
Feasibility of IP Restoration in a Tier 1 Backbone

Gianluca Iannaccone, Intel Research
Chen-Nee Chuah, UC Davis
Supratik Bhattacharyya, Sprint ATL
Christophe Diot, Intel Research

Abstract

Large IP networks usually combine protection and restoration mechanisms at various layers of the protocol stack to minimize service disruption in the event of failures. Sprint has chosen an IP-based restoration approach for building a highly available tier 1 IP backbone. This article describes the design principles of Sprint's network that makes IP-based restoration an effective and cost-efficient approach. The effectiveness of IP-based restoration is evaluated by analyzing network failure characteristics, and measuring disruptions in service availability during controlled failure experiments in the backbone. Current trends for improving the performance of IP-based restoration are also discussed.

Network survivability refers to the ability of a network to maintain uninterrupted service regardless of the scale, magnitude, duration, and type of failures. It has been one of the fundamental design goals of IP networks from the time the ARPANET was built.

There are two main approaches to providing network survivability in IP backbones: protection and restoration [1, 2]. *Protection* is based on fixed and predetermined failure recovery: as soon as a working path is set up, a protection path is also provisioned to forward the traffic if the primary path fails. *Restoration* attempts to find a new path on demand to restore connectivity once a failure has occurred.

Protection and restoration mechanisms can be provided at different layers in an IP backbone. The choice of mechanisms for a given network depends on three criteria. First, it has to be based on an understanding of the nature and causes of failures in a network. Lower-layer survivability mechanisms cannot detect failures occurring at higher layers. For example, an optical protection mechanism cannot protect against an IP router or forwarding software failure. On the other hand, higher-layer entities may be able to protect against (or recover from) lower-layer failures as long as there is an alternate path between communicating entities.

The second criterion is the speed of recovery with a given mechanism. For example, dynamic mesh and ring restoration capabilities at the optical layer can recover from failures in less than 100 ms [1]. On the other hand, the traditional restoration mechanism at the IP layer is rerouting, which may take seconds.

Finally, the cost of providing network survivability has to be considered. IP backbone operators need to invest in improving the reliability of their equipment, and also in provisioning spare capacity for use in the event of failures. Protection mechanisms are inherently more expensive given that resources must be committed without a priori knowledge of the next failure. The cost of protection increases with the scale of failures for which a network operator seeks protec-

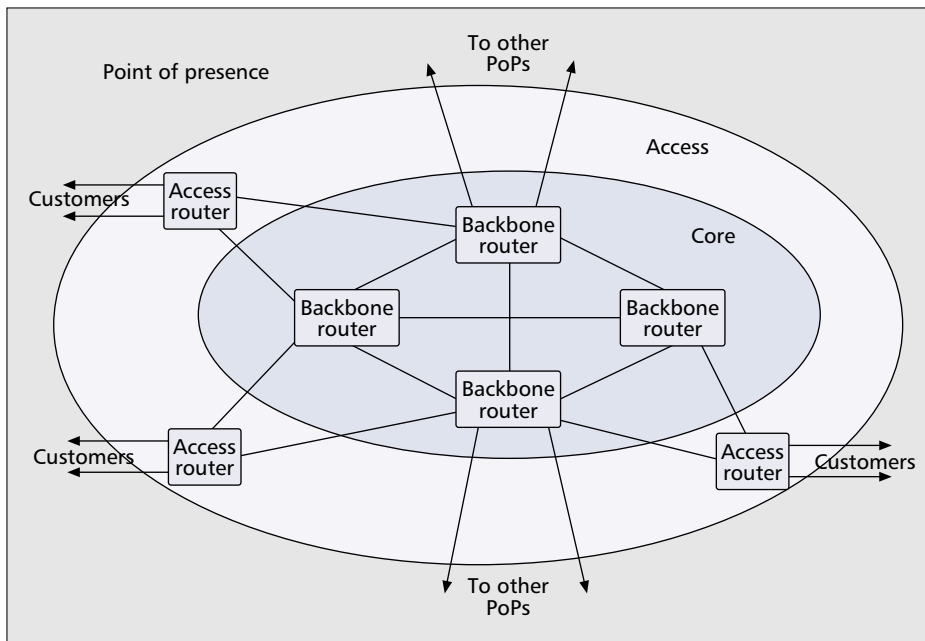
tion. It is also difficult to protect any arbitrary combination of failures. In this regard, restoration is more cost effective, since additional resources are allocated only after a failure has occurred. However, restoration mechanisms offer slower recovery because of the delay involved in finding these resources after the failure has been detected.

The cost of network survivability has played an important role in determining the mechanisms currently used by service providers for failure recovery. Installing new optical fibers to provide physically diverse protection paths has been hampered by several problems. First, installing additional fiber infrastructure requires substantial capital investment. Second, the process of installing a new fiber path is much slower than just increasing the transmission capacity of a single fiber. In addition, geographic limitations (mountains, bridges, etc.) have made it difficult to install protection paths with sufficient diversity. As an alternative to optical layer protection, Sprint, a tier 1 Internet service provider (ISP), has opted for IP-level restoration based on path recomputation through IP routing protocols, such as Intermediate System to Intermediate System (IS-IS) [3] and Open Shortest Path First (OSPF) [4].

In this article we examine the effectiveness of IP restoration for Sprint's backbone network. The specific questions we address are:

- Is IP restoration sufficient to handle failures in the Sprint network?
- Is IP restoration fast enough to provide highly available service across the Sprint network?

We illustrate our discussion using real measurement data collected from the Sprint backbone. We analyze link failure characteristics to understand whether IP restoration can ensure complete failure recovery. We describe experiments conducted to determine the recovery speed of IP restoration, and suggest how it can be improved. Our conclusion is that IP restoration, coupled with adequate capacity provisioning and careful network design, is indeed effective in providing network survivability for the Sprint network.



■ Figure 1. Intra-PoP topology label.

Components of IP-Based Survivability

Network survivability in Sprint's IP backbone is provided solely by restoration mechanisms at the IP level. When a network component¹ fails, the routing protocol at the IP level computes an alternate path for traffic around the failure. This approach imposes several requirements on network design:

Availability of restoration paths. Enough spare capacity should be available to carry traffic in the event of failures.

Localized failure recovery. Route recovery should happen as close as possible to the point of failure to minimize the amount of traffic that needs to be rerouted and the number of nodes that need to modify their packet forwarding decisions.

Satisfying service level agreements (SLAs). SLAs define performance guarantees ISPs offer their customers. A typical SLA includes upper bounds on the average packet loss rate and delays across the network. The restoration path used in the event of a failure should be able to satisfy these SLA guarantees.

Avoiding network partitions. The network should be designed such that it is not partitioned even in the event of large-scale failures involving multiple links and/or nodes. In other words, alternate paths between any two points in the network should be node-disjoint and link-disjoint to the greatest extent possible.

There are three key components in ensuring that the above requirements are met by the Sprint network: capacity provisioning, network topology design, and intradomain routing protocol. We now discuss each in detail.

Capacity Provisioning

IP is a connectionless best effort protocol that does not provide resource reservation or admission control. Therefore, an IP network cannot turn away traffic even when its capacity is reduced due to failures or traffic surges. The Internet is occasionally affected by widespread outages caused by natural or human disasters, or malicious attackers. During such an event, multiple ISPs may be affected. Multihomed customers that buy Internet connectivity from several ISPs may

direct all of their traffic toward the ISP least affected. Therefore, this ISP will experience a sudden surge in traffic volume while coping with multiple link and node failures. The provisioning of redundant capacity is the only feasible approach to survivability in tier 1 backbones, where it is impossible to predict and plan in advance for all possible failure scenarios. This approach has been commonly referred to as *overprovisioning*. A similar approach has been used for survivability in other types of networks (e.g., telephone networks).

The engineering rule for capacity provisioning in the Sprint network is to maintain the average utilization of any link under 50 percent (usually computed over 5 min intervals). An added benefit of low link utilization is that network congestion is avoided, and Sprint can easily adhere to the loss and delay guarantees in its customer SLAs.

Network Topology

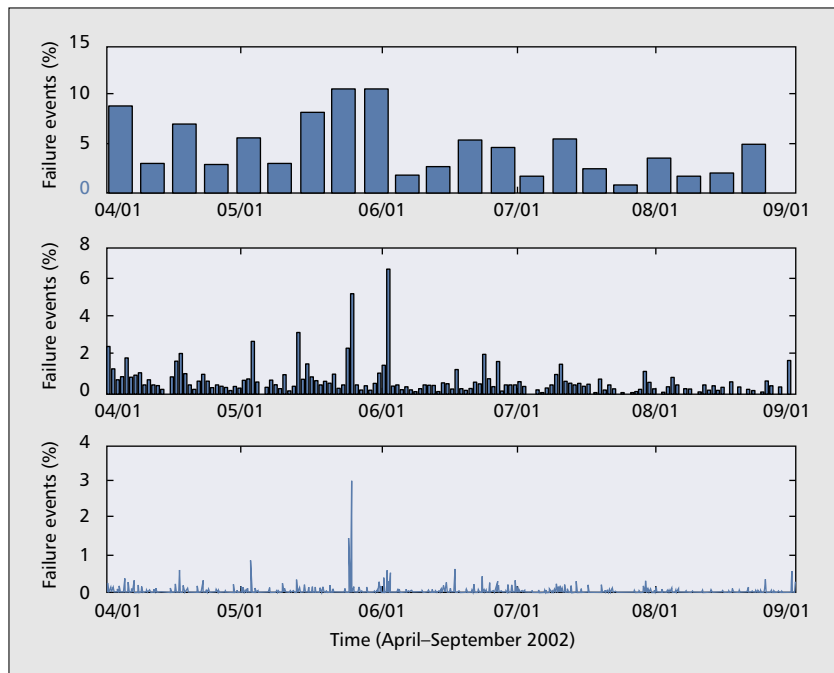
Sprint's IP backbone consists of a collection of points of presence (PoPs) connected via high-speed OC-48 (2.5 Gb/s) and OC-192 (10 Gb/s) links. This "logical" IP network is layered over an dense wavelength-division multiplexing (DWDM) optical network providing physical connectivity. In this work, we refer to logical links as IP links in order to distinguish them from the physical links in the underlying optical fiber infrastructure. A PoP consists of a set of IP routers in a single physical location (usually a city or large metropolitan area). Each PoP connects customers, ranging from large corporate networks to regional ISPs and data centers, to the Sprint network.

Intra-PoP Topology — Figure 1 shows a simplified view of a typical PoP topology in the Sprint's network. Each PoP has a two-level hierarchical structure. At the lower level, customer links are connected to access routers. These access routers are in turn connected to backbone routers. The backbone routers provide connectivity to other PoPs and to other large backbone networks (peers). To provide fault protection, each access router in a PoP is connected to at least two backbone routers, while all the backbone routers are connected in a full mesh topology.

This configuration facilitates fast local restoration for intra-PoP failures. For example, if the link between an access router and a backbone router fails, the restoration path to reroute traffic between the access router and the rest of the network is only one hop longer than the primary path. Moreover, the delay difference between the primary and restoration paths is negligible, since all routers in a PoP are physically collocated. Finally, this restoration is invisible outside the PoP since this additional hop within a PoP does not affect the end-to-end network path of traffic across the backbone (explained in detail later). Similarly, the full mesh connectivity of backbone routers makes fast local restoration possible when a backbone router or a link between two backbone routers fails.

Inter-PoP Topology — A fully meshed inter-PoP topology would guarantee the availability of several alternate paths

¹ Network components include links, routers, router line cards, and optical equipment.



■ Figure 2. Failure notifications over three timescales: weekly (top), daily (middle), and hourly (bottom).

between any two PoPs in the event of a failure. However, the cost of building such a topology would be prohibitive, in terms of the optical fiber infrastructure and the number of router ports required. As a less expensive option, each PoP in Sprint's network is connected to a subset of other PoPs. The number of PoPs attached to any given PoP varies between 2 and 10. This ensures that the loss of all IP links between a single PoP pair does not isolate either PoP from the rest of the network.

In addition, two adjacent PoPs have multiple parallel links connecting them, with each link terminating on a different router within the same PoP. This design offers several advantages with respect to network survivability. It ensures that a restoration path is always available between a pair of adjacent PoPs in case of router or link failure. Second, traffic between the two PoPs can be load balanced on the parallel links. If one of the links fails, the others can easily absorb the additional traffic. Finally, the effect of the failure is kept localized to the PoP pair since there is no change in the path taken by traffic that was routed via this PoP pair before the failure.

Regardless of how PoPs are interconnected at the IP level, IP restoration is possible only if there is physical connectivity at the optical fiber level. In order to guarantee fault resilience in the face of optical failures, parallel IP links between adjacent PoPs must be mapped onto fiber paths that are as physically disjoint as possible. Unfortunately, the choice of fiber paths is limited by two factors. First, geographical obstacles such as mountains and rivers make it difficult to install geographically diverse fiber paths. Second, optical fiber has been historically installed along train tracks and pipelines, thereby being limited to relatively few paths.

Furthermore, the mapping of IP links must also meet SLA requirements. End-to-end delay across the Sprint network is a key SLA metric. Current capacity provisioning practices in the Sprint network results in relatively low link utilization. Therefore, there is negligible queuing of packets at routers, and end-to-end delay is dominated by the propagation delay along the optical fiber path. Hence, the length of the fiber path must be taken into account when addressing the IP-to-fiber mapping problem.

Intradomain Routing

Sprint's IP network uses the link state IS-IS protocol [3] for intradomain routing. Every link in the network is assigned a weight, and the cost of a path is measured as the sum of the weights of all links along the path. Each node participating in the IS-IS protocol announces all its adjacent links to the rest of the network via flooding. This enables every node to independently build a complete database of the cost of the path(s) to reach every other node. A node forwards traffic toward another node along the minimum cost path, which is computed using Dijkstra's shortest path first (SPF) algorithm. Thus, setting link weights provides a mechanism to control the traffic flow through the network.

In the event of a link failure,² the failure is announced by the nodes connected by that link. Every other node in the network updates its database upon hearing about the failure and recomputes its minimum cost paths around the failed link.

If there are multiple minimum cost paths between a pair of nodes, IS-IS allows traffic to be split evenly among these paths. This capability of IS-IS is referred to as equal cost multipath (ECMP), and is a key component of IP restoration. Traffic can be split among the equal cost paths on a per-packet or per-flow basis, where a flow is defined based on IP source and destination addresses. Per-packet splitting causes packets belonging to the same flow to be forwarded along different network paths, potentially reordering them and degrading the performance of higher-level protocols such as TCP. Therefore, the Sprint network uses ECMP with per-flow splitting. Load balancing allows better use of available resources and also gives the network the ability to absorb short-lived spikes in traffic demand that may result from failures with very short durations [5].

The assignment of IS-IS link weights forms the basis of ECMP load balancing on inter-PoP links in Sprint's network. Link weights are assigned in two steps. First, inter-PoP links are assigned weights treating each PoP as a single node. All parallel links between a PoP pair are assigned the same weight. Hence, the traffic between two PoPs is equally split over multiple parallel links at each intermediate PoP (ECMP). Next, intra-PoP links are assigned weights that are significantly smaller than the inter-PoP link weights. This ensures that the inter-PoP link weights primarily determine the paths for traffic across the backbone.

Network Failures

The first step toward defining and measuring IP restoration performance is to develop a detailed understanding of how often failures occur in a network and how long they last. Unfortunately, very little is known about the failures of IP networks; in this section we present some observations based on the analysis of routing protocol messages in Sprint's operational network. The successive step is then to study how the restoration of the forwarding path is performed and measure the impact of failures on traffic. This will be discussed later.

² Since IS-IS is a link state protocol, only link failures are announced. However, when a node fails, all links connected to that node are considered to have failed.

General Failure Characteristics

We use the PyRT listener³ to collect IS-IS routing updates exchanged over the Sprint's backbone. The router connected to PyRT treats the listener as a neighboring router to which it forwards all routing messages it receives from the rest of the network. Since IS-IS routing messages are always flooded through the entire network, our listener is informed of every routing-level change occurring anywhere in the network. For further details we refer the reader to [6].

Whenever IP-level connectivity between two directly connected routers is lost, each router independently broadcasts a *link down* message through the network. When the failed equipment is repaired, each router broadcasts a *link up* message. Note that loss of connectivity at the IP level may be triggered by a variety of causes such as a fiber cut, router interface failure, or an IS-IS protocol malfunction. We refer to each such event as a *failure event*.

The Sprint backbone is in constant evolution with new links being added and older ones being decommissioned every week. When a link is decommissioned, link down messages are broadcast, but there is no subsequent link up message. In order to distinguish link decommissioning from actual failure events, we consider only those failure events for which we subsequently receive a link up message within the next week.

Figure 2 shows the distribution of failure events that occurred on the Sprint U.S. network over a five-month period (April–August 2002) on a weekly, daily, and hourly basis.⁴ Failures are fairly well spread out across weeks, days, and even over the course of a single day. Clearly, they need to be taken into account as part of everyday operations and not just as extraordinary events.

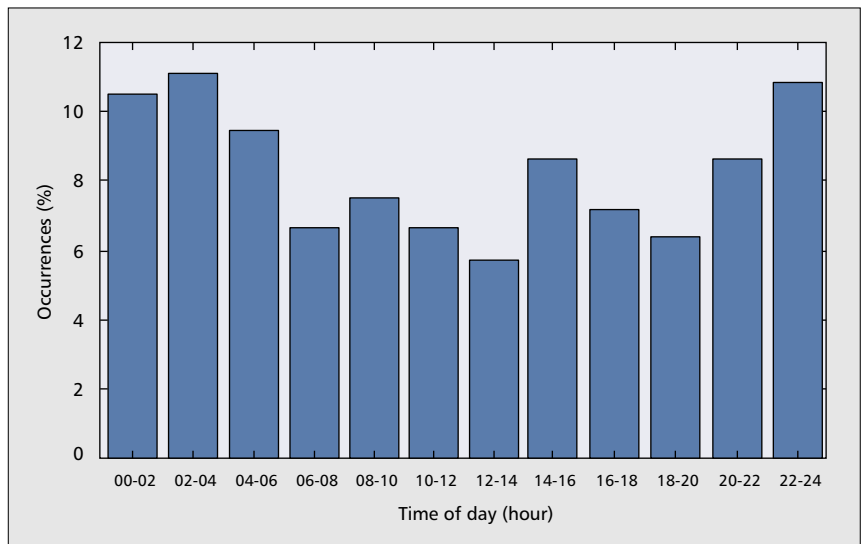
In Fig. 2 we consider failures due to both scheduled maintenance and unplanned events or accidents. It is interesting to investigate the relative magnitude of unplanned failures since it is desirable to eliminate (or at least minimize) them.

Maintenance windows are scheduled during late night/early morning; hence, a breakup of failure events by time of day sheds light on this issue. In Fig. 3 we show all the failures (over the five months of interest) grouped in 2-h bins by time of day (where the time zone is U.S. eastern time). We observe that about 45 percent of the failure events occur between 10 p.m. and 6 a.m. If we take into account the 3-h time difference between the east and west coasts of the United States, this is the time window in which most maintenance windows are scheduled.

Although all failure events during this period are possibly not scheduled maintenance, the fact that this time period accounts for almost half of all failures indicates that maintenance activities do account for a significant portion of the failure events we observe.

Sources of Failures

It is not possible to identify exactly the factors causing failures from IS-IS routing updates alone. However, we can speculate on the causes using the time needed to repair the fault. After a failure is detected via loss of connectivity and the traffic is



■ Figure 3. Failures by time of day (two-hour windows).

rerouted over an alternate path, connectivity repair takes place. The repair may be automated or require human intervention, and may take from tens of seconds to several days depending on the scale of the event.

The time to repair a failure (which we refer to as *failure duration*) may provide some valuable insight on the possible cause. For example, fiber cuts due to accidents require several hours to repair [6]. On the other hand, a router may take several minutes to reboot and restart, and tens of seconds to reset an interface card. Software errors or routing processor overloads may also induce a router to erroneously infer loss of connectivity with one of its neighboring routers; in these cases, repair can be just a matter of seconds (i.e., the time needed to renegotiate the IS-IS adjacency).

Figure 4 shows the cumulative distribution of the duration of failures over the five-month period of interest and provides some insights into the sources of network failures:

- Only 10 percent of the failures last longer than 45 min. These failures are probably related to hardware failures (optical fibers, interface cards) that require human intervention.
- About 40 percent of the failures are repaired in 1–15 min. We conjecture that these events were caused by software errors and required just a router reboot or interface reset.
- About 46 percent of all failure events last less than 1 min. These events could be due to an overloaded router that fails to process keep-alive messages and mistakenly considers connectivity with a neighboring router to be lost. Otherwise, faulty optical equipment can also cause the router to temporarily consider a link to be down.
- The remaining 4 percent of the failures are repaired in 15–45 min and most likely involve human intervention (e.g., substitution or maintenance of hardware equipment).

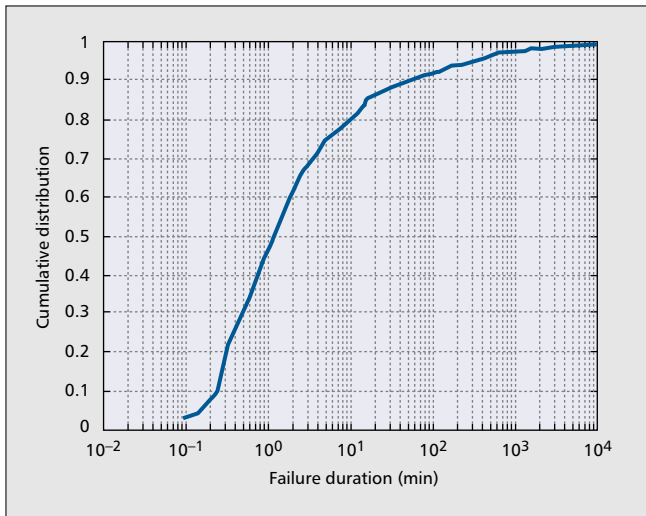
An important observation is that most failures are short-lived. After only a few minutes, the network topology reverts to its initial status (less than 10 min for 81 percent of cases). We refer to all failures that last less than 10 min as *transient failures*.

Investigating transient failures further, we have observed that they are also mostly isolated. Figure 5 plots the distribution of simultaneous failures and clearly illustrates this phenomenon. The y-axis in Fig. 5 represents the fraction of time (at 1 s granularity) that any given number of links are reported down.

The figure shows the distribution of simultaneous transient failures, long-lived failures (i.e., with a duration longer than 45 min), and all failures. We observe that almost 70 percent of transient failures are isolated. On the other hand, longer failure events are more likely to happen in clusters; less than

³ <http://ipmon.sprint.com/PyRT>

⁴ For proprietary reasons we are unable to provide absolute numbers, only the percentage of the total number of failures.



■ Figure 4. Duration of failure events.

30 percent of them are isolated. The presence and magnitude of transient failures has significant implications on the design of mechanisms for traffic engineering. Later we will explore some proposed solutions that specifically address and factor in transient failures and topology changes.

Performance of IP Restoration

A common metric for comparing the performance of different survivability mechanisms is restoration time. Restoration time is defined as the time from loss of connectivity until service is fully restored and communication between higher-layer entities resumes.

Relatively long restoration times (on the order of several seconds) have generally been considered the major weakness of IP restoration mechanisms compared to optical protection/restoration mechanisms. For example, synchronous optical network (SONET) networks are required to restore connectivity in less than 50 ms. However, the choice of protection/restoration mechanisms must be based on the restoration time requirements of existing and emerging applications. In that regard, the American National Standards Institute (ANSI) [7] reports that restoration times below 200 ms are unlikely to have any impact on services (including voice services), while delays between 200 ms and 2 s would only slightly degrade performance on voice services [7]. Therefore, subsecond restoration is commonly regarded as a valid practical threshold.

IP restoration can be divided into three basic phases:

- Failure detection
- Failure notification to the control plane and other network entities
- Forward path recomputation and establishment of the restoration path

In the rest of this section, we address each phase in greater detail with specific reference to the Sprint IP backbone network, a network that covers more than 70 countries on five continents.

Our observations are based on several controlled failure experiments inside the Sprint operational backbone network that allowed us to observe traffic disruption and routing performance [5]. All figures we provide are based on the IS-IS protocol and the router equipment present in the Sprint network at the time of the experiments.

Failure Detection

IP routers make use of two mechanisms to discover if neighboring routers are in a working state:

SONET framing provides the information on the link con-

nectivity for all point-to-point connections and is capable of raising an alarm within 10–20 ms from loss of connectivity.

Keep-alive messages are continuously exchanged between routers to report that the routing software is functional. Given that there is no reliability mechanism associated with keep-alive messages, the detection time for failures is longer, on the order of 30–60 s. Keep-alive messages are used to detect routing software failures, and also to detect failures in the presence of switched networks (e.g., Ethernet LANs).

The core network of a major ISP usually consists of point-to-point connections between routers. This solution is preferred to the use of switched networks because it provides faster detection time with SONET framing.

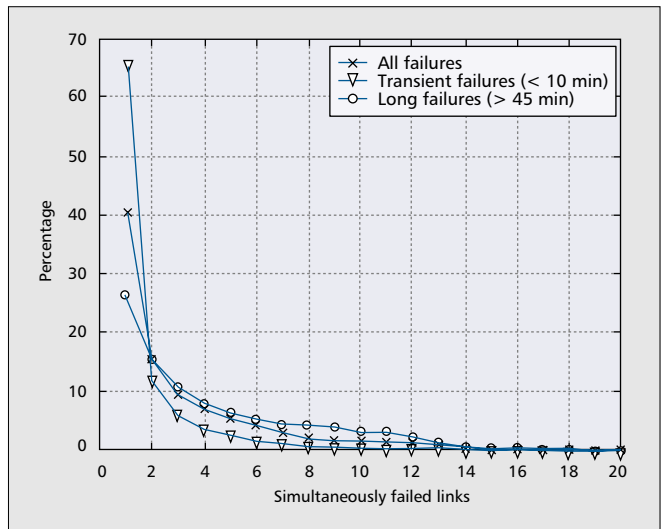
Failure Notification

Once a failure is detected, a router needs to decide when to notify the IS-IS protocol stack. Failures detected via SONET alarms are subject to a *notification timer* used to filter out very short transient link flaps. This timer is also used in the presence of protection mechanisms at the optical layer that may restore connectivity before the intervention of the routing protocol. For this reason, a recommended value for this timer has been 2 s. Today, given the absence of optical protection, a common value for the notification timer is between 10 and 20 ms.

The next step in the notification process is to inform other routers in the network of the topology changes. To this end, routers construct a packet, called a *link state packet* (LSP), with the new information and flood it on the network.

Flooding is used to trigger link database update and route recomputations at all the routers in the network. The news of the failures must propagate through the network as fast as possible to prevent routers having an inconsistent view of the network that could lead to routing instability. Nevertheless, in the case of link connectivity oscillations, LSP flooding may consume excessive network resources. To prevent this, an *LSP generation timer* is used to rate limit the generation and transmission of LSPs.

The time needed to notify all the routers in the network of the topology changes obviously depends on the size of the network. Transmission times are limited by the propagation delay but also by the processing time at each router: we estimate that it may take up to 10 ms/hop. Given that the network is designed to allow localized failure recovery, traffic forwarding is restored as soon as the LSPs reach the routers one or two hops away from the failure.



■ Figure 5. Distribution of simultaneous failures.

Forward Path Re-computation

Upon notification of a topology change, each router uses the Dijkstra algorithm to compute the shortest path to every other node (called a *routing tree*). In a network with n nodes and l links the complexity is $O(l \log n)$.

The computation of shortest paths is therefore a very processor-intensive task and may require anywhere between 100 and 400 ms to complete for a network of the size of the Sprint backbone. This estimate also includes the time taken to update the routing table (i.e., the next hop address for each destination prefix of which the router has knowledge).

To reduce the router processor load, a *shortest path computation timer* is used to aggregate multiple closely spaced LSPs and perform only one computation of the routing tree that incorporates all the announced topology changes. A recommended value for this timer is 5.5 s, although given the processing power of current backbone routers it can be reduced to a few milliseconds.

In the case of distributed router architectures where the packet forwarding is performed in hardware (as it is for most high-speed backbone routers), an additional step is required to propagate the new routing tree to all the interfaces. During this last step the router will not be able to forward packets although it has already identified the restoration path. On today's routers, it takes approximately 1 ms to update 20 routing table entries. Failures that impact thousands of entries are fairly common, requiring several hundreds of milliseconds to complete the forwarding table update.

In summary, failure recovery time is dominated by the hold-off timers introduced to reduce the risk of network instabilities. In Table 1 we summarize all the components of failure recovery. For the timers, we have indicated two values, a default value recommended in the past and a minimum value that indicates what current routers can support according to most vendors.

Years of incremental improvement on the router software and hardware architecture have made routers capable of handling higher routing processing loads without threatening overall network stability. As we can see from the table, reducing the timers involved in IP restoration allowed Sprint to bring the recovery time down from 8–9 s to less than 1 s. Nevertheless, several improvements are needed to reduce the processing components (i.e., routing tree computation and update of the forwarding information). In the next section we present some of the solutions that have been proposed in this regard.

Trends in IP Restoration Solutions

In the previous section we discussed how to fine-tune IS-IS timers to provide fast IP restoration. In this section we survey current research trends and technology improvements that can further improve the performance of IP restoration. Some of the mechanisms and protocol extensions have already been implemented by router vendors, but have not yet been widely deployed in the Internet.

Traffic Engineering Solutions

Most large ISP networks use a link state protocol such as IS-IS or OSPF for intradomain routing. Setting link weights is the primary traffic engineering technique for networks running IS-IS or OSPF. Common traffic engineering objectives are to minimize end-to-end delays and balance traffic load evenly across the network.

A drawback of most current approaches is that they view the link weight assignment problem as a static problem, largely ignoring network link failures. However, in practice, operators of large backbone networks have to deal with link failures on a

Timer	Default value	Minimum value
Notification timer	2 s	10 ms
LSP generation timer	50 ms	1 ms
Shortest path computation timer	5.5 s	1 ms
Processing phase		Typical values
LSP processing	10 ms/hop	
SPF computation	100–400 ms	
Forwarding information update	20 entries/ms	

■ Table 1. Components of the failure restoration time.

daily basis. An obvious approach is to recompute and change link weights to reroute traffic in the event of a failure. Fortz *et al.* [8] have shown that changing just a few link weights is usually sufficient to rebalance the traffic. However, changing link weights during a failure may not be practical for two reasons. First, the new weights require route recomputations at all routers, thereby aggravating the disruption in forwarding already caused by the failure. Second, the high frequency of short transient failures we observe in the IP backbone makes it impossible for a human operator to reassign link weights in real time and for the routing layer to converge rapidly enough to alleviate the problem. To address this problem, Nucci *et al.* [9] proposed a heuristic-based optimization technique that assigns link weights taking into account the isolated failure of any link in the backbone. This approach protects against link overloads during failures, thereby preventing any disruption in service availability due to transient link failures.

Routing Protocol Extensions

Several extensions have been proposed to IS-IS and OSPF to improve restoration times. The most promising ones are mentioned below.

Content-Based Rules for Processing LSPs — As described earlier, the generation and propagation of LSP messages is rate-limited independent of the content of the LSP. An LSP that reports a link failure needs immediate attention, however, since IS-IS has to find a restoration path around the failure. On the other hand, an LSP that reports the repair of a failed link can tolerate a longer delay in processing since IS-IS would already have found an alternate route around the failure. Therefore, LSPs that report loss of connectivity should receive higher priority than those reporting link recoveries.

Precomputation of Second Shortest Paths — Currently, each node computes the minimum cost path to every other node in the network. An alternate path is computed only if the primary path becomes unusable due to a failure. Instead, each node may consider the failure of one or more of its directly connected links and compute alternate routes to all other nodes in the event of such a failure [10]. This expedites the process of switching to an alternate path in the event that a directly connected link fails. Such an approach may lead to routing loops, but it is possible to devise a network topology and link weight assignment that limit the likelihood of loops [10].

Router Architecture

Faster IP restorations can be achieved by speeding up the processing of routing messages and updating the forwarding information at a router. This involves software and hardware modification of existing router architecture. As described earlier, two critical steps in the update of a router's forwarding information are:

- Recomputation of shortest paths to all other routers
- Dispatching new forwarding information to all network interface cards

These steps are particularly time consuming, and vendors have introduced the following modifications in router architecture to speed up the process.

Incremental SPF — Instead of recomputing all shortest paths from scratch for every topological change, a router can use an *incremental shortest path first* (iSPF) algorithm that starts with the existing set of shortest paths and only performs the necessary incremental updates. These incremental algorithms have been known for many years, but have not been widely deployed due to their complexity. However, advances in memory and CPU technologies in modern routers have made iSPF a viable solution for today's ISP networks. Experiments in the Sprint network indicate that iSPF could reduce shortest path recomputation time to an average of less than 50 ms from the current 100–400 ms range.

Prioritizing IP Network Prefixes Updates — The transmission of the new forwarding information for a set of IP network prefixes from a router's central processor to all its interface cards can be prioritized according to the relative importance of each prefix [10]. This priority could be based on the network to which the prefix belongs (e.g., prefixes internal to a network and learned through IS-IS are more important than external prefixes learned through Border Gateway Protocol) or the amount of traffic carried by the prefix.

Conclusion

IP networks have traditionally been designed to provide uninterrupted service in the face of failures. This capability is provided by dynamic routing protocols that are able to detect and find alternate paths around failures. Moreover, the goal has been to build survivable IP networks without any assumption about the protection or restoration capabilities of the technology underlying the IP layer.

In this article we show that it is possible to achieve significant improvements in restoration time by tuning the timers that govern failure recovery. We also outline current trends in traffic engineering, IP routing protocols, and router architectures that can improve performance further.

We conclude that IP-based restoration is viable for the Sprint IP backbone and has several advantages. It provides high network availability at a small fraction of the cost of building protection or restoration mechanisms in the underlying optical layer. At the same time, it is ideally suited to the IP design paradigm: loosely organized networks with best effort delivery and no admission control.

References

- [1] A. Fumagalli and L. Valcarengi, "IP Restoration versus WDM Protection: Is There an Optimal Choice?" *IEEE Network*, vol. 14, no. 6, Nov. 2000, pp. 34–41.
- [2] L. Sahasrabudde, S. Ramamurthy, and B. Mukherjee, "Fault Management in IP-over-WDM Networks: WDM Protection vs. IP Restoration," *IEEE JSAC*, vol. 20, no. 1, Jan. 2002.
- [3] D. Oran, "OSI IS-IS Intra-domain Routing Protocol," RFC 1142, Feb. 1990.
- [4] J. Moy, "OSPF v. 2," RFC 2328, Apr. 1998.
- [5] G. Iannaccone *et al.*, "Analysis of Link Failures in an IP Backbone," *Proc. ACM Sigcomm Internet Measurement Wksp.*, Nov. 2002.
- [6] R. Kuhn, "Sources of Failure in the Public Switched Telephone Network," *IEEE Comp.*, vol. 30, no. 4, Apr. 1997.
- [7] ANSI, "Technical rep. on Enhanced Network Survivability Performance," T1.TR.68, Feb. 2001.
- [8] B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS Weights in a Changing World," *IEEE JSAC*, vol. 20, no. 4, May 2002.
- [9] A. Nucci *et al.*, "IGP Link Weight Assignment for Transient Link Failures," *Proc. 18th Int'l. Teletraffic Cong.*, Aug. 2003.

- [10] C. Alaettinoglu and A. Zinin, "IGP Fast Reroute," IETF Routing Mtg., Atlanta, GA, Nov. 2002.

Biographies

GIANLUCA IANNACCONI (gianluca.iannaccone@intel.com) received his Ph.D. degree in computer engineering from the University of Pisa, Italy, in 2002. From 2000 to 2001 he was a student visitor at Sprint Advanced Technology Laboratories. He joined Sprint in 2001 as a research scientist in the IP Research group. Recently, he moved to Intel Research, Cambridge, United Kingdom. His current research interests are network monitoring system design, traffic analysis and characterization, and availability and survivability of IP networks.

CHEN-NEE CHUAH (chuah@ece.ucdavis.edu) received her B.S. in electrical engineering from Rutgers University, and her M.S. and Ph.D. degrees in electrical engineering and computer sciences from the University of California, Berkeley, in 1997 and 2001, respectively. From 2001 to 2002 she was a postdoctoral researcher at Sprint Advanced Technology Laboratories. She is currently an assistant professor in the Department of Electrical and Computer Engineering at the University of California, Davis. She received the NSF CAREER Award in 2003 for her research on robust, secure, and stable routing.

SUPRATIK BHATTACHARYA (supratik@sprintlabs.com) holds an engineering degree from Jadavpur University, India, and M.S. and Ph.D. degrees in computer science from the University of Massachusetts, Amherst. Since completing his Ph.D. in 1999, he has been a member of the IP Research Group at the Sprint Advanced Technology Laboratory. His research interests include Internet measurement and monitoring, routing, traffic engineering, and multicast.

CHRISTOPHE DIOT (christophe.diot@intel.com) received a Ph.D. degree in computer science from INP Grenoble in 1991. From 1993 to 1998 he was a research scientist at INRIA Sophia Antipolis, working on new Internet architecture and protocols. From 1998 to 2003 he was in charge of IP research at Sprint Advanced Technology Laboratories. He recently moved to Intel Research, Cambridge, United Kingdom. His current interests are measurement techniques and Internet architecture.