Achieving Sub-Second IGP Convergence in Large IP Networks*

Pierre Francois
Dept CSE, Université catholique de Louvain (UCL), Belgium
francois@info.ucl.ac.be

Clarence Filsfils and John Evans
Cisco Systems
{cf,joevans}@cisco.com

Olivier Bonaventure
Dept CSE, Université catholique de Louvain (UCL), Belgium
Bonaventure@info.ucl.ac.be

ABSTRACT

We describe and analyse in details the various factors that influence the convergence time of intradomain link state routing protocols. This convergence time reflects the time required by a network to react to the failure of a link or a router. To characterise the convergence process, we first use detailed measurements to determine the time required to perform the various operations of a link state protocol on currently deployed routers. We then build a simulation model based on those measurements and use it to study the convergence time in large networks. Our measurements and simulations indicate that sub-second link-state IGP convergence can be easily met on an ISP network without any compromise on stability.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Routing protocols;
C.2.6 [**Internetworking**]: Routers

General Terms

Measurement, Experimentation, Performance

Keywords

Intradomain routing, IS-IS, OSPF, convergence time

1. INTRODUCTION

OSPF and IS-IS are the link-state (i.e. LS) Interior Gateway Protocols (i.e. IGP) that are used in today's IP networks. Those protocols were designed when IP networks were research networks carrying best-effort packets. Their initial goal was to allow the routers to automatically compute their routing and forwarding tables without consuming too much CPU time during network instabilities. This explains why, until recently, the typical LS IGP convergence in Service Provider (i.e. SP) networks used to be in tens of seconds [1, 16]. Nowadays, following the widespread deployment of real time applications such as VoIP and the common use of Virtual Private Networks (VPN), much tighter Service Level Agreements (SLA) are required, leading to LS IGP convergence requirements from sub-3-second to sub-second. [11, 10].

This paper shows that sub-second LS IGP convergence can be conservatively met on a SP network without any compromise on stability.

*This work was supported by Cisco Systems within the ICI project. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of Cisco Systems.

The paper is structured as follows: we firstly provide an overview of a typical IS-IS convergence. While for ease of reading we use the IS-IS terminology throughout the paper, the analysis equally applies to OSPF. We then characterise each of the components of the convergence on a single router in terms of its execution time and robustness against unstable network conditions. Next, we build a simulation model based on those measurements and use it to evaluate the convergence time in two large SP networks and the influence of the characteristics of the network itself on the convergence.

2. LINK-STATE IGP CONVERGENCE

A detailed description of IS-IS can be found in [26]. IS-IS supports a multi-level hierarchy, but as most large ISPs running IS-IS use a single level, we do not consider it in this paper.

The overall operation of an IS-IS router can be sketched as follows. First, the router will exchange HELLO PDUs with its neighbours to determine its local topology. The router will then describe its local topology inside a link-state packet (LSP) that will be reliably flooded throughout the network. This LSP will contain at least the identifier of each neighbour, associated with the metric of the directed link from the router to this neighbour. Note that a mechanism called two-way connectivity check authorises routers to use a link (i.e. to consider it as being part of a path to a given destination) only if both adjacent routers describe it as being up and running [18]. The LSP will typically also contain information about the IP addresses attached to the router as well as various optional parameters such as the Traffic Engineering information [26]. When broadcast networks, such as Ethernet LANs, are used, the situation is slightly different. On each broadcast network, the IS-IS routers attached to the network will elect a designated router. This router will "represent" the broadcast network and will generate a LSP describing this network and the routers attached to it. Thus, an IS-IS router attached to several broadcast networks may generate several LSPs.

With IS-IS, two types of events can force a router to flood a new LSP. First, a new LSP is generated and flooded each time the information contained in the LSP (neighbours, IP addresses, metrics, TE information, ...) changes. Second, to avoid problems in case of undetected memory or transmission errors, each LSP has a lifetime. Upon expiration of the LSP's lifetime, its parent router must flood it again. While the IS-IS specification did mention a default lifetime of 20 minutes, in practice, large SP's usually set it to its maximum value (i.e. 18 hours) to reduce the background flooding noise.

To ensure that LSPs are reliably flooded throughout the network, each LSP is acknowledged on each link. When the IS-IS specification was written, the link speeds were much lower (i.e. T1) and the

CPU's were much slower. Hence, in order to prevent LSP Update packets from congesting links and overloading neighbours' CPU's, a pacing timer of 33ms was specified between any two consecutive LSP transmissions on the same link.

Once a LSP describing a topology change has reached a router, this router updates its Link State Database (LSDB) which triggers a request to update the routing table (i.e. commonly called Routing Information Base, RIB). To update its RIB, a router must compute is Shortest Path Tree (SPT) based on the information stored in the LSDB. The RIB update itself triggers the update of the Forwarding Information Base (FIB). The FIB is a copy of the RIB that is optimised for forwarding efficiency. On distributed platforms, the convergence process ends with the distribution of the FIB modifications to the various linecards of the router.

In summary, a typical IS-IS convergence can be characterised as $D + O + F + SPT + RIB + DD$ where D is the link failure detection time, O is the time to originate the LSP describing the new topology after the link failure, F is the complete flooding time from the node detecting the failure (i.e. Failure node) to the rerouting nodes that must perform a FIB update to bring the network in a consistent forwarding state, SPT is the shortest-path tree computation time, RIB is the time to update the RIB and the FIB on the main CPU and DD is the time to distribute the FIB updates to the linecards in the case of a distributed router architecture.

3. COMPONENTS OF THE CONVERGENCE TIME

This section characterises each convergence components in terms of its execution time and its robustness against unstable network conditions.

3.1 Router Architecture, Processor Performance, Operating system

As we will see later, the main convergence bottleneck is the RIB component. It is thus clear that the faster the processor, the faster the convergence.

A distributed router architecture with hardware packet processors is very well suited for faster IS-IS convergence as it dedicates all its CPU power to the sole control plane operation: the central CPU (also called RP) handles all the routing protocol operations (IGP, BGP, RIB, FIB) and downloads the FIB update to the CPU's on the linecards which write them into the hardware packet processors. The operating system run by the RP and LineCard (LC) CPU's implements a process scheduler with multiple priorities and preemption capabilities. This allows for example for the IS-IS process to be scheduled immediately upon link failure even if a process of lower priority was running at that time (i.e. a maintenance process).

During a convergence on a distributed platform, at least two processes of the same priority must share the CPU: the IS-IS process to update the RIB and FIB, and the so-called IPC process to distribute the resulting FIB modifications to the LC CPU's. The RIB update being the key bottleneck, prioritisation techniques have been developed to ensure that IS-IS starts the RIB update with the most important prefixes. To ensure that these most important modifications are immediately distributed to the linecards, a small process quantum is often used (i.e. $50ms$). In practice, this leads to the following operation: immediately after completing SPF, IS-IS starts updating the RIB with the most important prefixes. When the $50ms$ quantum is over, the IPC process is scheduled and these most important updates are distributed to the linecards. When the $50ms$ quantum is over, the IS-IS process is rescheduled and the RIB updates continues followed by the IPC distribution and so forth. In the worst-case, in very large networks with lots of IS-IS prefixes, ten or more such rounds may be required which would lead to worst-case convergence time for the last prefix over the second. The use of this RIB update prioritisation technique and the parallelism between the RIB update and the FIB distribution to the linecards ensure that the most important prefixes are updates well under the second as we will see later.

3.2 Link Failure Detection

The dominant use of Packet over SDH/SONET (POS) links in SP backbones and hence the ability to detect a link failure in a few tens of milliseconds is a major enabler of sub-second IGP convergence.

Inbuilt mechanisms in SDH and SONET (Loss of Signal (LOS), Loss of Frame (LOF), Alarm Indication Signal (AIS) etc.) allow the linecard hardware to detect the failure in less than 10 milliseconds [24]. Immediately thereafter, a high-priority interrupt is asserted on the LC CPU which causes a POS routine to be executed. This routine enforces a user-specified hold-time which, if configured, delays the communication of the failure to the central CPU to allow for the SONET/SDH protection to occur (sub-$50ms$ in most cases, sub-$110ms$ in extreme conditions). If such protection is not provided by the SONET/SDH network, then the user will not configure any such delay and the failure will immediately be signalled to the common CPU [8]. This latter will update the interface status and hence schedule IS-IS for reaction. We have instrumented a Cisco 12000 router with a GRP[1] processor and Eng2 PoS linecards. The unit under test (i.e. UUT) was running a modified software image that contains additional probes. This instrumented software image allowed us to perform white-box measurements. We verified that this modified software image had the same performance as a normal software image. We inserted the UUT in an emulated IS-IS network of 700 nodes, 3000 links and 2500 prefixes. It was running BGP with 160000 routes. SNMP probes were sent to the UUT to obtain an average 5-min CPU utilisation of 30% (it is rare for such routers to have an average CPU utilisation higher than 10%). On top of this excessive load, 16 BGP flaps per second were continuously sent to further stress the router. We repeated 5000 POS failures and measured the delta time between the high-priority interrupt on the LC CPU and when the IS-IS process is scheduled on the main CPU.

The objective of the test is to characterise, in a loaded distributed-architecture router, how much time is added by the software infrastructure to the sub-$10ms$ failure detection provided by the SONET hardware.In the lab, the measured Percentile-90 was $8ms$ (for a total worst-case detection of $10 + 8 = 18ms$). The measured percentile-95 was $24ms$ and the worst-case measurement was $51ms$.

This confirmed the theoretical expectation: in almost all the cases, the rapid SDH/SONET hardware detection on the LC is complemented with a prompt signalling from the LC CPU to the main CPU leading to an overall detection of less than $20ms$. When the control plane load increases, although very rare as confirmed in our test results, it becomes possible that another process was owning the CPU when the detection occurred on the LC. In the worst-case, this process is, first, of the same priority than IS-IS (i.e. BGP); second, it was scheduled just before the failure occurred; third, it is busy enough to consume its full quantum of $50ms$.

While POS represents the vast majority of link types between

[1]In reality, most such types of routers are now equipped with PRP2 processors which are more than twice as fast as the GRP and with more recent linecards with much faster LC CPU's. This slower hardware combination was chosen to emphasise the conservative property of the analysis.

routers, the same sub-$20ms$ property was confirmed for two other common link types: back-to-back Gigabit Ethernet and Spatial Reuse Protocol (SRP).

When SONET/SDH link or path alarms are cleared (indicating a link or node recovery), timers are used to hold the interface down for an additional 10s before the routing protocols are informed to ensure robustness against unstable situations such as flapping links. Router vendors have generalised the interface state dampening concepts to non-POS links and have extended it with adaptive timers, which can change their rate of responsiveness based upon the stability of the interface [5]. This "dampening" of good news protects the routing protocol from network instability, caused by flapping links for example.

For link-layers which do not have such a link management capability, the worst-case time to detect a failed neighbour is dependent upon the hello mechanism of the IGP. With the use of faster IGP hellos [19], the worst-case time to detect neighbour failure can be much reduced, resulting in improved convergence times. This protocol has been however built mainly for adjacency discovery and parameter negotiation and is most often supported on the central processor card of a distributed-architecture router. It is thus unlikely that very fast failure detection may be achieved, as it would require an intensive use of the central processor.

To implement faster hello's, Bidirectional Forwarding Detection (BFD) can be used [13]. The main advantage of BFD over the Hello messages of IS-IS is that BFD can be easily implemented on the linecards themselves. Thus, shorter time limits can be set and a fast detection is possible without major impact on the main CPU.

In conclusion, the majority of the router interconnects benefit from very fast failure detection (sub-$20ms$) without any compromise on stability.

3.3 LSP Origination

A rapid dissemination of updated Link State Packets is essential for rapid convergence. But an unstable device can lead to the generation of an excessive number of LSPs.

Traditionally LSP generation timers have been statically defined, that is they were set to fixed values [18]. Those statically defined timers have been set to limit the routing protocol overheads incurred during times of network instability, more precisely when links flap. This consequently also impacts the convergence times that can be achieved in a stable network.

To overcome this problem and to achieve both rapid and stable convergence, dynamic, rather than static, timers have been introduced to control the LSP generation process [16]. The concept of dynamic timers is that they can adapt their duration and hence responsiveness depending upon the stability of the network: when the network is stable, the timer is short and ISIS reacts within a few milliseconds to any network topology changes; in times of network instability, however, the timer exponentially increases in order to throttle the rate of ISIS response to network events. This scheme ensures fast exchange of routing information when the network is stable (down to a few ms to 10's of ms) and moderate routing protocol overhead when the network is unstable, thus allowing the network to settle down.

The duration between when ISIS is scheduled and the LSP generation is finished was measured on the previously described testbed: the measured percentile-50 and -100 were $8ms$ and $12ms$.

In conclusion, the origination time is extremely fast ($<= 12ms$) without any compromise on stability.

3.4 Flooding

The flooding time from the Failure node to the Rerouting nodes is the sum at each hop of the bufferisation, serialisation, propagation and the ISIS processing time.

Serialisation, the time taken to clock the packet on the link, is negligible on a SP backbone (1500-byte are sent in less than 5μs at OC-48 speed). Bufferisation is also negligible: most SP networks are capacity planned outside congestion [4] and routers prioritise routing updates through input and output buffers, as proposed notably in [23].

We will evaluate the impact of the propagation delay on the IGP convergence with the simulation model in section 4. We focus the remainder of this section on optimisations for fast flooding time per hop [6] and their lab characterisation.

First, a single-threaded IS-IS implementation must ensure that the LSP is flooded before the RIB is updated. Indeed, this latter can take several hundreds of millisecond and such a delay would jeopardise the overall network convergence when the local node is not the sole rerouting node.

A second important optimisation enabled with fast flooding behaviour is related to the pacing timer. The value of $33ms$ suggested by the IS-IS specification [18] is outdated by current link speeds (40G nowadays vs T1 15 years ago) and processor performance and is potentially quite damaging to the IS-IS convergence time. Indeed, upon a node failure, in the worst-case, all the LSP's of the neighbours of the failed node are required to compute the correct alternate path(s). Assuming a node with 10 neighbours, we see that with the default pacing timer suggested by the IS-IS specification, the last LSP could be unnecessarily delayed by $300ms$.

Fast flooding has been introduced to overcome the effects of pacing on convergence. Its ideal implementation bypasses pacing on LSPs that describe a new link-state change event, and applies pacing on Refresh and TE LSPs. Such an implementation requires that link flaps do not trigger bursts of LSP origination describing unstable link states. More conservative implementations of Fast Flooding let routers bypass the pacing on the same kinds of LSPs, but the burst size is controlled and pacing is re-applied by routers detecting that a configurable amount of LSPs have been fast flooded within a configurable amount of time [6].

In order to characterise the resulting fast-flooding behaviour, we send a LSP to the previously described UUT and measure the time until the same LSP is seen on its other interfaces. The measured Percentile90, 95 and 100 for 1000 measurements were respectively $2ms$, $28ms$ and $52ms$. As for the link failure detection, this worst-case is measured very rarely as it requires that a process of the same priority as ISIS was scheduled just before the event and was busy enough to consume its entire process quantum. In practice, the probability of occurrence will even be smaller and this worst-case should be neglected. Indeed, due to the meshing of the networks, several parallel paths exist between the failure and rerouting nodes and hence for the worst case to really occur, these conditions must occur at the same time along all the parallel paths.

In conclusion, we have shown that the time to flood one LSP is negligible compared to the sub-second convergence objective.

3.5 SPT Computation

The dynamic timers described in the context of controlling LSP generation in section 3.3 have also been applied to control the occurrence of SPF recalculations. This allows IGPs to be tuned such that when the network is stable, their timers will be short and they will react within a few milliseconds to any network topology changes. In times of network instability, however, the SPF timers will increase in order to throttle the rate of response to network events.

This scheme ensures fast convergence when the network is stable and moderate routing protocol processing overhead when the network is unstable.

The computational complexity of a typical implementation of the SPF algorithm is $O(nlog(n))$ where n is the number of nodes [9]. Therefore, in a network designed for fast IGP convergence it is best practise to minimise the number of nodes in the topology. For example, Ethernet connections used as point-to-point links between routers should be modelled by the IGP as point-to-point links rather than multi-access links to avoid introducing too many pseudo-nodes in the topology.

Incremental SPF (iSPF) [17] is an important algorithmic optimisation to SPF computation and hence should be considered for a faster IGP convergence [1]. iSPF analyses the impact of the new LSP/LSA on the previously computed SPT and minimise the amount of computation required. For example, if the change only involves "leaf" information, e.g. a new IP prefix has been added to node X, then the previous SPT is still correct and all what is required is to read the best path to node X and add an entry in the routing table for the prefix via that path. This operation is called partial route calculation and is notably described in [3]. Another straightforward example relates to link deletion. When the topological change does belong to the previous SPT, iSPF determines the subset of nodes impacted and restarts the SPT computation from there, reusing the non-impacted region of the previous SPT. The further away the failure, the smaller the impacted subset and hence the bigger the iSPF computation gain compared to a full SPF. Last but not least, if the link does not belong to the previous SPT then the whole SPF computation may be skipped, as the old SPT is still valid.

We varied the size of the IS-IS network connected to our UUT from 500 to 10000 nodes and measured the duration of a full SPT computation for each network size. In those topologies, the average router had 4 links. The obtained distribution showed a good linearity ($R^2 > 0.99$) with the cloud size: Full-SPT(PRP2 processor) $\sim$45μs per node. We obtained similar results on real ISP topologies. A network of 700 nodes (large by current standards) is thus computed in the worst-case in $31.5ms$. In practice, the computation will often be much faster than this thanks to the iSPF optimisation.

In conclusion, we have shown that the SPT computation is executed very fast (tens of milliseconds) and without any compromise on stability (dynamic throttle timers).

3.6 RIB and FIB update

The RIB/FIB update duration is linearly dependent with the number of modified prefixes[2].

Our UUT is once again used and a link failure is created such that all the 2500 prefixes from our topology are impacted by the failure. Packet generators create 11 streams, each of 1000 packets per second. The 11 streams are equally spread across the full table size (position1, 250, 500...2500).

We repeated the measurement 100 times and plot in figure 1 the percentile-0, 50, 90, 100 and the average update time. We repeated these tests with various processor speeds (GRP, PRP1, PRP2), various linecard types, local versus remote failure types and load balancing or not prior to the failure. Fig 1 provides the results when the UUT is equipped with a PRP1 processor, Eng4+ linecards, the failure is remote from the UUT and the UUT was not load-balancing before the failure.

As expected, the results primarily depend on the main processor

[2]Routers have been improved to only modify the impacted prefixes. In the past, in some cases, the full FIB was rewritten [22]

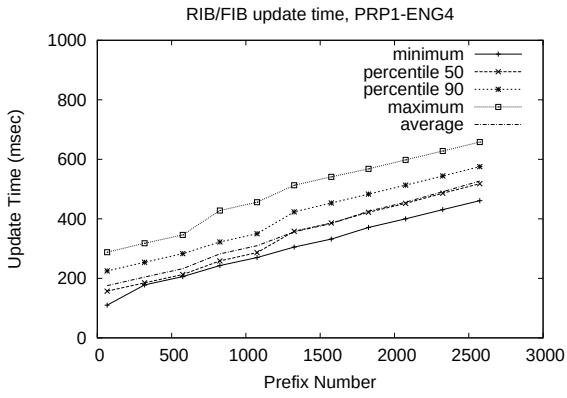

Figure 1: Minimum, Percentile-50, Percentile-90 and maximum RIB+FIB update time with PRP1 processor and Eng4+ linecards

performance (i.e. a PRP1 is twice more performant than the GRP. A PRP2 is faster than the PRP1) as this is the key bottleneck in the convergence process. The type of failure, the type of linecard, the load balancing state have a very moderate impact on the measured convergence and hence can be neglected in the remainder of this paper. A linear regression on the percentile-90 indicated a cost per routing table update of $\sim$146μs. This is the cost per RIB/FIB update.

Three approaches exist to minimise the RIB/FIB update component: network design rule to minimise the number of IGP prefixes, protocol and implementation optimisation to allow the prioritisation of some prefixes above others during RIB/FIB update and finally the intrinsic optimisation of the table management code. We will discuss here the two first approaches.

At the extreme, a designer could recognise that the only important prefixes that should be present in the IGP are those tracking premium content destinations (e.g. subnets with VoIP gateways) and BGP next-hops. In a large ISP, there are typically a few hundred of such important prefixes. All the other prefixes only track links interconnecting routers. These prefixes are rarely used as destination addresses since few hosts send packets directly to the interfaces of the routers. This information about the prefixes of the internal links could be advertised in iBGP. Unfortunately, many networks have not been designed like this as historically people did not care a lot about convergence. It is thus likely to see several thousands prefixes in the IGP of a large SP network while only a small fraction of them are really important. We thus face a problem where the RIB/FIB update component linearly scales by a number of several thousands while this number should in reality be much smaller.

Introducing prefix prioritisation solves this problem: the important prefixes are updated first and hence the worst-case RIB/FIB update duration now scales based on a much smaller number (the number of important IGP prefixes as opposed to the total number of IGP prefixes). Prefix prioritisation for IS-IS has been defined in [7]. It introduces three priorities (high, medium, low) and guarantees that the routing table is always updated according to these priorities. A default heuristic classifies the /32 prefixes as 'medium' priority and the other prefixes as 'low' priority. The /32 prefixes are indeed likely more important than other prefixes as they characterise BGP speakers and tunnel termination services (i.e. L2VPN). Finally, a customisation scheme based on IS-IS tagging is provided

(e.g. subnet with VoIP gateways can be classified as 'high' importance and hence will always be updated first).

3.7 Distribution Delay

As we saw previously, the router implementation is optimised to allow for the parallel execution of the routing table update on the central CPU and the distribution of these modifications to the linecards.

The distribution of this information may be further optimised with for example the use of multicast transport between the server (central CPU) and the clients (linecard CPU's).

Reusing once again the same testbed, we measured the delta time between when a prefix is updated on the central CPU and when the related entry is updated on the LC. As expected, this 'distribution delay' was measured to be on average less than $50ms$ and in the worst-case less than $70ms$.

4. SIMULATION MODEL

The previous sections identified all the factors that influence the convergence time inside each router. In a large SP network, the total convergence time will also depend on the network itself. To evaluate this dependance, we modified an OSPF implementation [12] for the SSFNet Simulator [21] to take into account the particularities of IS-IS and the white-box measurements presented earlier.

4.1 Router model

The measurements analysed in section 3 show that there are variations in the measured delays. Those variations are due to several factors such as the physical architecture of the router, the scheduler of the router's operating system, ... To take those variations into account, we modified the simulator to use a randomly chosen delay within a $[min, max]$ range each time an event duration is considered in the simulator.

The first component of our model is the time required to detect the failure of a link. For a low delay link, we use the lab measurements presented in section 3. For long delay links such as trans-oceanic links, we randomly select one location and take into account the time to propagate the failure detection signal from this location to the two routers. In both cases, the two routers attached to a link will not detect its failure exactly at the same time. Once a simulated router has detected a failure, it will originate a new LSP. We do not model the LSP generation timers in the simulator and allow the router to flood its LSP immediately. When a simulated router receives a LSP, it processes this LSP in [2,4]ms. Our router model supports both normal pacing and fast flooding as described in section 3.4. After the arrival of a LSP indicating a failure, a simulated router needs to decide when to perform the SPT computation. We model the exponential backoff mechanism described in section 3.5. This mechanism is configured with three parameters : $spf\_initial\_wait$, $spf\_exponential\_increment$ and $spf\_maximum\_wait$.

We model the SPT computation time as a function of the number of nodes in the network with some jitter to take into account the other processes that may be running on the router's CPU. We only consider the full SPT computation and do not model the incremental variants. To model the time required to update the FIB of a router, we first compute the number of prefixes whose FIB entries have changed. The FIB update delay is then obtained by multiplying the number of FIB entries to be updated with the time required to update one entry. Our simulator models two types of FIB updates : *full* and *incremental*. With the full FIB update, the simulated router updates the FIB entry of each prefix after a recomputation of the SPT. This corresponds to routers such as those analysed in [22]. With the incremental FIB update, the simulated router only updates the FIB entries for the prefixes whose nexthop has been modified after the recomputation of the SPT. This corresponds to the measurements discussed in section 3.6.

4.2 Convergence time

In section 3, we evaluated the convergence time of a router by sending packets through it and measuring the delay between the failure and the transmission of the first packet on a new interface after update of the FIB. This approach is not applicable for a large simulated networks because up to a few hundred of routers must be considered and sending packets is expensive in the simulator. Furthermore, sending packets as used by [20] only samples the routers' FIBs at regular intervals.

To evaluate the convergence time of a network after a failure we use an approach similar to the one proposed by Kerapula et al. in [14]. When there are no failures inside the network, the routing is consistent, i.e. any router is able to reach any other router inside the network. After a link failure, the routers that were using the failed link need to update their FIB. Each router will update its FIB at its own pace, depending on the arrival time of the LSPs and its configuration. While the FIBs are being updated, the routing may not be consistent anymore. To determine the convergence time, we check the consistency of the FIBs of all simulated routers after the update of the FIB of *any* router. To do this, our simulator follows all the equal cost paths that a packet sent by a router S with D as destination could follow. If there is a forwarding loop for any $Source - Destination$ pair or if a router is forwarding packets on a failed link, then convergence is not reached. We defined the *instant of convergence* as the last moment at which the routing becomes and remains consistent. Note that it is possible to find situations where the network converges transiently, then goes back into an inconsistent forwarding state, to finally reach a consistent forwarding state. This is the reason why we say we consider the last transition to a consistent forwarding state.

5. SIMULATION RESULTS

In this section, we used the simulation model described in the previous section to first evaluate whether sub-second convergence after link and router failures is possible in large SP networks. We analyse the impact of the flooding component on the convergence time. We show that the RIB/FIB Update component is the determinant one and explain why fast-flooding is required to quickly converge after a router failure.

We use two representative, but very different SP topologies. The first one, GEANT, is the pan-European Research Network (`http://www.geant.net`). It connects all the National Research networks in Europe and has interconnections with research networks in other continents. GEANT is composed of 22 routers, 21 in Europe and one in New-York, USA. The network topology is highly meshed network with a lot of redundancy in the core (Germany, Switzerland, France, UK, Netherlands) and fewer redundancy in the other parts of the network. Each POP is composed of a single router. It only contains continental links, which means that link delays are generally very low, except links that connect the network to the access router in New York.

The second studied network contains the backbone nodes of a worldwide Tier-1 ISP. The backbone of this network has about 200 routers routers in Europe, America and Asia. It is representative of a large commercial SP network. Each POP is usually composed of two core routers as well as several aggregation and access routers. In each POP, the core routers terminate the high bandwidth inter-POP links and are interconnected with redundant links.

To ease the comparison between the simulation results, we selected the same parameters for each network. Table 1 reports the values of all the relevant parameters. The only differences between the two networks are the SPF computation time that is function of the number of nodes and the number of prefixes advertised by each router, obtained with a LSP trace analysis.

Table 1: Simulation parameters

lsp_process_delay	[2,4]ms
pacing_timer	$\{6, 33, 100\}ms$
fast_flooding	on/off
spf_initial_wait	$\{10, 25, 50, 100\}ms$
spf_exponential_increment	$\{25, 50, 100\}ms$
spf_maximum_wait	$10000ms$
spf_computation_time	[20,30]ms in Tier-1 ISP [2,4]ms in GEANT
rib_fib_prefix_update_delay	[100,110] μs/prefix
rib_fib update type	incremental/full

5.1 IGP convergence after link failures

We begin our simulation study with link failures, the most frequent event that can occur in the topology of a network [15]. For GEANT, we simulated the failures of all links. For the Tier-1 ISP, we simulated the failures of the 50 links that carried the largest number of router to router paths.

When a link fails, the two routers attached to it detect the failure and originate a new LSP. Thanks to the two-way connectivity check [18], a link is considered as having failed as soon as *one* of the two LSPs containing the link has been received by a rerouting router. This implies that the first LSP received after a failure is sufficient to allow any rerouting router to update its FIB.

We first used simulations to check that the sub-second IGP convergence target could be met in the GEANT network. For those simulations, we failed one link at a time. In figure 2, each curve shows the sorted simulated convergence times for the 50 links failures in GEANT. For the simulations, we set the $spf\_initial\_wait$ to $10ms$ or $100ms$ and evaluated the impact of the type of FIB update. The simulations show that the sub-second convergence after link failure is easily met in the GEANT network. This was expected given the delays in the network. A closer look at the four curves in figure 2 shows that a lower $spf\_initial\_wait$ reduces the convergence time. The simulations also shown the benefits of performing an incremental FIB update. This is because when a link fails, only a small portion of the prefixes are reached via the failed link.

Achieving sub-second IGP convergence in Tier-1 SP networks is more challenging given the number of nodes, prefixes and the larger link delays found in a worldwide network.

Figure 3 shows that with all the considered parameters sub-second convergence is achieved. The overall convergence time in the Tier-1 SP is larger than in GEANT. This difference is mainly due to three factors. First, the link delays are larger in the Tier-1 SP. Second, the Tier-1 SP contains more IGP prefixes than GEANT. Third, the larger number of nodes in the Tier-1 SP leads to a longer SPF computation time. As for the simulations with GEANT, using a low $spf\_initial\_wait$ and incremental FIB updates reduces the convergence time. Note that in the Tier-1 SP, the benefit of using incremental FIB updates is much higher than in GEANT. This is because the total number of prefixes in the Tier-1 ISP is ten times larger than the number of prefixes in GEANT.

To evaluate the impact of the topology on the IGP convergence, we performed simulations with several modifications to the topology of the Tier-1 ISP. We used the best simulation settings obtained from figure 3, i.e. $10ms$ $spf\_initial\_wait$ and incremental FIB updates.

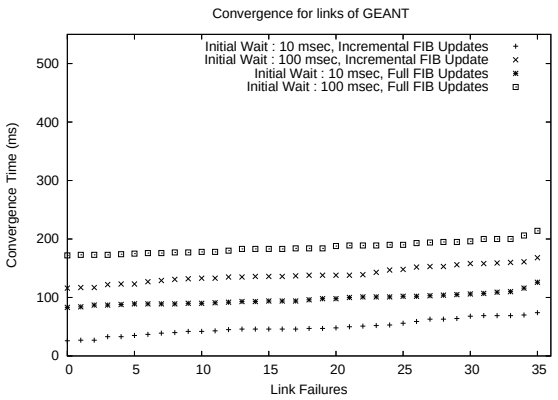

Figure 2: Convergence time for the link failures of GEANT, Initial Wait value set to $10ms$ and $100ms$, Full and Incremental FIB Updates

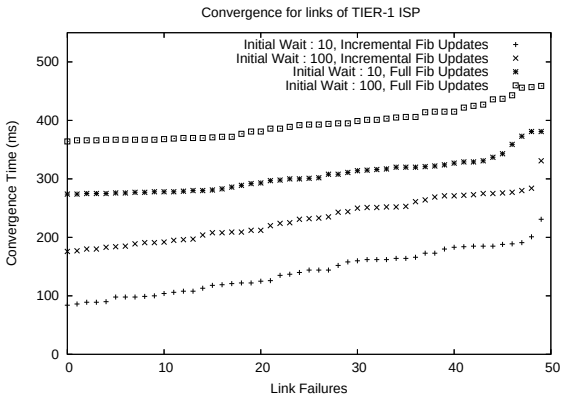

Figure 3: Convergence times for 50 link failures of Tier-1 ISP, Initial Wait value set to $10ms$ and $100ms$, Full and Incremental FIB Updates

First, to evaluate the impact of the link propagation delays on the convergence time, we built a new topology with all link delays set to one millisecond. Figure 4 shows that the IGP convergence times are only slightly reduced with this modification. This is mainly because first the SPF and FIB update times are the key factors in the IGP convergence of the studied network. Second, the IGP weights in this network, as in most SP networks, were set to favour rerouting close to the failure. This implies that rerouting occurs close to the failed link and hence the propagation time of the LSPs is a small component of the overall convergence.

Second, we modified the Tier-1 SP topology and set all link weights to one instead of the weight configured by the operator. The simulations show that this setting increases the IGP convergence time. This is because with such weights the rerouting routers can be farther from the failure than with the IGP weights configured by the network operators. Another consequence of this weight setting is that the FIB of more routers needs to be updated after each failure.

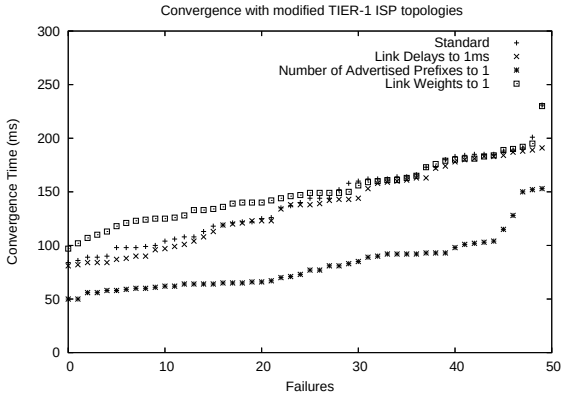

Figure 4: Convergence time for the link failures in the modified Tier-1 ISP, Initial Wait value set to $10ms$, Incremental FIB Updates

We obtained the most significant improvements in the convergence times by reducing the number of prefixes advertised by each router. When each router advertises a single prefix, convergence times are halved for nearly all the considered failures in the Tier-1 ISP. This shows that the number of advertised prefixes is one of the most important components of the convergence time. Similar results were obtained with similar modifications to the GEANT topology.

5.2 IGP convergence after router failures

Besides independent link failures, ISP networks also need to face correlated link and router failures [15]. To model such failures, we consider that all the links attached to a router fail at the same time. There are other types of SRLG failures (e.g. all links using the same optical fibre), but we did not have enough information on the physical structure of the simulated networks to correctly model those failures. For GEANT, we considered the failures of all routers. For the Tier-1 ISP we simulated the failures of the 23 routers that were connected to the 50 most loaded links of the network.

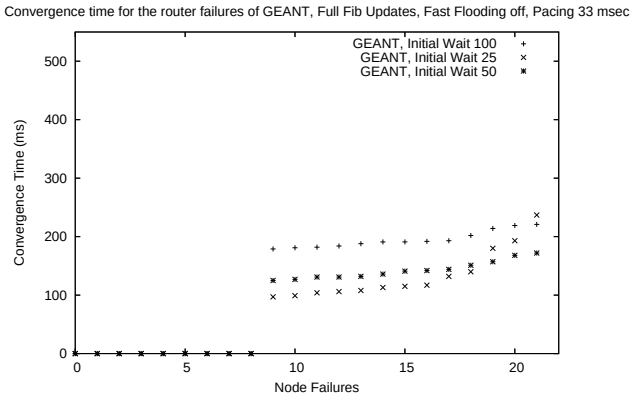

Figure 5: Convergence time for the router failures of GEANT, Full FIB Updates, Fast Flooding off, Pacing $33ms$

The main difference between the failure of a single link and the failure of multiple links is that in the latter case, the first LSP received by a router is not always sufficient to describe the entire failure. In the case of a router failure, all the LSPs of the neighbours of the failed router might be necessary to correctly update the FIB.

To evaluate the convergence time in the case of a router failure, we first consider a configuration that corresponds basically to IS-IS routers that have not been optimised for fast convergence : $33ms$ pacing timer without fast-flooding and full FIB updates.

The simulations performed in GEANT (figure 5) show that this parameter setting allows to achieve sub-second convergence in case of router failures. In GEANT, the worse convergence time after a router failure was less than $250ms$. Surprisingly, the convergence time for some router failures was $0ms$. In fact, according to the IGP weights used by GEANT, those routers act as stub and do not provide any transit. When such a stub router fails, the reachability of the other routers is not affected. A closer look at the simulation results reported in figure 5 shows that the value of the $spf\_initial\_wait$ parameter does not have the same influence as with the link failures. For some router failures, the GEANT network can converge faster with a $100ms$ $spf\_initial\_wait$ than when this parameter is set to $25ms$. The simulation traces revealed that with a $25ms$ $spf\_initial\_wait$ some routers in the network had to update their FIB twice to allow the routing to converge. Those recomputations increase the convergence time.

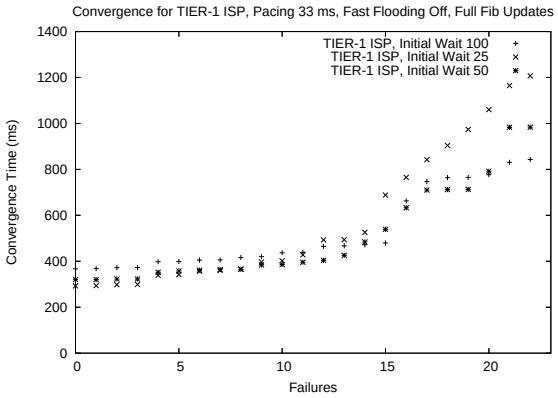

Figure 6: Convergence time for 23 router failures of Tier-1 ISP, Full FIB Updates, Fast Flooding off, Pacing $33ms$

We used the same parameter setting for the Tier-1 SP. Figure 6 shows that, in this case, the sub-second convergence is not achieved for router failures. We can see that for only 60% of the router failures, the convergence time is between 200 and $400ms$. For the other router failures, the convergence time can be as high as $1400ms$. A closer look at the simulation traces revealed the reasons for those large convergence times.

The main problem is that some routers update their FIB before having received all the LSPs of all neighbours of the failed router. Unfortunately, this first update is not sufficient to allow the router to compute a correct FIB and a second, and sometimes third, update of the FIB is necessary. Given the number of prefixes advertised in the Tier-1 SP, those multiple full FIB updates explain around $660ms$ of the total convergence time. The remaining $600ms$ for some router failures are due to a cascading effect. With a single-threaded IS-IS implementation, a router cannot participate in the flooding of LSPs while it is recomputing its SPT or updating its FIB. With the standard pacing timer of $33ms$ and a $spf\_initial\_wait$ of $25ms$, a router can only receive one LSP from each of its direct neighbours before deciding to recompute its SPT. In some cases, correspond-

ing to the left part of figure 6, those early LSPs are sufficient to correctly compute the final FIB and allow the network to converge. However, for the router failures corresponding to the right part of figure 6, the router spends almost $250ms$ to recompute its SPT and update its FIB. During this time, it does not flood LSPs and thus routers downstream do not receive updated LSPs and compute incorrect SPTs and FIBs. We verified this by analysing the traces and by setting the pacing timer to $100ms$. In this case, the convergence time was much larger. When the $spf\_initial\_wait$ is set to $50ms$ or $100ms$, the convergence time is reduced but still rather large.

To solve this problem, we must configure the routers to ensure that the routers only trigger their SPT computation once they have received all the LSPs describing the failure. This is possible by using the fast-flooding mechanism described in section 3.4.

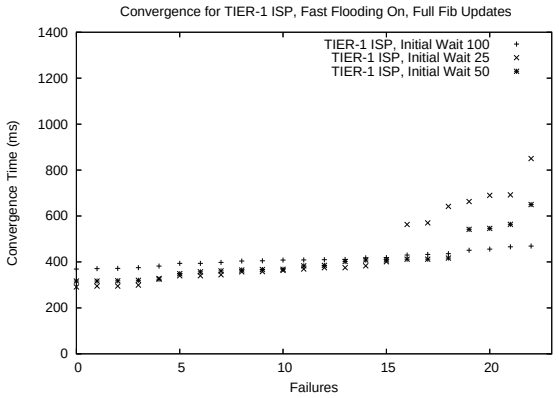

Figure 7: Convergence time for 23 router failures of the Tier-1 ISP, Full FIB Updates, Fast Flooding on

Figure 7 shows that when fast-flooding is used together with the full FIB updates, the sub-second convergence objective is easily met for all considered router failures in the Tier-1 SP. For 60% of the router failures (left part of figure 7), the $spf\_initial\_wait$ only has a limited influence on the convergence time. For the remaining router failures (right part of figure 7), a $spf\_initial\_wait$ of $100ms$ provides the lowest convergence time. With a $25ms$ or $50ms$ $spf\_initial\_wait$, the simulation traces reveal that some routers are forced to perform more than one update of their FIB, leading to a longer convergence time.

Besides the utilisation of fast-flooding, another possible modification to the configuration of the router would be to use incremental FIB updates. For the link failures, the improvement was significant.

Figure 8 summarises the simulations performed with fast-flooding and incremental FIB updates in the Tier-1 SP network. These simulations show that sub-second convergence is conservatively met also for the router failures in this network[3]. As explained earlier, the main benefit of using incremental FIB updates is to reduce the time required to update the FIB in all routers. When a failure affects only 10 prefixes on a given router, the FIB update time is around $1ms$ compared to the $220ms$ full FIB update time. This implies that even if a router triggers its SPT too early, it will block the LSP flooding for a shorter period of time. Furthermore, if a router needs to update its FIB twice, then fewer prefixes will be modified during the second update and this update will be faster.

We also used this simulation scenario to evaluate how the convergence time was affected by the configuration of the exponential

[3]Note that the y scale changed in figure 8 compared to figure 7.

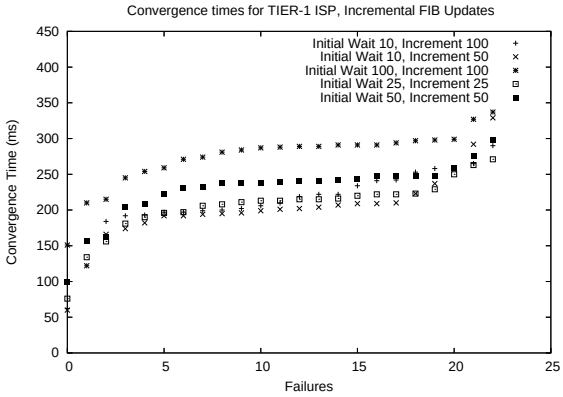

Figure 8: Convergence time for 23 router failures of Tier-1 ISP, Incremental FIB Updates, Fast Flooding on

backoff mechanism associated with the SPT trigger. The simulation results shown in figure 8 reveal that the most important parameter is the $spf\_initial\_wait$. As explained earlier, it should be set to ensure that for most failures, all LSPs have been received by all routers before the computation of the SPT. Our simulations do not indicate an optimal setting for the $spf\_exponential\_increment$. Finally, the setting of the $spf\_maximum\_wait$ depends on the acceptable CPU load on the routers during network instabilities.

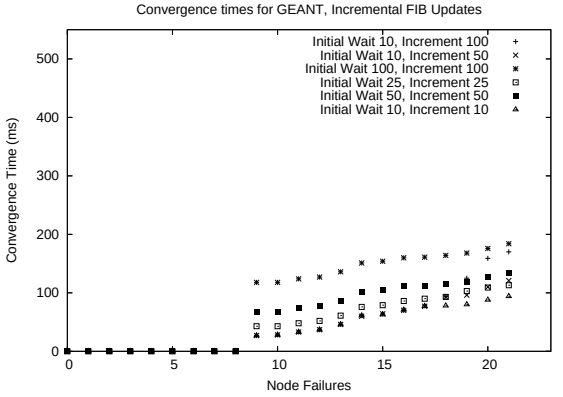

Figure 9: Convergence time for the router failures of GEANT, Incremental FIB Updates, Fast Flooding on

We also performed simulations with fast-flooding and incremental FIB updates in the GEANT network. The simulation results reported in figure 9 show that a low $spf\_initial\_wait$ combined with a low $spf\_exponential\_increment$ provide the best IGP convergence times. A low $spf\_exponential\_increment$ is sufficient in this network given the small number of nodes and prefixes.

Our simulations clearly show that sub-second IGP convergence can be conservatively met in large SP networks with an appropriate tuning of the IGP configuration. First, the pacing timer should not be applied to urgent LSPs. Second, routers must flood urgent LSPs before recomputing their SPT and updating their FIB. Fast-flooding features are thus recommended for fast convergence. Third, the router should need to modify the FIB entries only for the prefixes affected by the failure (incremental FIB Updates), and prefix prioritization should be used to let the most important ones be updated

first. Fourth, using an incremental algorithm to update the SPT would also reduce the convergence time. Finally, in a large network, the configuration of the $spf\_initial\_wait$ on all routers in the network depends on the types of expected failures. If only individual link failures are expected, then the $spf\_initial\_wait$ can be set to a very low value such as $2ms$. If the network must converge quickly after router or SRLG failures, then our simulations show than in the Tier-1 SP network, a $spf\_initial\_wait$ of $50ms$ is appropriate. In operational networks, we would advice a more conservative value such as $150ms$. This value will allow the network to meet the sub-second IGP convergence objective with a sufficient margin to take into account various delays that could occur in the network and that cannot be accurately modelled in a simulator.

6. RELATED WORK

The convergence of IGP protocols has been studied by various authors. Alaettinoglu et al. present in [1] an analysis of the convergence of ISIS and explore changes to the ISIS specification and propose some improvements to routers implementations. Since the publication of [1], the IETF and router manufacturers have worked on improving the convergence of IGP protocols. First, the IETF is currently working on a new protocol : BFD [13] to provide a fast failure detection. Compared to the fast ISIS hello timers proposed in [1], the main advantage of BFD is that it can be implemented directly on the linecards. Second, the router manufacturers have tuned their implementations as explained in section 3. With the implementation of incremental SPF algorithms, the cost of running SPF is not an issue anymore. Our measurements indicate that the main component of the IGP convergence, at the router level, is the FIB update time. Basu and Riecke used simulations to evaluate the convergence time of OSPF in large ISP networks [2]. Their simulations mainly evaluate the impact of using Hello Timers of a few hundred milliseconds on the convergence time and CPU load. With such timers, they obtain a convergence time of around a second. Our measurements indicate that on today's routers, a faster failure detection is possible by relying on the link layer. Finally, Iannacone et al. evaluate in [11] the feasibility of providing faster restoration in large ISP networks. This feasibility was evaluated by using rough estimates of the possible IGP convergence time in a large ISP network. In this paper, we have shown quantitatively that fast IGP convergence is possible by using measurement based simulations.

Shaikh and Greenberg present in [22] a detailed black-box measurement study of the behaviour of OSPF in commercial routers. Compared to this study, our measurements show in details the various factors that affect the performance of ISIS and take into account the multiple improvements to the ISIS implementations since the publication of [22]. In [25], Villfor shows by measurements in a large ISP network that sub-second convergence can be achieved by tuning the ISIS parameters and using some of the techniques described in this paper. Those measurements confirm that sub-second convergence can be achieved while maintaining the stability of the IGP.

7. CONCLUSION

In this paper, we have presented a detailed study of all the factors that affect the convergence of link state IGP protocols in large ISP networks.

We have first presented a detailed measurement study of all the factors that, on a single router, influence the convergence time. This time can be characterised as $D + O + F + SPT + RIB + DD$ where the detection time (D), the LSP origination time (O) and the distribution delay (DD) are small compared to our sub-second objective. The flooding time (F) depends on the network topology and thus on the link propagation delays. The SPT computation time depends on the number of nodes in the network, but can be significantly reduced by using an incremental SPT computation. Finally, the RIB time that corresponds to the update of the RIB and the FIB is the most significant factor as it depends linearly on the number of prefixes affected by the change. Note that by using proritization techniques, it is possible to provide faster convergence for the most important prefixes.

We have then used simulations to evaluate the IGP convergence time in large ISP networks. Our simulations show that, in the case of link failures, a convergence time of a few hundred of milliseconds can be achieved by using a low initial wait timer for the SPF computation and incremental FIB updates. We also show that advertising fewer prefixes in the IGP significantly reduces the convergence time. When considering router or SRLG failures, the convergence time is only slightly larger provided that the pacing timer is disabled for urgent LSPs and that the initial wait timer is not too low.

Overall, our analysis shows that with current router technology sub-second IGP convergence can be provided without any compromise on stability.

Acknowledgements

We would like to thank Nicolas Simar and Thomas Telkamp for their help in obtaining SP topologies.

8. REFERENCES

[1] C. Alaettinoglu, V. Jacobson, and H. Yu. Towards millisecond IGP congergence. Internet draft, draft-alaettinoglu-ISIS-convergence-00.ps, work in progress, November 2000.

[2] A. Basu and J. Riecke. Stability issues in ospf routing. In *SIGCOMM '01*, pages 225–236, New York, NY, USA, 2001. ACM Press.

[3] R. W. Callon. Use of OSI IS-IS for routing in TCP/IP and dual environments. Request for Comments 1195, Internet Engineering Task Force, Dec. 1990.

[4] S. Casner. A fine-grained view of high-performance networking. Presented at NANOG22, May 2001.

[5] Cisco. IP Event Dampening. `http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newf%t/120limit/120s/120s22/s_ipevdp.htm`.

[6] Cisco. IS-IS Fast-Flooding of LSPs Using the fast-flood Command. Technical document, `http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature%_guide09186a00801e87ab.html`.

[7] Cisco. IS-IS Support for Priority-Driven IP Prefix RIB Installation. Technical document,`http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newf%t/120limit/120s/120s26/fslocrib.pdf`.

[8] Cisco. SONET triggers. `http://www.cisco.com/warp/public/127/sonetrig.pdf`.

[9] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[10] C. Filsfils. Fast IGP convergence. Presented at RIPE47, January 2004.

[11] G. Iannaccone, C. Chuah, S. Bhattacharyya, and C. Diot. Feasibility of IP restoration in a tier-1 backbone. *IEEE Network Magazine*, January-February 2004.

[12] D. W. Jacob. SSF Implementation of OSPFv2 v0.2.2. `http://ssfnet.d-jacob.net/`.

[13] D. Katz and D. Ward. Bidirectional forwarding detection. Internet draft, draft-ietf-bfd-base-02.txt, work in progress, March 2005.

[14] R. Keralapura, C. N. Chuah, G. Iannaccone, and S. Bhattacharyya. Service Availability: A New Approach to Characterize IP Backbone Topologies. In *Proceedings of IEEE IWQoS*, 2004.

[15] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C.-N. Chuah, and C. Diot. Characterization of failures in an IP backbone. In *IEEE Infocom2004*, Hong Kong, March 2004.

[16] A. Martey. *IS-IS Network Design Solutions*. Cisco Press, 2002.

[17] J. M. McQuillan, I. Richer, and E. C. Rosen. An overview of the new routing algorithm for the arpanet. In *Proceedings of the sixth symposium on Data communications*, pages 63–68. ACM Press, 1979.

[18] D. Oran. OSI IS-IS intra-domain routing protocol. Request for Comments 1142, Internet Engineering Task Force, Feb. 1990.

[19] J. Parker, D. McPherson, and C. Alaettinoglu. Short Adjacency Hold Times in IS-IS. Internet draft, draft-parker-short-isis-hold-times-01.txt,work in progress, July 2001.

[20] D. Pei, L.Wang, D. Massey, S. F. Wu, and L. Zhang. A study of packet delivery during performance during routing convergence. In *Proceedings of IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, June 2003.

[21] Renesys. SSFNet, Scalable Simulation Framework for Network Models. `http://www.ssfnet.org/`.

[22] A. Shaikh and A. Greenberg. Experience in black-box ospf measurement. In *Proceedings of the First ACM SIGCOMM Workshop on Internet Measurement*, pages 113–125. ACM Press, 2001.

[23] A. Shaikh, A. Varma, L. Kalampoukas, and R. Dube. Routing stability in congested networks: experimentation and analysis. *SIGCOMM Comput. Commun. Rev.*, 30(4):163–174, 2000.

[24] J.-P. Vasseur, M. Pickavet, and P. Demeester. *Network Recovery: Protection and Restoration of Optical, SONET-SDH, and MPLS*. Morgan Kaufmann, 2004.

[25] H. Villfor. Operator experience from isis convergence tuning. Presented at RIPE47, January 2004.

[26] R. White and A. Retana. *IS-IS: Deployment in IP networks*. Addison-Wesley, 2003.