (7)$$

Now, consider $\partial_j^m(\mathcal{H}_i^m(k))$, the jth upper shadow over $[m]$ for $\mathcal{H}_i^m(k)$. As a result of Corollary 2, all elements in $\partial_j^m(\mathcal{H}_i^m(k))$ must contain $[w]$. We can therefore partition $\partial_j^m(\mathcal{H}_i^m(k))$ in a similar fashion:

- $\partial_j^m(\mathcal{H}_i^m(k))^+ := \{T \in \partial_j^m(\mathcal{H}_i^m(k)) : w+1 \in T\}$;
- $\partial_j^m(\mathcal{H}_i^m(k))^- := \{T \in \partial_j^m(\mathcal{H}_i^m(k)) : w+1 \notin T\}$.

We now prove the following properties of $\partial_j^m(\mathcal{H}_i^m(k))^+$ and $\partial_j^m(\mathcal{H}_i^m(k))^-$, which allow us to express the cardinality of the upper shadow in Theorem 8.

Lemma 2: $\partial_j^m(\mathcal{H}_i^m(k))^+ = \{T \in \mathcal{H}_j^m : [w+1] \subset T\}$.

Proof: Every element T in $\partial_j^m(\mathcal{H}_i^m(k))^+$ must contain $[w]$ by Corollary 2 and $w+1$ by definition. Therefore, T must contain $[w+1]$. In addition, for any element T in $\mathcal{H}_j^m$ that contains $[w+1]$, let U be the set with the i smallest elements in T. Since $i \geq w+1$, U contains $[w+1]$ and is in $\mathcal{H}_i^m(k)$ by Corollary 3. As a result, the subset T, being a superset of U, is in the jth upper shadow of $\mathcal{H}_i^m(k)$. ■

Corollary 4: $|\partial_j^m(\mathcal{H}_i^m(k))^+| = \binom{m-(w+1)}{j-(w+1)}$.

Lemma 3:

$$g_w(\partial_j^m(\mathcal{H}_i^m(k))^-) = \partial_{j-w}^{m-(w+1)}(g_w(\mathcal{H}_i^m(k)^-)).$$

Proof: For any element $T \in \partial_j^m(\mathcal{H}_i^m(k))^-$, there must exist an element $U \in \mathcal{H}_i^m(k)$ such that $U \subset T$. Since $w+1 \notin T$, it follows that $w+1 \notin U$, which implies $U \in \mathcal{H}_i^m(k)^-$. By applying the same bijection g_w to $\partial_j^m(\mathcal{H}_i^m(k))^-$, $g_w(T)$ is a subset of $[m-(w+1)]$ with size $j-w$ and is a superset of $g_w(U)$. In other words

$$g_w(\partial_j^m(\mathcal{H}_i^m(k))^-) \subseteq \partial_{j-w}^{m-(w+1)}(g_w(\mathcal{H}_i^m(k)^-)).$$

Now, given $T \in \partial_{j-w}^{m-(w+1)}(g_w(\mathcal{H}_i^m(k)^-))$, there exists $U \in g_w(\mathcal{H}_i^m(k)^-)$ such that $U \subset T$. It follows that $g_w^{-1}(U) \subset g_w^{-1}(T)$. Since $g_w^{-1}(U) \in \mathcal{H}_i^m(k)^-$, it follows that $g_w^{-1}(T) \in \partial_j^m(\mathcal{H}_i^m(k)^-)$. Therefore, $T \in g_w(\partial_j^m(\mathcal{H}_i^m(k)^-))$, which means

$$g_w(\partial_j^m(\mathcal{H}_i^m(k))^-) \supseteq \partial_{j-w}^{m-(w+1)}(g_w(\mathcal{H}_i^m(k)^-))$$

which proves the lemma. ■

Corollary 5:

$$|\partial_j^m(\mathcal{H}_i^m(k))^-| = \left|\partial_{j-w}^{m-(w+1)}\left(\mathcal{H}_{i-w}^{m-(w+1)}\left(k - \binom{m-(w+1)}{i-(w+1)}\right)\right)\right|.$$

Proof:

$$\begin{aligned}|\partial_j^m(\mathcal{H}_i^m(k))^-| &= |g_w(\partial_j^m(\mathcal{H}_i^m(k))^-)| \\ &= |\partial_{j-w}^{m-(w+1)}(g_w(\mathcal{H}_i^m(k)^-))| \\ &= \left|\partial_{j-w}^{m-(w+1)}\left(\mathcal{H}_{i-w}^{m-(w+1)}\left(k - \binom{m-(w+1)}{i-(w+1)}\right)\right)\right|.\end{aligned}$$

The second equality is due to shadowneg, and the third equality is due to (7). ■

The expression for $|\partial_j^m(\mathcal{H}_i^m(k))|$ for $k > 1$ follows immediately from Corollaries 4 and 5.

Appendix C
Proof of Theorem 9

We note that there is a similar result in the context of single-layer network reliability [22], however our result is more general in that it reveals the impact of the difference of MCLC values more explicitly. The following three lemmas will be used to compute p_0. First, recall that d and c are the size of MCLC under lightpath routings 1 and 2, respectively. Also, the values N_i and M_i are the coefficients in its failure polynomial.

Lemma 4: For $i \geq d$, $\frac{N_{i+1}}{\binom{m}{i+1}} \geq \frac{N_i}{\binom{m}{i}}$.

Proof: If i is less than the Min Cross Layer Cut value d, then $N_i = 0$, and the inequality holds trivially. Thus, we assume $i \geq d$. Let $\mathcal{H}_i$ and $\mathcal{H}_{i+1}$ be the sets of cross-layer cuts of size i and $i+1$, respectively. Define the set $\mathcal{K}$ to be $\{(X,Y)|X \in \mathcal{H}_i, Y \in \mathcal{H}_{i+1}, X \subset Y\}$. Since any superset of a cross-layer cut

is also a cross-layer cut, each cross-layer cut in $\mathcal{H}_i$ is a subset of exactly $m-i$ cross-layer cuts in $\mathcal{H}_{i+1}$. Therefore, we have

$$|\mathcal{K}| = N_i(m-i). \tag{8}$$

On the other hand, each cross-layer cut in $\mathcal{H}_{i+1}$ is a superset of at most $i+1$ cross-layer cuts in $\mathcal{H}_i$, which implies

$$|\mathcal{K}| \le N_{i+i}(i+1). \tag{9}$$

The theorem follows immediately from (8) and (9). ∎

Corollary 6: For any $0 \le d \le i \le m$, $N_i \ge \frac{\binom{m}{i}N_d}{\binom{m}{d}}$.

Proof:

$$\frac{N_i}{N_d} = \prod_{j=d+1}^{i} \frac{N_j}{N_{j-1}} \ge \prod_{j=d+1}^{i} \frac{\binom{m}{j}}{\binom{m}{j-1}} = \frac{\binom{m}{i}}{\binom{m}{d}}.$$

∎

We will also use the following result to bound the tail probability of the binomial distribution.

Lemma 5 [32]: For $r > mp$, $\sum_{i=r}^{m}\binom{m}{i}p^i(1-p)^{m-i} \le \binom{m}{r}p^r(1-p)^{m-r}\cdot\frac{r(1-p)}{r-mp}$.

Now, we provide the value of p_0 such that for all $p < p_0$, the reliability of lightpath routing 1 is no less than the reliability of lightpath routing 2. We first establish the following bounds for $F_1(p)$ and $F_2(p)$.

Lemma 6: For all $p \in [0,1]$, we have the following.

1)

$$U_1(p) := \sum_{i=0}^{c} N_i p^i(1-p)^{m-i} + \sum_{i=c+1}^{m}\binom{m}{i}p^i(1-p)^{m-i} \ge F_1(p).$$

2)

$$L_2(p) := \sum_{i=0}^{c} M_i p^i(1-p)^{m-i} + \sum_{i=c+1}^{m}\frac{\binom{m}{i}M_c}{\binom{m}{c}}p^i(1-p)^{m-i} \le F_2(p).$$

Proof:

$$\begin{aligned} U_1(p) &= \sum_{i=0}^{c} N_i p^i(1-p)^{m-i} + \sum_{i=c+1}^{m}\binom{m}{i}p^i(1-p)^{m-i} \\ &\ge \sum_{i=0}^{c} N_i p^i(1-p)^{m-i} + \sum_{i=c+1}^{m} N_i p^i(1-p)^{m-i} \\ &= F_1(p). \end{aligned}$$

$$\begin{aligned} L_2(p) &= \sum_{i=0}^{c} M_i p^i(1-p)^{m-i} + \sum_{i=c+1}^{m}\frac{\binom{m}{i}M_c}{\binom{m}{c}}p^i(1-p)^{m-i} \\ &\le \sum_{i=0}^{c} M_i p^i(1-p)^{m-i} \\ &\quad + \sum_{i=c+1}^{m} M_i p^i(1-p)^{m-i}, \qquad \text{by Corollary 6} \\ &= F_2(p). \end{aligned}$$

Therefore, it suffices to find a p_0 such that for all $p < p_0, U_1(p) \le L_2(p)$. The following lemma provides such a p_0.

Lemma 7: Let $p_0 = \frac{M_c}{\binom{m}{c+1}+\frac{c}{c+1}M_c}$. Then, $U_1(p) \le L_2(p)$ for $p < p_0$.

Proof: If $p < p_0 = \frac{M_c}{\binom{m}{c+1}+\frac{c}{c+1}M_c} = \frac{M_c(c+1)}{\binom{m}{c}(m-c)+cM_c}$, we have

$$\frac{c+1}{p} > \frac{\binom{m}{c}(m-c)}{M_c} + c$$

which implies

$$\frac{(m-c)p}{c+1-mp} < \frac{M_c}{\binom{m}{c}-M_c}. \tag{10}$$

Now, note that

$$\begin{aligned} c+1 &> \frac{\binom{m}{c}(m-c)+cM_c}{M_c}\cdot p \\ &\ge \frac{M_c(m-c)+cM_c}{M_c}\cdot p = mp. \end{aligned}$$

Therefore, Lemma 5 is applicable for $r = c+1$. It follows that

$$\begin{aligned} &L_2(p) - U_1(p) \\ &= M_c p^c(1-p)^{m-c} - \left(1-\frac{M_c}{\binom{m}{c}}\right)\sum_{i=c+1}^{m}\binom{m}{i}p^i(1-p)^{m-i} \\ &\ge M_c p^c(1-p)^{m-c} \\ &\quad -\left(1-\frac{M_c}{\binom{m}{c}}\right)\binom{m}{c+1}p^{c+1}(1-p)^{m-(c+1)}\cdot\frac{(c+1)(1-p)}{(c+1)-mp}, \\ &\qquad \text{by Lemma 5} \\ &= p^c(1-p)^{m-c}\left(M_c - \frac{\left(\binom{m}{c}-M_c\right)(m-c)p}{(c+1)-mp}\right) \\ &> p^c(1-p)^{m-c}\left(M_c - \frac{\left(\binom{m}{c}-M_c\right)M_c}{\binom{m}{c}-M_c}\right), \\ &\qquad \text{by Equation (10)} \\ &= 0. \end{aligned}$$

As a result, for $p < p_0$, we have $F_1(p) \le U_1(p) < L_2(p) \le F_2(p)$, which means that the unreliability of lightpath routing 1 is always smaller when p is within that range. ∎

REFERENCES

[1] G. Iannaccone, C. n. Chuah, S. Bhattacharyya, and C. Diot, "Feasibility of IP restoration in a tier-1 backbone," *IEEE Netw.*, vol. 18, no. 2, pp. 13–19, Mar.–Apr. 2004.

[2] L. Valiant, "The complexity of enumeration and reliability problems," *SIAM J. Comput.*, vol. 8, no. 3, pp. 410–421, Aug. 1979.

[3] J. S. Provan and M. Ball, "The complexity of counting cuts and of computing the probability that a graph is connected," *SIAM J. Comput.*, vol. 12, no. 4, pp. 777–788, Nov. 1983.

[4] C. Tanguy, "Exact two-terminal reliability of directed networks," in *Proc. DRCN*, La Rochelle, France, Oct. 2007, pp. 1–8.

[5] A. Shooman and A. Kershenbaum, "Exact graph-reduction algorithms for network reliability analysis," in *Proc. IEEE GLOBECOM*, Dec. 1991, vol. 2, pp. 1412–1420.

[6] A. Rosenthal and D. Frisque, "Transformations for simplifying network reliability calculations," *Networks*, vol. 7, pp. 97–111, 1977.

[7] A. Shooman and A. Kershenbaum, "Methods for communication-network reliability analysis: Probabilistic graph reduction," in *Proc. Annu. Rel. Maintain. Symp.*, 1992, pp. 441–448.
[8] G. Lingner, T. Politof, and A. Satyanarayana, "A forbidden minor characterization and reliability of a class of partial 4-trees," *Networks*, vol. 25, pp. 139–146, 1995.
[9] T. Politof, A. Satyanarayana, and L. Tung, "An $O(n \cdot \log(n))$ algorithm to compute the all-terminal reliability of $(K_5, K_{2,2,2})$ free networks," *IEEE Trans. Rel.*, vol. 41, no. 4, pp. 512–517, Dec. 1992.
[10] M. Chari, T. Feo, and J. S. Provan, "The delta-wye approximation procedure for two-terminal reliability," *Oper. Res.*, vol. 44, no. 5, pp. 745–757, 1996.
[11] E. Neufeld and C. Colbourn, "The most reliable series-parallel networks," *Networks*, vol. 15, pp. 27–32, 1985.
[12] R. Karp and M. Luby, "Monte Carlo algorithms for planar multiterminal network reliability problems," *J. Complexity*, vol. 1, no. 1, pp. 45–64, 1977.
[13] D. Karger, "A randomized fully polynomial time approximation scheme for the all terminal reliability problem," *SIAM Rev.*, vol. 43, no. 3, pp. 499–522, 2001.
[14] G. Fishman, "A comparison of four Monte Carlo methods for estimating the probability of $s - t$ connectedness," *IEEE Trans. Rel.*, vol. R-35, no. 2, pp. 145–155, Jun. 1986.
[15] M. Ball, "Complexity of network reliability computations," *Networks*, vol. 10, pp. 153–165, 1980.
[16] T. Brecht, "Lower bounds for two-terminal network reliability," Master's thesis, Dept. Comput. Sci., Univ. Waterloo, Waterloo, ON, Canada, 1985.
[17] R. Van Sylke and H. Frank, "Network reliability analysis: Part I," *Networks*, vol. 1, pp. 279–290, 1972.
[18] H. AboElFotoh and C. Colbourn, "Computing 2-terminal reliability for radio-broadcast networks," *IEEE Trans. Rel.*, vol. 38, no. 5, pp. 538–555, Dec. 1989.
[19] C. Colbourn and D. Harms, "Bounding all-terminal reliability in computer networks," *Networks*, vol. 18, pp. 1–12, 1988.
[20] A. Ramanathan and C. Colbourn, "Counting almost minimum cutsets with reliability applications," *Math. Program.*, vol. 39, pp. 253–261, 1987.
[21] R. G. Addie and R. Taylor, "An algorithm for calculating the availability and mean time to repair for communication through a network," *Comput. Netw. ISDN Syst.*, vol. 20, pp. 109–114, 1989.
[22] D. Bauer, F. Boesch, C. Suffel, and R. Van Slyke, "On the validity of a reduction of reliable network design to a graph extremal problem," *IEEE Trans. Circuits Syst.*, vol. CAS-34, no. 12, pp. 1579–1581, Dec. 1987.
[23] L. Nel and C. Colbourn, "Combining Monte Carlo estimates and bounds for network reliability," *Networks*, vol. 20, pp. 277–298, 1990.
[24] M. Ball and J. S. Provan, "Bounds on the reliability polynomial for shellable independence systems," *SIAM J. Algebr. Discrete Methods*, vol. 3, pp. 166–181, 1982.
[25] K. Lee, H.-W. Lee, and E. Modiano, "Reliability in layered networks with random link failures," MIT, Cambridge, MA, 2009 [Online]. Available: http://web.mit.edu/kylee/www/reliability.pdf
[26] K. Lee, E. Modiano, and H.-W. Lee, "Cross-layer survivability in WDM-based networks," *IEEE Trans. Netw.*, vol. 19, no. 4, Aug. 2011, to be published.
[27] F. Boesch, "On unreliability polynomials and graph connectivity in reliable network synthesis," *J. Graph Theory*, vol. 10, no. 3, pp. 339–352, 1986.
[28] R. Motwani and P. Raghavan, *Randomized Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 1995.
[29] B. Bollobás, *Combinatorics: Set Systems, Hypergraphs, Families of Vectors, and Combinatorial Probability*. New York: Cambridge Univ. Press, 1986, p. 34.
[30] R. Karp and M. Luby, "Monte-carlo algorithms for enumeration and reliability problems," in *Proc. Annu. Symp. Found. Comput. Sci.*, Nov. 1983, pp. 56–64.
[31] E. Modiano and A. Narula-Tam, "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.
[32] W. Feller, *An Introduction to Probability Theory and Its Applications*. New York: Wiley, 1968.

Kayi Lee (S'99–M'05) received the B.S. degree in computer science and mathematics and the M.Eng. and Ph.D. degrees in computer science from the Massachusetts Institute of Technology (MIT), Cambridge, in 1999, 2000, and 2011, respectively.

He is currently a Software Engineer with Google, Inc., Cambridge, MA. His research interests include routing and network survivability.

Hyang-Won Lee (S'02–M'08) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2001, 2003, and 2007, respectively.

He is currently a Post-Doctoral Research Associate with the Massachusetts Institute of Technology, Cambridge. His research interests are in the areas of congestion control, wireless resource allocation, and network survivability.

Eytan Modiano (S'90–M'93–SM'00) received the B.S. degree in electrical engineering and computer science from the University of Connecticut, Storrs, in 1986, and the M.S. and Ph.D. degrees in electrical engineering, from the University of Maryland, College Park, in 1989 and 1992, respectively.

He was a Naval Research Laboratory Fellow between 1987 and 1992 and a National Research Council Post Doctoral Fellow from 1992 to 1993. Between 1993 and 1999, he was with the Massachusetts Institute of Technology (MIT) Lincoln Laboratory, Lexington, where he was a project leader for MIT Lincoln Laboratory's Next Generation Internet (NGI) project. Since 1999, he has been with the faculty of MIT, where he is a Professor with the Department of Aeronautics and Astronautics and the Laboratory for Information and Decision Systems (LIDS). His research is on communication networks and protocols with emphasis on satellite, wireless, and optical networks.

Dr. Modiano is currently an Associate Editor for the IEEE/ACM TRANSACTIONS ON NETWORKING. He has served as an Associate Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY, and as a Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS special issue on WDM network architectures, the *Computer Networks* special issue on broadband Internet access, the *Journal of Communications and Networks* special issue on wireless ad hoc networks, and the IEEE/OSA JOURNAL OF LIGHTWAVE TECHNOLOGY special issue on optical networks. He was the Technical Program Co-Chair for IEEE WiOpt 2006, IEEE INFOCOM 2007, and ACM MobiHoc 2007. He is the co-recipient of the SIGMETRICS 2006 Best Paper Award for the paper "Maximizing Throughput in Wireless Networks via Gossiping," and the WiOpt 2005 Best Student Paper Award for the paper "Minimum Energy Transmission Scheduling Subject to Deadline Constraints."