A Case for Using Service Availability to Characterize IP Backbone Topologies

Ram Keralapura, Adam Moerschell, Chen-Nee Chuah, Gianluca Iannaccone, and Supratik Bhattacharyya

Abstract: **Traditional service-level agreements (SLAs), defined by average delay or packet loss, often camouflage the instantaneous performance perceived by end-users. We define a set of metrics for** *service availability* **to quantify the performance of Internet protocol (IP) backbone networks and capture the impact of routing dynamics on packet forwarding. Given a network topology and its link weights, we propose a novel technique to compute the associated service availability by taking into account transient routing dynamics and operational conditions, such as border gateway protocol (BGP) table size and traffic distributions.**

Even though there are numerous models for characterizing topologies, none of them provide insights on the expected performance perceived by end customers. Our simulations show that the amount of service disruption experienced by *similar* **networks (i.e., with similar intrinsic properties such as average out-degree or network diameter) could be significantly different, making it imperative to use new metrics for characterizing networks. In the second part of the paper, we derive** *goodness factors* **based on service availability viewed from three perspectives: Ingress node (from one node to many destinations), link (traffic traversing a link), and network-wide (across all source-destination pairs). We show how goodness factors can be used in various applications and describe our numerical results.**

Index Terms: **Interior gateway protocol (IGP) routing dynamics, Internet protocol (IP) network goodness, service availability in IP networks.**

I. INTRODUCTION

Service-level agreements (SLAs) offered by today's Internet service providers (ISPs) are based on four metrics: End-to-end delay, packet loss, data delivery rate, and port availability. The first three metrics are usually computed network-wide and averaged over a relatively long period of time. For the fourth metric, the term "port" refers to the point at which a customer's link attaches to the edge of an ISP's network. Port availability therefore refers to the fraction of time this port is operational and measures a customer's physical connectivity to the ISP's network. None of these SLA metrics capture the ability of the network to carry customer traffic to Internet destinations at any point in time.

The main problem with the existing SLA specifications is that they do not capture the effect of instantaneous network conditions like failures and congestions. A recent study [1] shows that failures occur on a daily basis due to a variety of reasons (e.g., fiber cut, router hardware/software failures, and human errors) and can impact the quality of service (QoS) delivered to customers. When a link/node fails, all routers will independently compute a new path around the failure. At that time, routers may lack or have inconsistent forwarding information, resulting in packet drops or transient routing loops [2], [3]. However, not all failures impact the network equally. The failure of a critical backbone link carrying heavy traffic may be more detrimental than the failure of an access link connecting a single customer. Yet, these service degradations are camouflaged by the *average* parameters reported in current SLAs. Therefore, to *measure* Internet protocol (IP) network performance, it is essential to consider the network routing configuration and traffic pattern during link or node failures.

Reports from various tier-1 ISPs suggest that IP backbone networks are usually over-provisioned where the link utilization of backbone links is less than 50% of their total capacity [4], [5]. The reports also confirm that congestion due to link or router overload is a very rare event in backbone networks. During a link failure, event traffic on the failed link is rerouted and may congest links along alternate paths. However, such congestions are usually not significant for a single failure event. Heavy congestions may occur when there are multiple failures, but such events are relatively rare. Hence in our current work, we ignore the effect of congestions on network performance and only consider failures.

In this paper, we define a set of metrics for *service availability* of IP backbone networks that capture the impact of routing dynamics on packet forwarding. Instead of relying on active or passive measurements, we propose a methodology to *estimate* the service availability of a network in the presence of independent link failures. Specifically, given a topology (nodes, links, and connectivity) and routing information (link weights, link delays, and border gateway protocol (BGP) peering points), we are able to compute the potential impact on service due to link failures. To achieve this, we carefully model the factors identified in the measurement based study by Iannaccone *et al.* [6] that contribute to routing convergence. Convergence refers to the amount of time it takes for traffic forwarding to resume correctly on the backup path after a link failure. We wish to point out here that we focus mainly on single link failures for IP backbone networks since these are the dominant class of failures (i.e., over 70% of all failures in IP backbone networks) as observed by Markopoulou *et al.* [1].

We use the novel concept of service availability to evaluate known topologies, such as full-mesh, ring, and tier-1 ISP backbones. Our simulations show that the performance of a network

Manuscript received February 23, 2005; approved for publication by Min Young Chung, Division III Editor, October 11, 2005.

R. Keralapura, A. Moerschell, C.-N. Chuah are with the University of California, Davis, email: {rkeralapura, atmoerschell}@ucdavis.edu, chuah@ece.ucdavis.edu.

G. Iannaccone is with the Intel Research Labs, email: gianluca.iannaccone@intel.com.

S. Bhattacharyya is with the Sprint ATL, email: supratik@sprintlabs.com.

not only depends on routing dynamics, but also on various other factors like interior gateway protocol (IGP) link weight assignment and BGP prefix distribution. This brings out a necessity to identify new metrics that customers can use to differentiate networks. There have been many attempts to characterize the Internet topologies or to model their graph-theoretic properties [7]–[9]. The resulting models are useful for re-generating topologies that best model real networks, such as GT-ITM [10] and BRITE [11], but they do not provide any insights on the QoS that a particular network can provide.

Using the concept of service availability, we derive *goodness factors* based on three different perspectives: Ingress node (from one node to many destinations), link (traffic traversing a link), and network-wide (across all source-destination pairs). The goodness factors reveal how topologies with similar intrinsic graph properties, such as average out-degree or network diameter, do not necessarily offer the same level of service availability.

Finally, we describe several applications for the goodness factors in network planning and provisioning. For example, goodness from an ingress node perspective allows customers to choose the best place to connect to a network (or to choose among different providers), while link-based goodness helps an ISP to identify the set of critical links to be upgraded.

An earlier version of this work appeared in [12]. This journal paper improves that work and extends it with additional materials, including (i) use of a packet-level simulator (as opposed to a control-level simulator) to analyze service availability, (ii) incorporation of realistic failure models obtained from an ISP backbone network as reported in [1], and (iii) additional metrics and results that capture the service availability of IP backbone networks.

The rest of the paper is organized as follows. Section II identifies the importance of routing dynamics in estimating the end-to-end performance of a network and motivates this work. In Section III, we describe the proposed metrics and introduce the concept of service availability to characterize network topologies. We also define a set of network goodness factors. We discuss our numerical results in Section IV. We show the various applications of the goodness factors in Section V and conclude our paper in Section VI.

II. ANALYZING IMPACT OF ROUTING DYNAMICS

Intra-domain routing protocols, such as IS-IS [13] and OSPF [14], define how each node in the network responds to changes in the topology. Such protocols are also known as *link state protocols* where each node has complete knowledge of the network topology including all the links present in the network.

Upon detection of a link/node failure or a configuration change in the network, each node is responsible for disseminating the new topology description to all its neighboring nodes and recomputing the forwarding information in its own routing table. From the time of the failure or configuration change to the time all nodes have been informed of the change and have updated their routing tables, traffic disruptions (like packet drops and routing loops) are possible as the nodes may have an inconsistent view of the network.

We define the "convergence time" (CT) of a node due to a failure event in the network as the time taken by the node to update its routing and forwarding information in response to the failure. A node that does not have to update its routing or forwarding information has a convergence time of zero. The convergence time for any node n (that has to update its routing or forwarding information) due to a failure can be summarized as a combination of 3 components:

- *Detection time*: This is the time taken by the adjacent nodes to detect the failure. Today's IP routers provide several mechanisms to perform this function [15], but all of them are based only on local information exchanged between neighboring nodes. For example, *hello* messages at the IP layer or alarms at the optical layer. Hence, detection time represents a fixed price that is independent of the network topology or configuration.
- *Notification time*: This represents the time taken by the routing update to propagate through the network to reach n. In link state protocols, messages are flooded throughout the network. Therefore, the notification time strongly depends on the hop distance between n and the adjacent nodes. Each node along the forwarding route needs to process the message update before forwarding it, thus introducing a delay in the propagation of information.
- *Route computation and update time*: This is the time spent by node n to compute the new shortest path routing tree (that incorporates the failure information) and then update its forwarding information based on the new routing tree. The update procedure involves applying the changes to all the network prefixes that have been learnt via the BGP inter-domain protocol. The result of this computation is a forwarding table where each prefix is associated with a neighboring node as the *next hop*.

  The route computation and update time at any node heavily depends on the number of prefixes for which the next hop information needs to be changed. In turn, the number of prefixes to be updated not only depends on the location of the failure, but also on the distribution of prefixes to the next hops in the forwarding table. Indeed, the closer the failure occurs to a node, the larger will be the number of prefixes affected. Similarly, if a large number of prefixes share the same next hop node, a change in the topology close to that node will result in long route update time for all nodes in the network.

Based on the per-node CT, we define the "network convergence time" as the maximum value of CT among all the nodes in the network. This indicates the time at which all the nodes in the network have learned about the failure, updated their routing tables and have a consistent view of the network.

Service availability of a network depends on the convergence time. Providers attempt to increase the overall availability of their networks by reducing convergence time in order to speed up the recovery after a failure. As we described above, convergence time depends on (i) router technology used for failure detection, (ii) network topology, (iii) routing protocol configuration such as IGP weights and timers, and (iv) location of peering points with other networks that determines the distribution of network prefixes among egress nodes. Clearly, it is not

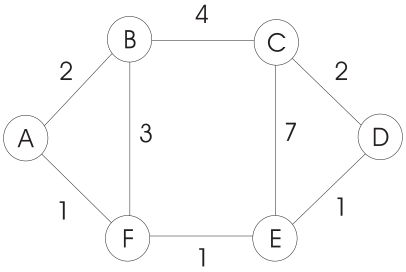

Fig. 1. Example of convergence time and service disruption.

possible to look at a subset of the above mentioned factors to derive the service availability of a network.

Consider the network illustrated in Fig. 1. The number on each link indicates the IGP link weight. Let node A be the traffic source and node D be the traffic sink. Consider the failure of link E–D. We assume that the nodes and the links have similar characteristics with a *detection time* of 500 ms, a *notification time* between neighboring nodes of 100 ms, and a node *route computation and update time* of 400 ms (except for node C for which we assume this to be 100 ms due to the fact that it does not have to change its route to reach D after the failure of link E–D). Consider the disruption observed for the traffic sent from node A to node D following the failure event at time $t_0 = 0$. Table 1 shows the routing events, changes in the traffic forwarding path, and service availability from node A to node D. In this example, we assume that a node notifies neighboring nodes only after it has completed the update of its own routing table. However, it is easy to verify that if the updates are sent before updating the routing table, the example yields similar results.

Interestingly, as the message update propagates across the network, the forwarding path from node A to node D changes four times. Some of these intermediate paths are valid thus restoring service between A and D, while some are not, causing packet drops (or *traffic black-hole*) and routing loops. For example, packets are dropped until node E has computed a new forwarding path to reach D and routing loops occur when nodes B and F have conflicting forwarding information.

Based on the example, we can derive some initial observations:

- The network convergence time provides a rough upper bound on service disruption time (i.e., the time for which service is not available). The service is not available when the packets cannot reach their destinations due to the lack of forwarding information. Given that traffic forwarding may resume even if all the nodes in the network have not updated their routing table, there may be a significant lack of correlation between network convergence and service availability. For example, in our illustration above, the cumulative time for which the service is not available amounts to 1.1 s while the network takes 1.9 s to converge. Therefore, network convergence time does not capture the entire routing dynamics.
- The network topology by itself does not help in understanding the service availability of a network. Additional information such as the location of peering points and size of routing tables needs to be considered in characterizing service availability. The time taken to update the routing table depends on the number of prefixes that are affected. Table 2 shows the impact of varying the number of prefixes that have D as the next hop node on convergence time and service availability. Even with the same topology and identical failure scenario, increasing route update time may lead to significant differences in service availability.

In the following section, we introduce a new topology metric that exploits the knowledge of routing dynamics to define and compare the *goodness* of topologies. It is based on the algorithm to compute the cumulative time for which service is not available (i.e., the service disruption time as presented in Table 1). The details of this algorithm are presented in Table 3.

III. SERVICE AVAILABILITY TO CAPTURE NETWORK ROUTING DYNAMICS

A. Service Availability

Service availability is an important concept for both ISPs and their customers. From an ISP's point-of-view, it is important to determine the overall service availability of its network so as to provide guarantees to all its customers. This is typically expressed as an average (or a summary) for all the customers connected to the network. It is also important for an ISP (and its customer) to understand the guarantees that are actually provided to customers connected to the network at a particular ingress node. Hence, we define service availability in three perspectives. The first perspective tries to capture service availability as seen by a customer connected to the ISP network at a given ingress node, while the other two perspectives try to capture service availability as seen by the ISP.

- *Ingress node perspective*: This is a measure of the network performance as seen by a particular ingress node where the traffic enters the network. It provides an insight about the level of service to expect when a customer connects to different ingress nodes of a network. It also helps an ISP to ensure that it can meet SLA specifications for customers connecting to different ingress nodes.
- *Link perspective*: The performance of a network should not heavily depend on the reliability of a few links in the network. In other words, the network should not have critical links whose failure results in serious performance degradation. Service availability from a link perspective evaluates the importance of various links for network performance.
- *Network perspective*: This measures the performance of the entire network from the perspective of an ISP. This is useful in designing a network to achieve high end-to-end performance.

We propose four metrics that capture routing dynamics in a network due to single link failures for analyzing service availability:

- *Service disruption time* (SD time): This represents the time for which service between a particular source-destination (OD) pair is disrupted. From the point of view of an ingress node, it indicates the loss in connectivity with all/some parts of the Internet due to a link failure.
- *Traffic disruption* (TD): This metric captures the total traffic disrupted between a particular source and destination node due to single link failures. The TD for a OD pair is computed

Table 1. Summary of routing events (with a route update time of 400 ms). Network convergence time = 1.9 s; service disruption time = 1.1 s.

Time	Event	Forwarding path (A–D)	Service from A to D	Notes
0 s	Failure of link E–D	A-F-E-D	No	
0.5 s	D, E: Failure detection	A-F-E-D	No	
0.9 s	D, E: Route update	A-F-E-C-D	Yes	Forwarding is restored
1.0 s	C, F: Notified of failure	A-F-E-C-D	Yes	
1.1 s	C: Route update	A-F-E-C-D	Yes	Path C to D is not affected
1.2 s	B: Notified of failure	A-F-E-C-D	Yes	
1.4 s	F: Route update	A-F-B-F-···	No	Routing loop B–F
1.5 s	A: Notified of failure	A-F-B-F-···	No	Routing loop B–F
1.6 s	B: Route update	A-F-B-C-D	Yes	
1.9 s	A: Route update	A-B-C-D	Yes	Network convergence

Table 2. Convergence time depends on routing update time in nodes.

Routing update time	100 ms	200 ms	300 ms	400 ms	500 ms	1000 ms
Convergence time	1.0 s	1.3 s	1.6 s	1.9 s	2.2 s	3.7 s
Time-service not available	0.8 s	0.9 s	1.0 s	1.1 s	1.2 s	1.7 s

Table 3. Algorithm to calculate service disruption time from node x to node y due to a single link failure.

Step 1: Initialize the *service disruption time*, $\phi_l(x, y)$, for the path from x to y due to the failure of the link l to 0, i.e., $\phi_l(x, y) = 0$. If the original path from x to y does not contain link l, then QUIT.

Step 2: Find the convergence time (CT) for each node and list the nodes in the increasing order of convergence time. Let the convergence time of the first node in the list be CT_1, second node be CT_2, and so on. In general, the convergence time for the n-th node in the list is CT_n. Note $CT_1 \leq CT_2 \leq CT_3 \leq \cdots \leq CT_n$. Set the *current node*, k, (which is the node number on the sorted list) to 0.

Step 3: Increment k to 1. Set $\phi_l(x, y) = CT_1$. At the time instant CT_1 after the failure event, find the path that a packet from the source node x, follows to reach the destination node y, taking into account that the first node in the list has converged and others have not. If the path has a routing loop or black-hole, then set $previousDisruption = true$, else set $previousDisruption = false$.

Step 4: Increment k by 1. At the time instant CT_k after the failure event, find the path that a packet from the source node x, follows to reach the destination node y, taking into account that the intermediate nodes might have converged or not.

Step 5: If the path that the packet follows does not contain the failed link l and has no routing loop, then set $previousDisruption = false$. Go to Step 7. Else go to Step 6.

Step 6: If the path that the packet follows contains the failed link l or has a routing loop, then the path from x to y is still disrupted. If $previousDisruption = false$, then do not update the *service disruption time* but set $previousDisruption = true$ else if $previousDisruption = true$, then update the *service disruption time* in the k-th iteration as, $\phi_l(x, y) = \phi_l(x, y) + CT_k - CT_{k-1}$.

Step 7: If there are more nodes in the list then go to Step 4. Else QUIT.

as the product of traffic rate between the OD pair and the service disruption time (as calculated in Table 3) for the OD pair. TD for an OD pair with no SD time (i.e., the failed link does not affect the OD pair) is 0. Similarly, TD for a OD pair that do not exchange any traffic is also 0.

Note that from the perspective of an ISP, TD is more important than SD time because of the fact that customers are usually compensated for the amount of traffic lost, irrespective of the duration for which the service is disrupted.

- *Number of delay parameter violations* (DV): In a well-designed network, the end-to-end delay along alternate paths found after a link failure is generally higher than the original path. We define *delay parameter* as the maximum end-to-end delay that can be tolerated by delay sensitive traffic in the network. If the end-to-end delay along the alternate path exceeds the delay parameter, then there is a delay parameter violation. DV measures the number of such delay parameter violations from the perspective of an ingress node due to single link failures. DV is calculated by first computing the alternate paths between all OD pairs after a link failure. The end-to-end delays along these alternate paths are then computed and compared with the delay parameter to determine if there is a delay parameter violation. The cumulative number of such violations from the perspective of a node results in the value for DV. Since backbone networks rarely experience congestions, we ignore the impact of queueing delays while calculating DV.

- *Number of OD pairs affected* (ODA): This metric denotes the number of source-destination pairs whose connectivity is affected by single link failures. The number of source-destination pairs affected by a failure does not indicate the magnitude of the failure either from the perspective of an ISP or its customer due to the elephant and mice phenomena in backbone network traffic. Although an ISP can use ODA as a metric for service availability, in the rest of the paper we do not consider this metric due to the fact that it ignores the actual amount of traffic that gets affected by the failure, but instead counts the number of affected pairs. However, we consider TD (which represents the total traffic affected between all source-destination pairs in the network and hence is more relevant to both ISPs and customers) instead of ODA to capture the impact of failures on service availability.

Note that SD time and TD capture the effect of a link failure during service disruption, while DV is a post-convergence effect. However, given complete network topology specifications (i.e., nodes, links, connectivity, link weights, link delays, and delay parameter), BGP prefix distribution and traffic rate in the network, all the metrics can be pre-computed and used to characterize the end-to-end performance of a network.

B. Goodness Factors

To use the notion of service availability in characterizing network topologies, it is necessary to capture it using a quantitatively measure. Intuitively, this measure should yield a numerical value that can estimate the end-to-end performance of a network and help in differentiating various topologies. In order to accomplish this, we define a set of *goodness factors* based on different perspectives of service availability. It is important to note that these factors can be defined differently depending on the specific scenario (for example, it could be driven by the cost involved, SLA specifications defined by ISP, etc.). The rest of the paper presents one specific example of such definitions.

We first introduce the notations that we use in the rest of this section. Consider a network with N nodes and M links. Let Γ and Λ represent the set of nodes and links, respectively. For any node i, there are $(N-1)$ different destinations in the network and hence $(N-1)$ different paths with node i as the ingress node. For the failure of link j, all/some/none of $(N-1)$ paths could be affected. Let Q_{ihj} and T_{ihj} denote the SD time and TD for the source-destination pair $i-h$, due to the failure of link j. Similarly, let S_{ij} denote the number of delay parameter violations from ingress node i along all $(N-1)$ paths due to the failure of link j.

B.1 Goodness from Ingress Node Perspective

Typically, many customers are connected to a network at an ingress node. TD represents the total traffic affected for all the customers connected to the ingress node due to a link failure and does not provide valuable information to individual customers. Instead, we use SD time and DV to measure the goodness of a network from an ingress node perspective. We define

$$GI_i = f(Q_i, S_i) \tag{1}$$

where GI_i is the goodness factor of the network from the perspective of ingress node i. Q_i is the average SD time for node i across all $(N-1)$ paths and M possible single link failures

$$Q_i = \frac{1}{M(N-1)} \sum_{\forall h \in \Gamma, h \neq i} \sum_{\forall j \in \Lambda} Q_{ihj}. \tag{2}$$

Similarly, S_i is the average of the number of delay parameter violations from node i, to all other nodes in the network due to various single link failures

$$S_i = \frac{1}{M} \sum_{\forall j \in \Lambda} S_{ij}. \tag{3}$$

The function f depends on SLA specifications between ISP and customers. For simplicity, we assume that goodness is inversely proportional to various metrics. Hence,

$$GI_i = \frac{C}{(Q_i)^q (S_i)^p} \tag{4}$$

where C is a constant. The exponents q and p are SLA dependent. In our simulations, we assume the exponent values to be 1.

B.2 Goodness from Link Perspective

The impact of a link failure on the network performance directly depends on total TD and total DV (i.e., sum of DV for all nodes) due to the failure. High values of these metrics for a link implies that the link is critical to network performance. We define the goodness factor from a link perspective based on similar assumptions on f as in (4)

$$GL_j = \frac{C}{(T_j)^t (S_j)^p} \tag{5}$$

where GL_j is the goodness from the perspective of link j. Similar to (4), the exponents t and p can be defined based on SLA. In our current work, we assume these values to be 1. T_j is the total TD due to the failure of link j

$$T_j = \sum_{\forall h \in \Gamma, h \neq i} \sum_{\forall i \in \Gamma} T_{ihj}. \tag{6}$$

Similarly, S_j is the total number of delay parameter violations due to the failure of link j, i.e.,

$$S_j = \sum_{\forall i \in \Gamma} S_{ij}. \tag{7}$$

To find critical links in a network, it is more useful to compute the *badness* of a link rather than its goodness. Hence, we define the badness of link j in the network as the inverse of it's goodness

$$BL_j = C(T_j)^t (S_j)^p. \tag{8}$$

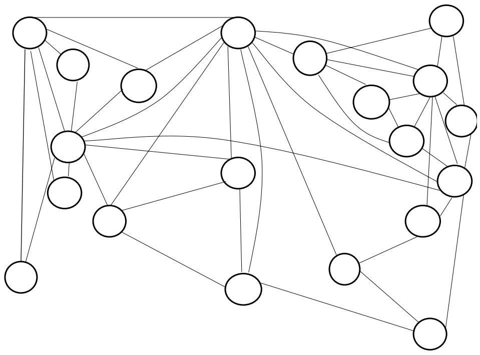

Fig. 2. ISP-A topology.

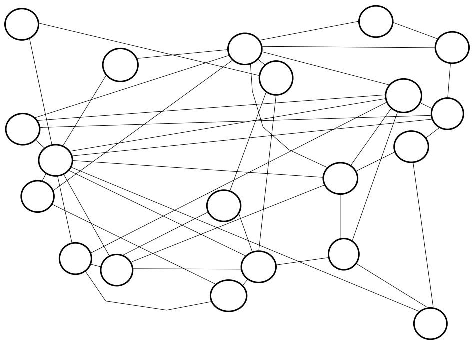

Fig. 3. ISP-B topology.

B.3 Goodness from Network Perspective

We define the goodness of the entire network as the sum of link goodness factors for various links in the network

$$GN = \sum_{\forall j \in \Lambda} \frac{C}{(T_j)^t (S_j)^p} \tag{9}$$

where the exponents t and p are SLA-dependent and we assume these values to be 1 in this work.

IV. RESULTS AND DISCUSSION

In this section, we use the metrics proposed in Section III to quantify the impact of link failures on service availability of a network. First, we illustrate how the algorithm presented in Section II is used to compute SD time, TD, and DV for different classes of network topologies. Then, we examine how traditional graph properties such as out-degree, network diameter, increase in tree depth, and disconnecting sets correlate to QoS offered by the network. We will show the effectiveness of goodness factors in differentiating various network topologies by capturing the routing dynamics that affect traffic forwarding performance. Lastly, we will discuss the implication of the graphs and how goodness factors can be used in evaluating "quality" of connectivity and network design applications.

A. Simulation Setup

We built a Java-based simulator to emulate intra-domain routing dynamics in the presence of link failures and to implement the algorithm presented in Table 3. The inputs to the simulator are complete network topology specifications, BGP prefix distribution, and traffic load along different links in the network. In our simulations, each node in the network topology is mapped to a geographic location. The delays for individual links are then calculated based on the geographical distance between the nodes that the link connects. We categorize the nodes in a network as large, medium, and small depending on the amount of traffic they generate. We consider 20% of the nodes as large nodes, 30% as medium nodes, and the rest as small nodes. This classification and distribution of nodes (i.e., large, medium, and small nodes) in a network is based on real-world observations in point of presence (PoP) level network topologies of various tier-1 ISPs. Also, based on our observations in a tier-1 ISP, the *average* traffic (both ingress and egress) carried by large nodes is about four times that of small nodes, while medium nodes carry twice as much traffic (on an average) as the small nodes. In our simulations, we used these observations to generate the traffic matrix.

We distribute BGP prefixes proportional to the traffic between nodes. Large traffic flow from a source node to destination node implies that the source node reaches a large number of prefixes in the Internet through the destination node. Our results with different assumptions for prefix distributions yielded interesting results. We will discuss this further in Section V.

Based on these inputs, the simulator runs Dijkstra's SPF algorithm to find the shortest path from every node to all other nodes in the network. It then simulates single link failures and executes Dijkstra's SPF algorithm again, to find new paths in the network. The SD time for various OD pairs are calculated based on the algorithm in Section II. TD is then calculated using SD time and network traffic distributions between different source-destination pairs. DV is determined using the link delay values and delay parameter specification. We assume that the maximum tolerable delay to be 100 ms in our simulations.

We consider the following two sets of topologies.

Set I: The first set of topologies represent standard topologies which includes ring, full-mesh, and two PoP-level tier-1 ISP topologies (ISP-A and ISP-B in Figs. 2 and 3). We use these topologies to show that the metric values calculated from our simulations are intuitively correct. All the topologies considered in the first set have 20 nodes in their networks. ISP-A and ISP-B have 44 links each while the ring and mesh topologies have 20 and 190 links, respectively.

Set II: For the second set of topologies, we consider 10 topologies which are "similar" in terms of their intrinsic properties like average out-degree and network diameter. The diameter refers to the maximum depth of all routing trees in the network. One of the topologies considered here is ISP-A (Fig. 2) from Set I. The other 9 topologies were generated by changing link connectivity in ISP-A. Each topology was generated using the following procedure:

- Randomly delete x links from ISP-A topology. We consider $10 \leq x \leq 15$ in our simulations.
- Randomly add x links back into the topology while honoring the following rules: (i) The minimum out-degree of any node in the network is 2. (ii) The resulting topology is con-

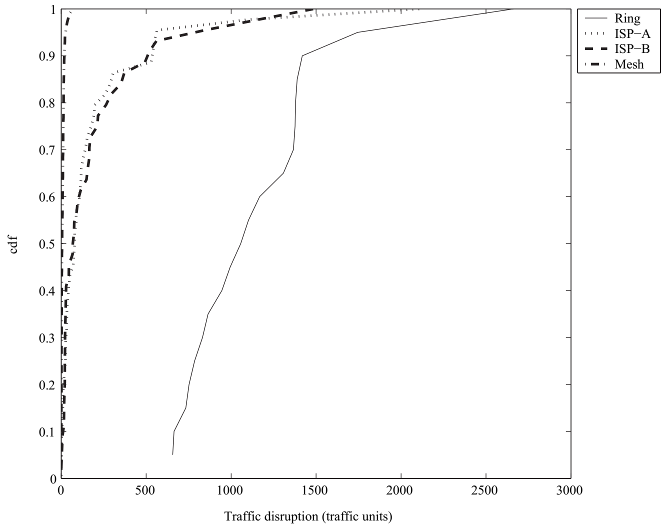

Fig. 4. cdf of traffic disruption due to single link failures.

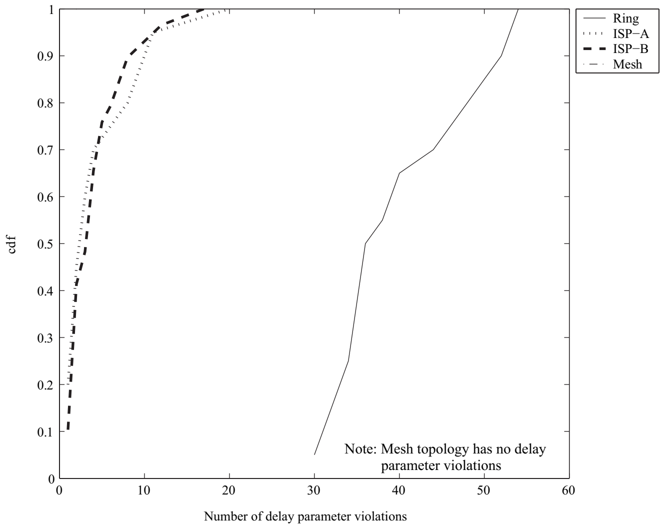

Fig. 5. cdf of number of delay parameter violations due to single link failures.

nected, i.e., a spanning tree for the resulting topology has 19 links.

All the topologies considered in Set II have 20 nodes and 44 links with an average out-degree of 4.4 per node. We found that the network diameter for the topologies lie in the range of 4–6.

In all cases, we assign equal weights to all the links in a network, thus making it a minimum-hop routing scheme. In reality, different IGP link weight assignment schemes yield different metric values and we will further explore this in Section V. To achieve a fair comparison, we consider the same traffic distributions in all the topologies. Traffic rate to/from large, medium, and small nodes remain the same in all the topologies. Finally, we want to point out that the values of goodness factors are normalized to 1 in all the results presented.

B. Service Availability for Known Topologies

In this section, we use the metrics proposed for service availability to study the performance of four known topologies (Set I). Intuitively, a mesh topology should perform the best among all the networks with the same number of nodes while a ring topology should perform the worst. Fig. 4 shows the cumulative distribution (cdf) of the total TD in each of the four networks for various single link failure scenarios. Every link in the mesh topology carries traffic between a single OD pair, thus resulting in small values of total TD for single link failures while every link in a ring topology carries traffic between multiple OD pairs resulting in very high values of total TD. These two topologies are the extreme cases for any topology with the same number of nodes. TD values for other topologies, like ISP-A and ISP-B, lie in-between these extreme values.

Fig. 5 shows the cdf of total DV for single link failures. There are no delay parameter violations for the mesh topology after single link failures while ring topology is significantly worse compared to other topologies for obvious reasons.

Fig. 6 shows the cdf of total number of OD pairs affected by single link failures. In the ring and mesh topologies, from the perspective of connectivity between various OD pairs, intuitively, we can see that all the links in the network should be equally important. In other words, a failure on any link in the network should affect the same number of OD pairs. From Fig. 6, we can see that in the ring topology each link failure affects 100 OD pairs, while in a mesh topology each link failure affects a single OD pair. ODA for ISP-A and ISP-B lie in-between the extreme values of the ring and mesh topologies.

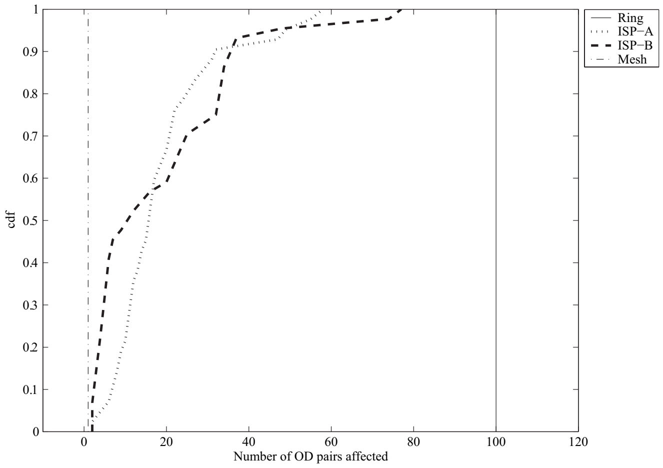

Fig. 6. cdf of number of OD pairs affected due to single link failures.

C. Goodness Factors vs. Static Graph Properties

The following case studies illustrate the limitations of static graph properties (like out-degree distributions or network diameter) to evaluate the performance of a network. This provides the motivation to characterize network graphs using *goodness factors*, which are directly derived based on estimated performance, i.e., service availability. All the following results are for topologies in Set II.

In the past, out-degree of a node has been used as a metric for characterizing a source node [16]. Fig. 7 shows the maximum, minimum, and average goodness factor values for various source nodes as a function of their out-degree. The goodness factors for source nodes with the same out-degree are consid-

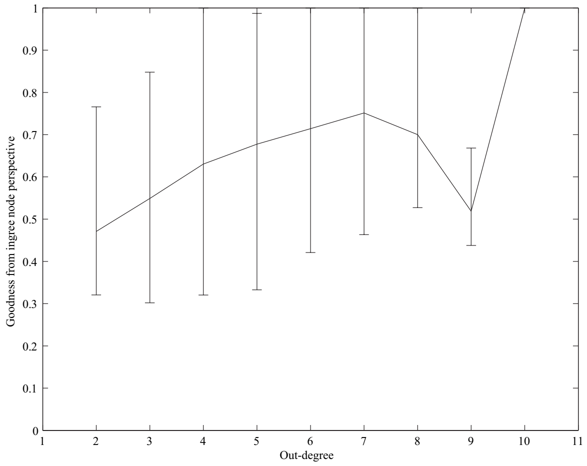

Fig. 7. Goodness factor from an ingress node perspective vs. out-degree in various topologies with normal prefix distribution.

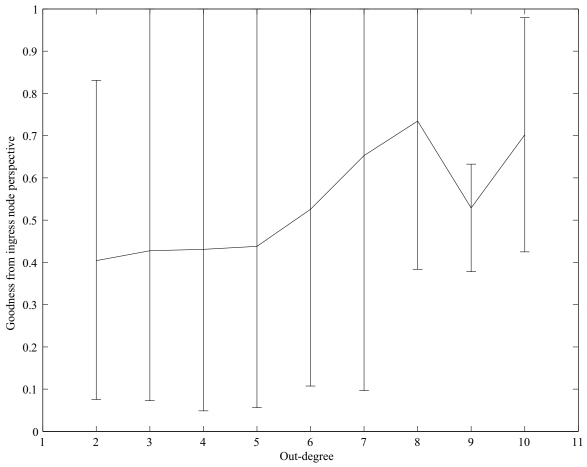

Fig. 8. Goodness factor from an ingress node perspective vs. out-degree in various topologies with extreme prefix distribution.

erably different. Also, higher out-degree does not always result in higher goodness factor values. The main problem in using out-degree to estimate the performance of a source node is that it does not capture the effect of various network characteristics such as BGP prefix distribution and link weight assignment scheme which have a significant impact on source node performance. To show this, we repeat the simulations by distributing BGP prefixes such that all the prefixes have a single exit point in the network (we refer to this as *extreme prefix distribution*). None of the other topology properties were altered. The graph in Fig. 8 shows that changing BGP prefix distribution in the network changes the source goodness factor for various nodes. This implies that out-degree is not an appropriate metric to measure the performance of source nodes.

Network diameter gives an estimate of the maximum convergence time in the network. One would expect that a network with a small diameter would exhibit small convergence time and hence offer better service availability. The top graph of Fig. 9 shows the network goodness factor for various topologies against network diameter. However, our results show that topologies with smaller diameter do not always result in higher

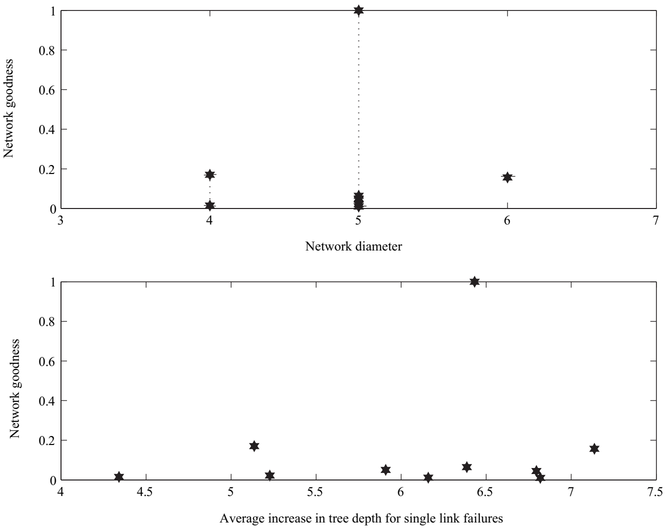

Fig. 9. Top graph shows the variation of network goodness factor with network diameter while the bottom graph shows the variation of network goodness with average increase in tree depth due to single link failures.

goodness factors. Also, topologies with same diameter exhibit different network goodness factor values. Like out-degree, network diameter does not account for several characteristics of a network and hence is not a good metric for predicting its performance.

Another metric related to network diameter is the increase in tree depth due to single link failures. In a well-designed network, the end-to-end delay depends directly on the depth of the routing tree from source to destination. Hence, DV depends on the increase in tree depth after a failure. Typically, lower values of increase in tree depth due to single link failures should result in better topologies. From our simulations, we find that this is not true. The bottom graph in Fig. 9 shows the average increase in tree depth for various topologies against the network goodness factors. Even though increase in tree depth can be used to roughly estimate DV, we find that it is not a good metric for predicting network goodness.

Another important traditional metric used to estimate network performance is the number of *disconnecting sets* in a network [7]. Disconnecting set of a topology is defined as a set of links, whose cardinality (i.e., the number of elements in the set) is less than the minimum node out-degree in the network and the removal of all the links in the set from the topology disconnects it into two or more smaller topologies [17]. The topologies that we consider here, have nodes with out-degree of two. Hence, by the definition above, the cardinality of the disconnecting set should not be more than 1. Since this definition yields a null set for all the topologies, we define a disconnecting set, D, with the following modifications:

- D partitions a topology into 2 parts with each part having at least 2 nodes.
- Exclude supersets: If D_1, D_2, $\cdots$, D_n are n disconnecting sets for a network, then $D_i \not\subseteq D_j$ when $i \neq j$.

There are numerous solutions to the above definition of D, but we consider only those sets for which the cardinality of D

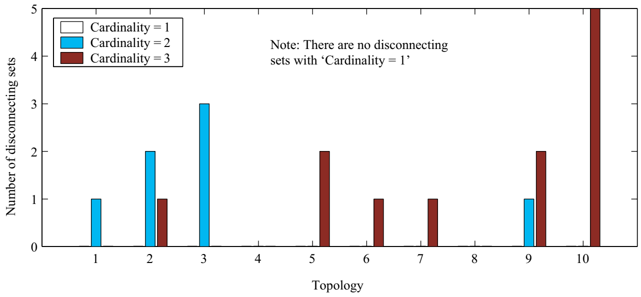

Fig. 10. Disconnecting sets for different topologies.

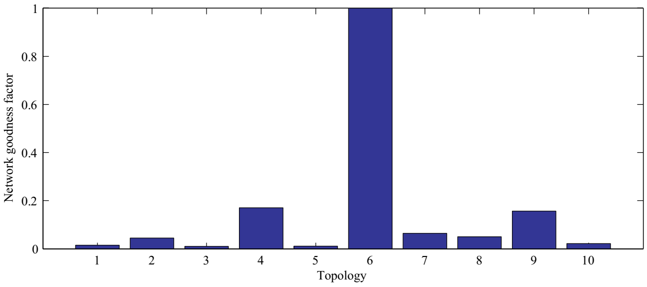

Fig. 11. Network goodness factors for different topologies.

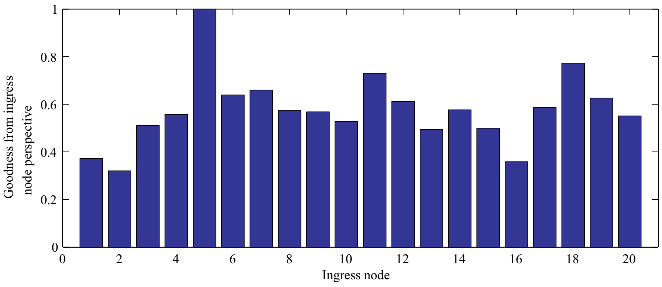

Fig. 12. Goodness factors from the perspective of various ingress nodes in topology-6.

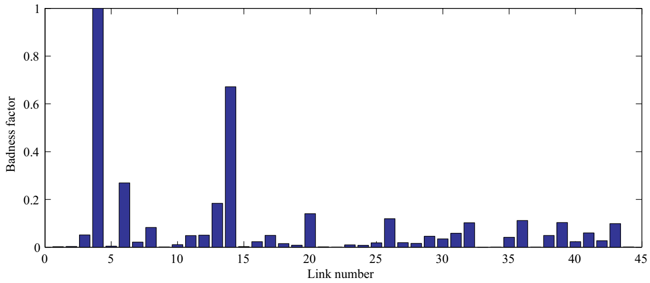

Fig. 13. Link badness factor for various links in topology-6.

is less than 3, i.e., sets with at most 3 links removed. Disconnecting sets determine links that have a big impact on network performance when they fail. The failure of links in the disconnecting set could potentially result in high TD and DV. As the cardinality of the disconnecting set increases, the criticality of the links in the set decreases. For example, a single link connecting two sub-topologies is more critical than two links connecting two sub-topologies. Fig. 10 shows the number of disconnecting sets with different cardinalities for various topologies and Fig. 11 shows the network goodness factor for various topologies. Comparing the two figures we can easily see that even though topology-4 and topology-8 have no disconnecting sets, they do not result in the best network goodness factor values. Hence, similar to the other traditional metrics, disconnecting sets are not good metrics to capture service availability of a network.

Fig. 11 shows that even though the topologies are similar, the end-to-end performance based on service availability is significantly different. ISPs can take advantage of this network differentiation to design their networks to provide high service availability to customers. It is also helpful in estimating the cost of compensating the customers for SLA violations of the network.

V. APPLICATIONS OF GOODNESS FACTORS

This section explores the various applications of the proposed *goodness factors*, and present our initial numerical results using topologies in Set II as case studies.

A. Goodness Factors from Ingress Node and Link Perspectives

Given an ISP network, Fig. 12 shows the performance that a customer can expect when connected to different ingress nodes. This helps a customer to choose an ideal location to get connected to the network. It also helps the ISP to ensure that it can meet the SLA specifications for customers at a given source node.

Fig. 13 shows badness factors of all the links for one specific network topology (topology-6). The graph shows that there are a handful of critical links (e.g., link-4 and link-14) that have a big impact on network performance when they fail. The rest of the link failures only result in minor service degradation. The ability of the badness factors to clearly distinguish the critical links in the network is extremely useful for capacity planning and traffic engineering purposes of an ISP. For example, this information allows the ISP to make an informed decision about bringing down a link for maintenance to minimize the impact on network performance. ISPs can also re-negotiate peering relationships to divert traffic away from critical links. With a better picture of how the failure of individual links can impact performance, the ISP can better estimate whether certain SLAs can be met.

B. Network-Wide Goodness Factors

Our results from the previous sections have established the importance of characterizing topologies based on meaningful performance metrics such as service availability. The proposed *network-wide goodness* aims to capture the impact of routing dynamics on service degradation measured in terms of TD and DV across all source-destination pairs. This goodness metric forms the basis for optimizing network design decisions. We explore three such applications in the following discussion.

IGP link weight assignment: IGP link weights determine the routing trees and hence can significantly influence the service availability of a network. We explore how goodness factors can be used to "evaluate" different link weight assignment schemes in Fig. 14. We consider the same ten topologies in Set II but with three different link weight assignments: (i) Weights pro-

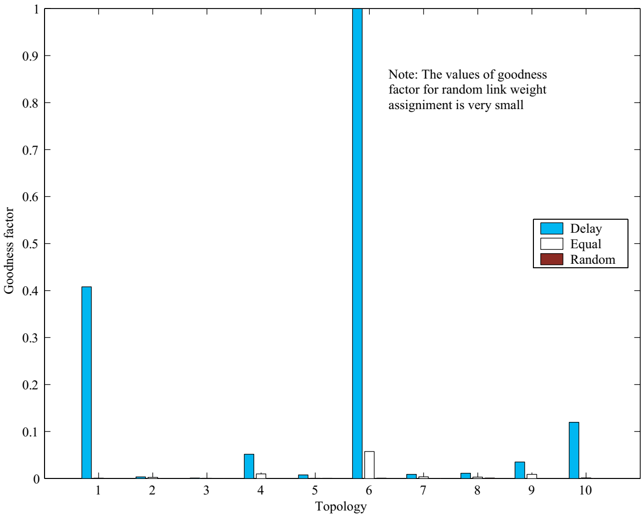

Fig. 14. Network goodness for topologies with different link weight assignment scheme.

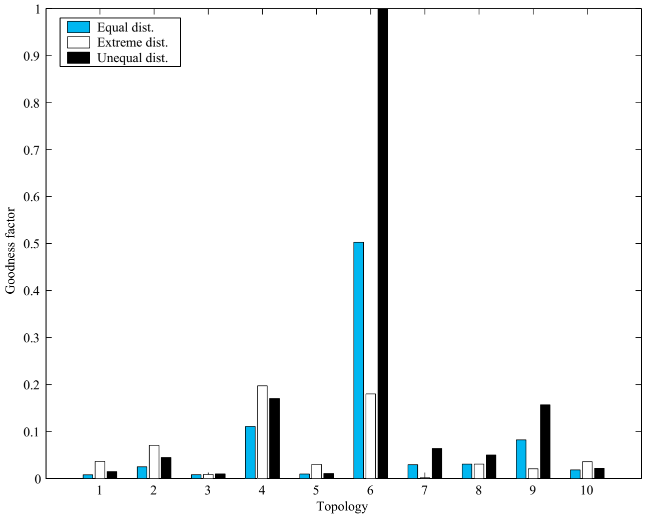

Fig. 15. Network goodness for topologies with different prefix distribution schemes.

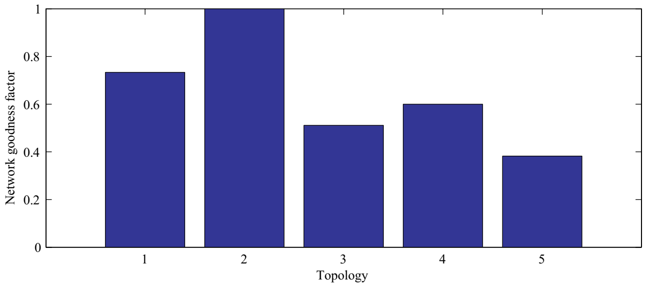

Fig. 16. Adding a new node into the network—network goodness for various solutions.

portional to link delays, (ii) equal weights, and (iii) random weights. Fig. 14 shows that the same topology can behave quite differently when different link weights are used. As part of our future work, we will explore how goodness factors can be used as an optimization metric for selecting link weights to provide the best service availability.

BGP peering points: As discussed in Section IV-C, the location of BGP peering points determine the prefix distribution size at different nodes, and hence also influences service availability. In Fig. 15, we compare network goodness for topologies with three ways of distributing BGP prefixes across different nodes: (i) Equal distribution, (ii) extreme distribution where all the BGP prefixes are located at one exit point, and (iii) typical prefix distribution (or unequal prefix distribution) observed in a tier-1 ISP. Depending on the network topology, different locations of BGP peering points result in different network goodness in terms of overall service availability. This illustration shows that goodness factors can be a useful metric in determining "ideal" BGP peering points that result in the most desired network performance.

Network upgrade: To cope with customer demands and meet SLAs, an ISP may have to schedule network upgrade to introduce a new node or link into its network. Deciding *where* in the existing network to connect this new node/link to, becomes a design challenge. In this case study, we consider adding a new node with 3 links into ISP-A network (Fig. 2). Fig. 16 shows five possible solutions and compares the resulting network-wide goodness of the new network. Solution-2 clearly shows the best network goodness in terms of the offered service availability in the presence of failures.

VI. CONCLUSIONS AND FUTURE DIRECTIONS

In this work, we examined the importance of incorporating network dynamics in characterizing topologies. Our simulations show that traditional metrics like out-degree, network diameter, and disconnecting sets that disregard network dynamics do not effectively capture the performance of a network. Hence, it calls for a new approach to characterize topologies. To fill this void, we proposed a novel methodology based on the concept of service availability and demonstrated its effectiveness using simulations.

To the best of our knowledge, this is the first work to consider network dynamics in characterizing topologies. The approach is the first step in the right direction and is appealing to both ISPs and customers alike. We have identified numerous applications for goodness factors in capacity planning, network design, and upgrade, but their detailed analysis is a part of our future work.

A. Directions for Future Work

In this paper, we have characterized various networks based on their service availability assuming that all link failures in backbone networks are equally likely with a uniform probability. In fact, recent studies (like [1]) have shown that different links exhibit different failure characteristics. Some links fail more often while others do not fail as frequently. In addition, some link failures last for a longer duration when compared to some others. In particular, the authors in [1] empirically show that in the sprint north American backbone network:

- A majority (70%) of the unplanned failure events are isolated, i.e., only affect a single link at a time, and hence can be modeled as independent link failures.
- Links are highly heterogeneous, i.e., some links fail sig-

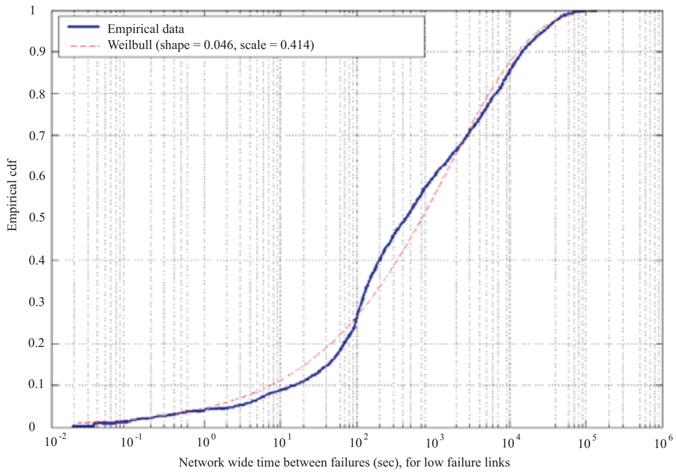

Fig. 17. Network-wide time between failures on low failure links and the approximation by Weibull distribution.

nificantly more often than others. This motivates the classification of the links into two categories—*high-frequency* and *low-frequency* links, and model them separately. Within each class, the number of failures, $n(l)$, for link l follows roughly a power-law, i.e., $n(l) \propto l^{-k}$, where k is found to be -0.73 for high-frequency links and -1.35 for low-frequency links.

- The empirical cumulative distribution function (cdf) for time between any two failures can be approximated by a Weibull distribution. For example, Fig. 17 shows the empirical cdf for the network-wide time between failures for low-frequency links. The Weibull parameters can be derived for each set of empirical data based on maximum-likelihood estimation, e.g., in this case, $\alpha = 0.046$ and $\beta = 0.414$. The cumulative distribution of the duration of failures observed over the same period show that most failures are transient (i.e., short-lived): 46% last less than a minute and 85% last less than ten minutes.

It is important to note that link failures in different networks can follow different distributions, and hence their goodness factors not only depend on the topology and operational network conditions, but also on the link failure model of the network.

Fig. 18 shows the cumulative distribution function of the traffic disruption in the ISP-A network using the failure model observed in [1]. The entire simulation time was eight hours. During each simulation run, different links were associated with different failure probabilities that were generated from a Weibull distribution (as in [1]). Every simulation run resulted in a curve in Fig. 18. We can see that changing association between failure probabilities and links results in very different distributions of traffic disruption implying that goodness factors depend heavily on the correct association of failure probabilities with network links. As a part of our future work, we plan to use a measurement-based approach to explore the sensitivity of goodness factors to link failure models in different networks.

In addition, much work remains in extending this work to incorporate multiple simultaneous link failures. As a complement to simulations, we plan to validate our results through experiments on a test-bed or measurements in real world networks.

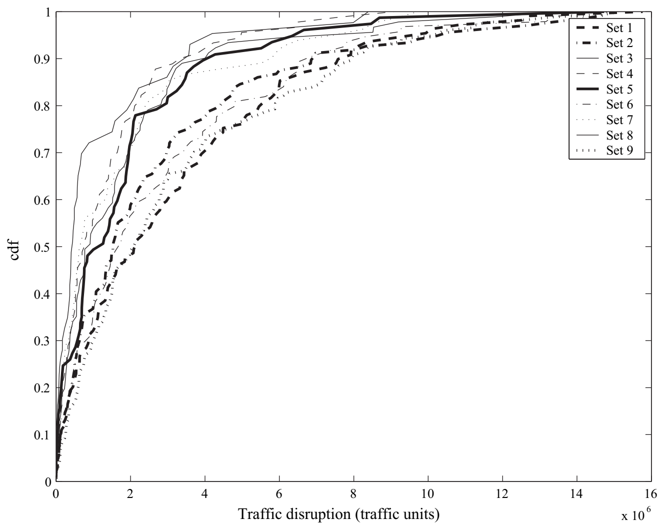

Fig. 18. TD for nine different associations of links with failure probabilities.

VII. ACKNOWLEDGEMENTS

This research project was supported by National Science Foundation CAREER grant number 238348 and University of California Micro with matching funds from Fujitsu and Sprint.

REFERENCES

[1] A. Markopoulou, G. Iannaccone, S. Bhattacharrya, C. N. Chuah, and C. Diot, "Characterization of failures in an IP backbone network," in *Proc. INFOCOM 2004*, Mar. 2004.

[2] C. Boutermans, G. Iannaccone, and C. Diot, "Impact of link failures on VoIP performance," in *Proc. NOSSDAV 2002*, May 2002.

[3] A. Sridharan, S. Moon, and C. Diot, "On the causes of routing loops," in *Proc. ACM Sigcomm Internet Measurement Conf. 2003*, Oct. 2003.

[4] C. Filsfils, "Deploying tight—SLA services on an IP backbone," June 2002, available at http://www.nanog.org/mtg-0206/ppt/filsfils.

[5] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and C. Diot, "Packet level traffic measurements from the sprint IP backbone," *IEEE Network*, Nov. 2003.

[6] G. Iannaccone, C. Chuah, R. Mortier, S. Bhattacharyya, and C. Diot, "Analysis of link failures in an IP backbone," in *Proc. ACM Sigcomm Internet Measurement Workshop 2002*, Nov. 2002.

[7] P. Radoslavov, H. Tangmunarunkit, H. Yu, R. Govindan, S. Shenker, and D. Estrin "On characterizing network topologies and analyzing their impact on protocol design," Tech. Rep. USC-CS-TR-00-731, Mar. 2000.

[8] C. Faloutsos, P. Faloutsos, and M. Faloutsos, "On power-law relationships of the Internet topology," in *Proc. ACM SIGCOMM'99*, Sept. 1999.

[9] R. Govindan and H. Tangmunarunkit, "Heuristics for Internet map discovery," in *Proc. INFOCOM 2000*, Mar. 2000.

[10] K. Calvert, M. Doar, and E. W. Zegura, "Modelling Internet topology," *IEEE Commun. Mag.*, June 1997.

[11] A. Medina, A. Lakhina, I. Matta, and J. Byers "BRITE: An approach to universal topology generation," in *Proc. MASCOTS 2001*, Aug. 2001.

[12] R. Keralapura, C. Chuah, G. Iannaccone, and S. Bhattacharyya "Service availability: A new approach to characterize IP backbone topologies," in *Proc. IEEE Int. Workshop Quality of Service*, Mar. 2004.

[13] Dave Oran, "OSI IS-IS intra-domain routing protocol," *RFC 1142*, Feb. 1990.

[14] J. Moy, "OSPF version 2," *RFC 2328*, Apr. 1998.

[15] G. Iannaccone, C.-N. Chuah, S. Bhattacharyya, and C. Diot, "Feasibility of IP restoration in a tier-1 backbone," *IEEE Network*, Mar. 2004.

[16] B. Waxman, "Routing of multipoint connections," *IEEE J. Select. Areas Commun.*, vol. 6, no. 9, pp. 1617–1622, Dec. 1988.

[17] R. Ahuja, T. Magnanti, and J. Orlin, *Network Flows: Theory, Algorithms, and Applications,* Prentice Hall, 1993.

Ram Keralapura is a Ph.D. candidate in Electrical and Computer Engineering department at University of California, Davis. He received his B.E. in Electrical Engineering from Bangalore University, India in 1998 and M.S. in Computer Science from the University of Alabama in 2000. From 2001 to 2002, he worked on optical switching technology in Sycamore Networks. His research interests are in the area of computer networks and distributed systems including protocols, routing, management, cross-layer design, security, traffic engineering, and overlay networks.

Adam Moerschell is currently a masters student in the Electrical and Computer Engineering Department at U.C. Davis. He received his B.S. in Computer Engineering from U.C. Davis in 2004. His research interests are in the area of computer architecture, graphics architecture, and general purpose computation on graphics processors. He received the Department of Electrical and Computer Engineering Citation in 2004.

Chen-Nee Chuah is currently an assistant professor in the Electrical and Computer Engineering Department at U. C. Davis. Chuah received her B.S. in Electrical Engineering from Rutgers University in 1995, and her M.S. and Ph.D. in Electrical Engineering and Computer Sciences from U. C. Berkeley in 1997 and 2001, respectively. From 2001 to 2002, she was a visiting researcher at Sprint Advanced Technology Laboratories. Her research interests are in the area of computer networking, wireless/mobile communications, network measurements, anomaly detection, overlay and peer-to-peer systems, and performance modeling. Chuah received the National Science Foundation CAREER Award in 2003 for her research on robust, stable, and secure routing through an integrated introspection and feedback framework. She received the U.C. Davis College of Engineering Outstanding Junior Faculty Award in 2004. She has served on the technical program committee of several ACM and IEEE conferences and workshops (including INFOCOM, MOBICOM, SECON, IWQoS, ICC, and VANET). She was the TPC co-chair of the First Workshop on Vehicular Ad Hoc Network (VANET), colocated with ACM MobiCom 2004. She is also serving as the TPC vice-chair for IEEE Globecom 2006.

Gianluca Iannaccone received his B.S. and M.S. degree in Computer Engineering from the University of Pisa, Italy in 1998. He received a Ph.D. degree in Computer Engineering from the University of Pisa in 2002. He joined Sprint as a research scientist in October 2001 working on network performance measurements, loss inference methods, and survivability of IP networks. In September 2003, he joined Intel Research in Cambridge, UK. His current interests include system design for fast prototyping of network data mining applications, privacy-preserving network monitoring, and routing stability for overlay networks.

Supratik Bhattacharyya is a principal member of Techical Staff at Sprint Advanced Technology Labs. in Burlingame, CA. He holds a Ph.D. in Computer Science from the University of Massachusetts. His research interests are in Internet systems and protocols and wireless communication and services. His current and past work cover a broad range of topics such as Internet routing, network monitoring and measurements, network fault tolerance, data mining and streaming algorithms, and disruption-resilient wireless services.