

Feasibility of IP Restoration in a Tier-1 Backbone

Gianluca Iannaccone, Chen-Nee Chuah, Supratik Bhattacharyya, Christophe Diot
Sprint Advanced Technology Laboratories
1 Adrian Court, Burlingame CA 94010

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 paper 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 by 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.

I. INTRODUCTION

The core of the Internet consists of several large networks (often referred to as tier-1 backbones) that provide transit services to the rest of the Internet. A tier-1 backbone has a layered network architecture, with an optical network at layer 1 providing physical connectivity. The Internet Protocol (IP) layer which provides traditionally resides at layer 3, over an intermediate layer 2 technology (e.g., ATM). However, the current trend is to place the IP layer directly over WDM optical networks. Various transport protocols run above the IP layer to support a multitude of user applications.

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 that the ARPANET was built [1].

There are two main approaches for providing network survivability in IP backbones, namely protection and restoration [2], [3]. *Protection* is based on fixed and pre-determined failure recovery: as soon as a working path is setup, 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 the 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 milliseconds [2]. On the other hand, the traditional restoration mechanism at the IP layer is re-routing, 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 that a network operators seeks protection for. 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 that are currently used by service providers for failure recovery. With the rapid growth of IP backbones, it is no longer economically viable to use optical layer protection mechanisms for the entire network. Installing new optical fibers to provide protection paths has been hampered by several problems. First installing additional fiber infrastructure requires substantial capital investment. Second, the process of installing new fiber path is extremely slow as compared to just increasing the transmission capacity of a single fiber. In addition, geographic limitations (mountains, bridges, etc.) has made it difficult to install protection paths with sufficient diversity. As an alternative to optical layer protection, tier-1 ISPs have opted for IP level restoration based on path re-computation through IP routing protocols, such as ISIS [4] and OSPF [5].

In this paper we examine the feasibility of IP restoration for Sprint’s tier-1 IP backbone. The specific questions that 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 only IP restoration, coupled with adequate capacity provisioning and careful network design, is indeed effective in providing network survivability for the Sprint network.

Section II introduces some of the concepts and terms related to IP backbone networks and illustrates how network design is influenced by specific IP restoration techniques. Section III presents data on the magnitude, frequency and impact of failures on a large IP backbone. Section IV describes the performance of the IP restoration mechanisms and the results of several experiments on the Sprint operational network. Then, Section V illustrates the new solutions that are under investigations to overcome some of the limitations of IP routing protocols. Finally, Section VI concludes the paper.

II. COMPONENTS OF IP-BASED SURVIVABILITY

Network survivability in Sprint's IP backbone is provided solely by restoration mechanisms at the IP level. In the event of a network resource¹ failure, the routing protocol at the IP level computes an alternate path for traffic around the failure. This approach imposes several requirements for 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 re-routed and the number of nodes that need to modify their packet forwarding decisions.
- **Satisfying Service Level Agreements (SLA).** SLAs define performance guarantees that ISPs offer to 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 intra-domain routing protocol. We now discuss each of them in detail.

A. Capacity provisioning

IP is a connectionless, best-effort protocol that does not provide resource reservation or access control. Therefore an IP network has to carry the same volume of traffic even when its capacity is reduced by failures. In addition, an ISP such as Sprint has to adhere to the performance metrics guaranteed in its customer SLAs at all times. 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”[6].

The engineering rule for capacity provisioning in the Sprint network is to maintain the average utilization of any link under

50%. 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 [7], [8], [9].

Following this engineering rule, the Sprint's network has been provisioned to prevent service disruptions even when multiple links or routers fail. 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. Multi-homed customers that buy Internet connectivity from several ISPs may direct all of their traffic towards the ISP that is the least affected. Therefore, this ISP will experience a sudden surge in traffic volume while coping with multiple link and node failures. Provisioning redundant capacity is the only way to ensure network availability during such events. A similar approach has been used for survivability in other types of networks (e.g., telephone networks).

B. Network Topology

Sprint's IP backbone consists of a collection of Points-of-Presence (PoPs) connected via high-speed OC-48 (2.5 Gbps) and OC-192 (10 Gbps) links. This “logical” IP network is layered over an 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 a large metropolitan area). Each PoP connects customers, ranging from large corporate networks to regional ISPs and web-servers, to the Sprint network. PoPs also connect to other large backbone networks via private links or through public Network Access Points (NAPs).

Intra-PoP topology. Figure 1 shows a simplified view of the typical PoP topology in the Sprint 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 the backbone routers. The backbone routers provide connectivity to other PoPs and to other large backbone networks (peers). For failure 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 co-located. 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 in Section II-C).

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. In addition, fully meshed networks provide protection from cascading

¹Network resources include links, routers, router line-cards, optical equipment, etc.

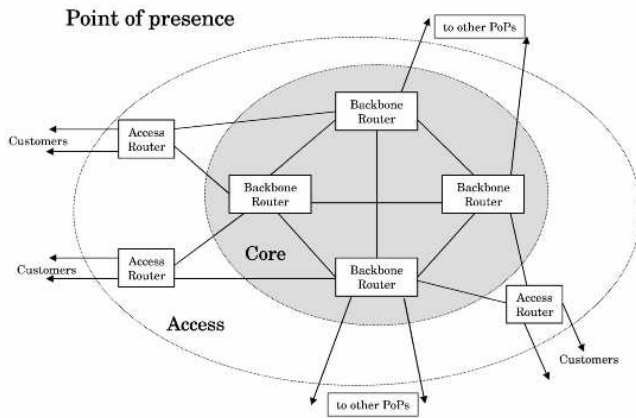


Fig. 1. Intra-PoP topology

failures [10], where the failure of one node triggers the failure of other nodes. The intra-PoP topology helps to distribute the traffic load homogeneously among backbone routers. Hence, in the case of router failure, the additional load is easily shared among the remaining routers.

Inter-PoP topology. A fully meshed inter-PoP topology would guarantee the availability of several alternate paths 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 alternative, 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 of the PoPs from the rest of the network.

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 a 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. 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 being 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 disjoint as possible. Unfortunately, the available fiber paths are 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, pipelines, etc., thereby getting limited to relatively few paths.

Furthermore, the mapping of IP links must also meet the SLA requirements. End-to-end delay across the Sprint network

is a key SLA metric. Current capacity provisioning practices in the Sprint network result 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 [8], [7]. Hence the length of the fiber path needs to be taken into account when addressing the IP-to-fiber mapping problem.

C. Intra-domain Routing

Sprint's IP network uses the link state protocol ISIS[4] for intra-domain 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 ISIS 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 towards another node along the minimum cost path which is computed using Dijkstra's shortest path first (SPF) algorithm. Thus setting link weights is the only way of controlling traffic flow through the network.

In the event of a link failure², 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, then ISIS allows traffic to be split evenly among these paths. This capability of ISIS is referred to as Equal Cost Multi-Path (ECMP), and is a key component of IP restoration. Traffic can be split among the equal cost paths on a per-packet basis or on a 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 [11], or short-term overload conditions [12].

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.

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

III. NETWORK FAILURES

The first step towards defining and measuring 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 and in this Section we present some observations based on the analysis of routing protocol messages in the 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 in Section IV.

A. General failure characteristics

We use the PyRT listener³ to collect ISIS 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 ISIS routing messages are always broadcast 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 [11].

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 the loss of connectivity at the IP level may be triggered by a variety of causes such as a fiber cut, router interface failure, ISIS protocol malfunction, etc. 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 the "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 5 month period (April – August 2002) on a weekly, daily and hourly basis⁴. 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 every day operations and not just as extraordinary events.

In Figure 2, we have considered both failures due to scheduled maintenance and due to unplanned events or accidents. It is interesting to investigate on 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 the time of day sheds light on this issue. In Figure 3 we show all the failures (over the 5 months of interest) grouped in two-hour bins by time of day (where the time-zone is US Eastern Time). We

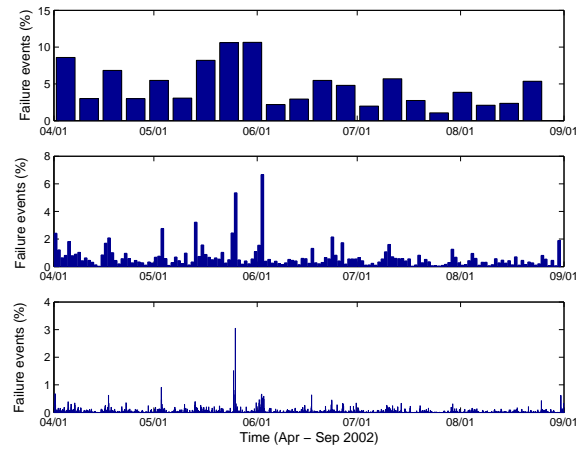


Fig. 2. Failure notifications over three time scales: weekly (top), daily (middle) and hourly (bottom)

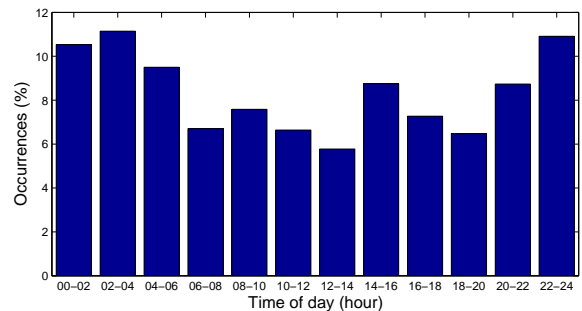


Fig. 3. Failures by time of day (2 hour windows)

observe that about 45% of the failure events occur between 10 PM ET and 6 AM ET. If we take into account the three hour time difference between the east and west coasts of the USA, then this is the time window during 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 that we observe. However, failure events during this period are likely to have less of an impact on traffic, because the backbone is relatively lightly loaded at night.

B. Sources of failures

It is not possible to identify exactly the factors causing failures from ISIS 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 re-routed over an alternative 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 (e.g., diggers damaging a fiber conduit) require several hours to repair [13]. On the other hand, a router may take several minutes to reboot

³<http://ipmon.sprint.com/PyRT>

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

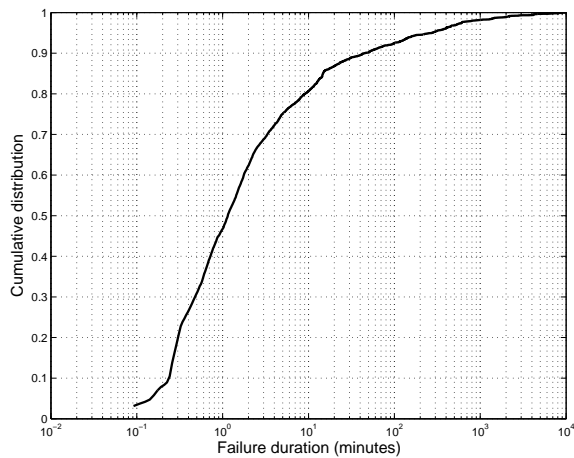


Fig. 4. Duration of failure events

and restart and tens of seconds to reset an interface card [7]. Software errors or routing processor overloads may also induce a router to erroneously infer the loss of connectivity with one of the neighboring routers; in these cases, repair can be just a matter of seconds.

We consider that a failure event starts with the reception of the *first* routing message that reports the loss of connectivity. The failure event ends (i.e. it is considered repaired) when the listener receives a “link up” message from *both* ends of the link. Although, our definition of failure duration is in compliance with the ISIS protocol specifications [4], it may not completely capture the time needed to repair. For example, in the case of loss of one of the “link up” messages we may overestimate the time needed to repair.

Figure 4 shows the cumulative distribution of the duration of failures over the 5 month period of interest and provides some insights into the sources of network failures:

- Only 10% of the failures last longer than 45 minutes. These failures are probably related to hardware failures (optical fibers, interface cards) that required human intervention.
- About 40% of the failures are repaired in one to fifteen minutes. Given the short time to repair, we conjecture that these events were caused by software errors and required just a router reboot or interface reset.
- About 46% of all failure events last less than a minute. Possible reasons comprehend an overloaded router that fails to process the keep-alive messages and mistakenly considers the 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% of the failures are repaired in 15 to 45 minutes and, most likely, have involved human intervention (e.g. substitution or maintenance of hardware equipment).

Therefore, a large portion of the failures (about 40%) are very likely related to the IP layer and, therefore, they can only be addressed via IP restoration mechanisms. This finding justifies the interest in identifying and deploying fast IP restoration techniques rather than relying mainly on optical protection.

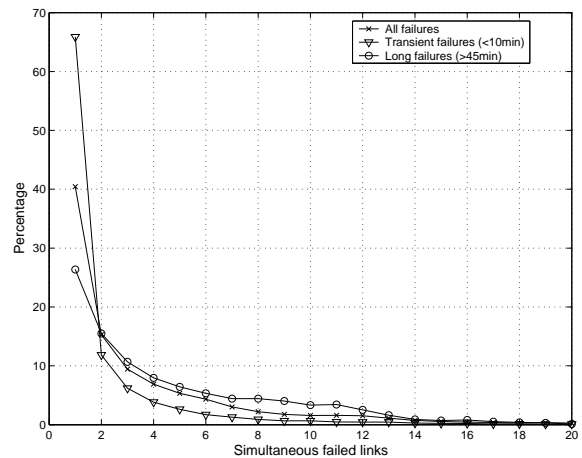


Fig. 5. Distribution of simultaneous failures

We believe that given the high percentage of failures that are related to IP equipment, IP restoration mechanisms should react as quickly as possible to failure notifications without waiting for the response from lower layers.

A second 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 minutes for 86% of the cases). We refer to all the failures that last less than 10 minutes as *transient failures*.

Investigating further on transient failures, 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 Figure 5 represents the fraction of time (with 1s 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 minutes) and all the failures. We observe that almost 70% of the transient failures are isolated. On the other hand longer failure event are more likely to happen in clusters: only less than 30% of them are isolated.

The presence and magnitude of transient failures has significant implications on the design of mechanisms for traffic engineering. In particular, short-lived link overloads due to the temporary re-routing of traffic over restoration paths has to be expected [12]. In Section V we will explore some proposed solutions that specifically address and factor in transient failures and topology changes.

IV. PERFORMANCE OF IP RESTORATION

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

The major weakness of IP restoration mechanisms has generally been considered their large restoration times when compared to optical protection mechanisms [2], [3]. However, the appropriate restoration time should only be dictated by the value that allows ISPs to accommodate the most common

mission-critical applications. Although SONET networks are required to restore connectivity in less than 50ms [14], it has been observed that longer restoration times have a minor impact on networking applications. In particular, ANSI [15] reports that restoration times below 200ms are unlikely to have any impact on services (including voice services), while delays between 200ms and 2s would only slightly degrade performance on voice services. Sub-second restoration is commonly regarded as a valid practical threshold.

Historically, IP restoration times have been inflated by the fear of threats to the routing stability [16]. In fact, there is a clear trade-off between restoration time and routing stability. A fast reaction to failure notifications may lead to routing instabilities given that routers will spend most of their CPU time updating their forwarding information. This could lead to routing inconsistencies resulting in routing-loops and unnecessary packet drops.

Until recently, network stability was seen as a much more pressing need by network users rather than fast failure recovery. This definition of priorities influenced the design of routing equipment, where most of the resources were devoted to improve the performance of packet forwarding and routing protocol stability rather than hardware reliability and failure recovery. Recent improvements in the router reliability and capacity to handle heavier routing processing loads, as well as high demand for mission-critical applications with stringent time constraints (e.g. voice services or virtual private networks) has overcome previous stability concerns.

IP restoration can be divided into three basic phases: i) failure detection, ii) failure notification to the control plane and to the other network entities, and, iii) forward path re-computation 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 with more than 1,000 routers, Points of Presence in almost 60 large metropolitan areas and 4 continents.

Our observations are also based on real-world experiments that involved to shut down some links inside the Sprint operational backbone network [11]. All figures we provide are based on the ISIS protocol although similar observations apply to OSPF.

A. Failure detection

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

- **Sonet framing** provides the information on the link connectivity for all point-to-point connections and is capable of raising an alarm within 10-20 ms from the 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, in the order of 30-60 seconds. 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 major ISPs 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.

Furthermore, on modern routers packet forwarding takes place directly in hardware on the interface cards, and it is, in general, not affected by failures of the routing software. Therefore, if keep-alive messages are used only to detect software failures, the impact on the packet forwarding of the long detection time is minimal.

B. Failure notification

Once a failure is detected, a router needs to decide when to notify the ISIS 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 presence of protection mechanisms at the optical layer that may restore the connectivity before the intervention of the routing protocol. For this reason, a recommended value for this timer has been 2s. 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 represents a fast and reliable way of propagating the information. The goal is to have all the routers in the network update their link database and trigger the re-computation of all the routes. The news about the failures need to propagate through the network as fast as possible in order to prevent routers from 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. Note that there is no differentiation based on the content of the LSP (e.g. loss or repair of connectivity). The recommended value for this timer is 50ms, but given today's router's processing power and link capacities it can be set down to 1ms without risks for the network.

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 10ms per hop. Given that the network is designed to allow localized failure recovery (see Section II) we expect that LSPs need to traverse one or two routers before restoration takes place.

C. 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 *routing tree*). In a network with n nodes and l links the complexity is $O(l \log n)$.

The computation of the shortest paths is therefore a very processor-intensive task and may require anywhere between 100ms and 400ms to complete for a network of the size of

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

TABLE I
COMPONENTS OF THE FAILURE RESTORATION TIME

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 the router has knowledge of.

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 seconds, although given the processing power of current backbone routers that timer can be reduced down to few milliseconds.

In the case of distributed router architectures where the packet forwarding is performed in hardware (the case 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 one millisecond 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, the failure recovery time is dominated by the hold-off timers introduced to reduce the risk of network instabilities. In Table I we have summarized all the components of the failure recovery. For the timers, we have indicated two values, a default value that has been 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 the overall network stability. As we can see from the table, fine tuning the timers involved in IP restoration allowed Sprint to bring the recovery time down from 8-9 seconds to less than one second. 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 that regard.

As a last note, we have to point out that Multi-Protocol Label Switching (MPLS) [17] has been proposed as a viable alternative to circumvent most of the latency drawbacks of IP restoration. MPLS provides a mechanism to determine how packets should be forwarded through a network by attaching a layer-2 label to the IP packet. Such label-based switching

attempts to relieve a router from having to interrogate each IP header, hence optimizing packet processing overhead.

With MPLS, packet forwarding is based on labels local to the ISP's network and not anymore on the (globally assigned) IP destination addresses. Hence, an ISP can introduce additional labels to provision for alternate forwarding paths in case of failures. This technique is also called MPLS protection switching [18]. However, there are several concerns regarding the use of MPLS for large backbones:

- ISPs are required to identify in advance possible failure scenarios and to plan the failure recovery. Only the failure scenarios considered during the design will result in a lower recovery time.
- Large-scale outages that cannot be planned in advance will still need to be addressed using IP restoration.
- Failure detection and notification is still based on the same mechanisms used for IP restoration. Therefore, for all failures that require more than one router to participate to the restoration, the gain in recovery time is limited.
- Significant management complexity is introduced to administer a large network with hundreds of routers, resulting in higher operational costs for ISPs. In particular, two separate routing tables need to be administered, the IP routing table and the MPLS label table.

Concluding, we find that MPLS protection switching introduces the complexities and limitations of the traditional protection mechanisms without completely addressing the drawbacks of IP restoration.

V. TRENDS IN IP RESTORATION SOLUTIONS

In the previous section, we discussed how to fine tune ISIS 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 they have not yet been deployed in the Internet.

A. Traffic engineering solutions

Most large ISP networks use a link state protocol such as ISIS or OSPF for intra-domain routing. Such a protocol associates a weight (or cost) with each network link and then routes traffic along the minimum cost path between every pair of nodes. Thus, setting link weights is the primary traffic engineering technique for networks running ISIS or OSPF. Common traffic engineering objectives are to minimize end-to-end delays and balancing 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 daily basis. An obvious approach is to recompute and change link weights to reroute traffic in the event of a failure. Fortz et al. [19] 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, which may lead to network instability and hence aggravating the situation. Secondly, the high frequency of short, transient failures that we observed in the IP backbone (Section III) 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. [20] 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.

An alternative approach [12] proposes to use deflection routing to divert traffic to less loaded links if a link is overloaded during a transient failure. However, care has to be taken to prevent routing loops and inordinately long delays when using deflection routing. [12] proposes a solution that guarantees loop-free deflection paths by setting some loose constraints on the network topology and the assignment of link weights.

B. Routing protocol extensions

Several extensions have been proposed to ISIS and OSPF to improve restoration times. The most promising ones are:

Content-based rules for processing LSPs. As described in Section IV, the generation and propagation of LSP messages is rate-limited independently of the content of the LSP. An LSP that reports a link failure needs, however, immediate attention since ISIS 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 ISIS would have already found an alternate route around the failure. Therefore, LSPs that report loss of connectivity should receive a higher priority than those reporting link recoveries [16].

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 [21]. This expedites the process of switching to an alternate path in the event that a directly connected link fail. Such an approach may lead to routing loops, but it is possible to devise a network topology and a link weight assignment that limit the likelihood of loops.

Greater flexibility in ECMP traffic splitting. Current implementations of ISIS and OSPF only allow equal splitting of traffic on multiple equal cost paths to achieve load balancing. In the case of per-packet splitting, the same fraction of traffic is sent on each equal cost path. In the case of per-flow splitting, the source and destination addresses of each flow is hashed onto one of the equal cost paths using a uniform hash function. This constraint often makes it impossible to achieve truly optimal distribution of traffic. Sridharan, et al. [22] have proposed to remedy this problem by controlling the set of *next hops*

assigned to each routing prefix entries in the forwarding table, thereby permitting unequal load splitting without changing the routing protocols or the forwarding mechanisms.

C. Router architecture

Faster IP restorations can be achieved by speeding up the processing of routing messages and the update of the forwarding information at a router. This involves software and hardware modification of existing router architecture. As described in Section IV, two critical steps in the update of a router's forwarding information are: i) recomputation of shortest paths to all other routers, and ii) 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 re-computing all shortest paths from scratch for every topological change, a router can use an *incremental shortest path first* algorithm (iSPF) 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 they have not been widely deployed due to their complexity [23]. However, the advance in memory and CPU technologies in modern routers has made iSPF a viable solution for today's ISP networks. For the Sprint network, iSPF could reduce the shortest path recomputation time to an average of less than 50ms from the current 100-400 ms range.

Multicasting forwarding information to router interfaces. In distributed router architectures, the central route processor needs to disseminate the forwarding information to all the interface cards. In these architectures the transmission may slow down the restoration. This is particularly true if an unicast reliable transport protocol is used over the shared bus that connects the interface cards to the route processor. Instead of repeatedly sending the same forwarding information to each line card in turn, a faster and more efficient approach is to use a reliable multicast protocol for updating all the interfaces simultaneously. This can improve ISIS restoration time by significantly reducing the time it takes to update the packet forwarding information on line cards in the event of a failure.

Prioritizing IP network prefixes updates. The transmission of the new forwarding information for a set of a 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 [21]. This priority could be based on the network that the prefix belongs to (e.g., prefixes internal to a network and learned through ISIS are more important than external prefixes learned through BGP) or on the amount of traffic carried by that prefix.

VI. CONCLUSION

IP networks have been traditionally 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 paper, starting from the example of the Sprint network we have described how pure IP restoration is both effective and economical. We presented measurement results on link failures and service restoration times in order to evaluate the performance of IP-based restoration. We showed that it is possible to achieve significant improvements in restoration time by tuning the timers that govern failure recovery. We also outlined current trends in traffic engineering, IP routing protocols and router architectures that can improve performance further.

We conclude that an IP-based restoration approach is viable for an ISP backbone and has several advantages. It provides high network availability at a small fraction of the cost for building protection or restoration mechanisms in the underlying optical layer. It avoids the complexity and the associated operational costs and overheads associated with newer technologies such as MPLS. At the same time, it is ideally suited for the IP design paradigm - loosely organized networks with best-effort delivery and no admission control.

REFERENCES

- [1] D. Clark, "The design philosophy of the DARPA internet protocols," in *Proceedings of ACM Sigcomm*, Aug. 1988.
- [2] A. Fumagalli and L. Valcarengi, "IP restoration versus WDM protection: Is there an optimal choice?" *IEEE Network Magazine*, vol. 14, no. 6, pp. 34–41, Nov. 2000.
- [3] L. Sahasrabudde, S. Ramamurthy, and B. Mukherjee, "Fault management in IP-over-WDM networks: WDM protection versus IP restoration," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 1, Jan. 2002.
- [4] D. Oran, "OSI IS-IS intra-domain routing protocol," RFC 1142, Feb. 1990.
- [5] J. Moy, "OSPF Version 2," RFC 2328, Apr. 1998.
- [6] L. W. McKnight and J. P. Bailey, *Internet Economics*. MIT Press, 1998.
- [7] C. Boutremans, G. Iannaccone, and C. Diot, "Impact of link failures on VoIP performance," in *Proceedings of NOSSDAV*, May 2002.
- [8] K. Papagiannaki, S. B. Moon, C. Fraleigh, P. Thiran, F. Tobagi, and C. Diot, "Analysis of measured single-hop delay from an operational backbone network," in *Proceedings of IEEE Infocom*, New York, USA, June 2002.
- [9] C. Fraleigh, F. Tobagi, and C. Diot, "Provisioning IP backbone networks to support latency sensitive traffic," in *Proceedings of IEEE Infocom*, Mar. 2003.
- [10] A. E. Motter and Y.-C. Lai, "Cascade-based attacks on complex networks," *Phys. Rev. E*, vol. 66, no. 065102, Dec. 2002.
- [11] G. Iannaccone, C.-N. Chuah, R. Mortier, S. Bhattacharyya, and C. Diot, "Analysis of link failures in an IP backbone," in *Proceedings of ACM Sigcomm Internet Measurement Workshop*, Nov. 2002.
- [12] S. Iyer, S. Bhattacharyya, N. Taft, and C. Diot, "An approach to alleviate link overload as observed on an IP backbone," in *Proceedings of IEEE Infocom*, Mar. 2003.
- [13] R. Kuhn, "Sources of failure in the public switched telephone network," *IEEE Computer*, vol. 30, no. 4, Apr. 1997.
- [14] American National Standard for Telecommunications, "Synchronous optical network (SONET) - Basic description including multiplex structure, rates and formats," ANSI T1.105, 1995.
- [15] —, "Technical report on enhanced network survivability performance," ANSI T1.TR.68, Feb. 2001.
- [16] D. Katz, "Why are we scared of SPF? IGP scaling and stability," NANOG, Toronto, Canada, June 2002.
- [17] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," RFC 3031, Jan. 2001.
- [18] T. Chen and T. Oh, "Reliable services in MPLS," *IEEE Communications*, vol. 37, no. 12, p. 58.
- [19] B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS weights in a changing world," *IEEE Journal on Selected Areas in Communications*, Feb. 2002.
- [20] A. Nucci, B. Schroeder, S. Bhattacharyya, N. Taft, and C. Diot, "IS-IS link weight assignment for transient link failures," Sprint ATL, Tech. Rep. TR02-ATL-071000, July 2002.
- [21] C. Alaettinoglu and A. Zinin, "IGP fast reroute," IETF Routing Meeting, Atlanta, GA, Nov. 2002.
- [22] A. Sridharan, R. Guerin, and C. Diot, "Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks," in *Proceedings of IEEE Infocom*, Apr. 2003.
- [23] R. Perlman, *Interconnections: Bridges and Routers*, 2nd ed. Addison-Wesley Publishing Company, 2000.