Network Working Group
Internet-Draft
Expires: August 28, 2008

I. Hokelek
M. Fecko
P. Gurung
S. Samtani
S. Cevher
J. Sucec
Applied Research
Telcordia Technologies, Inc.
February 25, 2008

Loop-Free IP Fast Reroute Using Local and Remote LFAPs
draft-hokelek-rlfap-01.txt

Status of this Memo

Copyright Notice

Abstract

This draft describes a new loop-free IP fast reroute mechanism which enhances the IP Fast-ReRoute (IPFRR) [1,15] by introducing the concept of pre-computed remote Loop-Free Alternate Paths (rLFAPs) on top of the IPFRR local LFAP. In rLFAP, a router which is adjacent to the failed resource switches over to pre-computed LFAPs, if they

exist, immediately after failure detection. Multi-hop neighbors (MNBs) are notified about this remote failure as quickly as possible using fast failure notification mechanism. Upon receipt of failure information, MNBs activate their pre-computed remote LFAPs that they maintain for protecting against remote failures within their multi-hop neighborhoods. In the worst case, where IPFRR results in forming micro-loops, rLFAP completely prevents micro-loops for single link failures and quickly converges to a loop-free path in case of multiple link failures.

1. Introduction

Fast reroute in communication networks is defined as the dynamic and timely redirection of a primary path to an alternate secondary path in response to degradation/failure of the primary path. IP Fast-ReRoute (IPFRR) drafts [1,15] provide a framework for the fast-reroute mechanism which minimizes the adverse effects of link or node failures by timely invoking the pre-computed repair paths.

IPFRR performs well when a router which is adjacent to the failed resource (link or router) has Loop-Free Alternate Paths (LFAPs). However, major issues arise either when there is no local LFAP or the router assumes that the alternate path is loop-free but this might not be true if there are inconsistencies in the routers' FIBs (e.g., due to failure propagation and FIB update delays). In the latter case, a micro-loop might be formed when the primary path is replaced with the alternate path.

A relative performance of a fast-reroute mechanism should not be worse than the performance with relying on the routing protocol's (e.g., OSPF or IS-IS) standard repair mechanism. The performance metrics include repair delay, additional overhead, ability to handle micro-loops, backward compatibility, complexity (processing and memory), and the repair path coverage and quality. An ideal but non-realistic fast-reroute mechanism would have zero repair delay, create no extra overhead, either be micro-loop free or handle all micro-loops, be backward compatible, require minimum amount of processing power and memory, and cover all failure scenarios.

Fast reroute can be severely impaired by micro-loops which represent the worst case scenario for IPFRR. Ref. [2] summarizes the techniques proposed for minimizing the adverse effects of micro-loops. These techniques either can resolve the micro-loops only partially [3], or can suffer high delays [4, "incremental cost advertisement"], or are highly complex [5,6]. Therefore, a fast reroute mechanism that is micro-loop free and has comparable performance results to IPFRR in terms of the above metrics will significantly help IETF to reach its goal of dynamic and timely rerouting.

This document describes a new micro-loop free fast reroute mechanism

which enhances IPFRR [1,15] by introducing the concept of pre-computed remote LFAPs (rLFAPs) on top of the IPFRR local LFAP. This mechanism achieves complete and fast micro-loop prevention at the minimal amount of extra complexity. In rLFAP, a router that is adjacent to the failed resource immediately switches over pre-computed LFAPs if LFAPs for protecting against this local failure exist (the same as IPFRR) and instantly propagates failure information to multi-hop neighbors (MNBs) (e.g., X-hop neighbors, where X is an integer number representing how many hops away from the failure). Upon receipt of failure information, MNBs activate their pre-computed LFAPs that they maintain for protecting against remote failures within their MNBs. Note that this draft uses the existing well-defined LFAP criteria in [15] but extends its applicability by calculating remote LFAPs to protect against remote failures.

2. Overview of rLFAP

The number of LFAP choices to bypass a failed resource is either equal or higher at routers which are multi-hop away from the failed component compared to the router adjacent to the failure (the latter is the subset of the first). This fact together with the concept of remote LFAP, which protects against failures at multi-hop routers, is the key motivation behind rLFAP.

One might think that rLFAP would not be fast enough since there is a certain failure notification delay before activating remote LFAPs. However, in rLFAP, each router pre-computes two sets of LFAPs: the first set includes local LFAPs which are activated instantly (if exist) when a local failure is detected and the second set includes remote LFAPs which are activated when notifications of remote failures arrive from multi-hop routers. Therefore, rLFAP follows the approach proposed in the IPFRR draft [1] for its local best-effort fast reroute and quickly continues to correct the previous best effort decisions as failure notifications keep on arriving from MNBs. Since rLFAP gradually corrects the previous decisions, micro loops are detected and corrected before or shortly after they are formed (rLFAP completely prevents micro-loops for single link failures and quickly converges to a loop-free path in case of multiple link failures).

The main features of rLFAP are as follows:
- Prevention of micro-loops without tunneling
- Handling multiple link/node failures
- Faster than IPFRR in the worst case scenarios; comparable otherwise
- Ability to distinguish link and node failures
- Scalability due to limiting the scope of MNBs to X hops
- Minimal additional overhead for fast failure notification

2.1 Micro-loop Prevention Using Remote LFAPs

```
S----------------A------------------B
|                                   |
|                                   |
|                                   |
|                                   |
E-----------------------------------D
|                                   |
|                                   |
|                                   |
|                                   |
F-----------G-----------H-----------J
```

Figure 1: Micro-loop prevention with rLFAP (link E-D fails)

An example link failure scenario is shown in Figure 1, where link metrics are unity (i.e., hop-count). The primary path from S to D is S-E-D before link E-D fails. After E detects the failure, IPFRR switches instantly over the shortest alternate path E-S-A-B-D. In this case, a micro-loop is formed between S and E and will cause network disruption until a new primary path S-A-B-D converges if none of the techniques in Ref. [2] is employed. However, rLFAP immediately starts using LFAP S-A-B-D when S is notified about the failure of link E-D. Note that LFAP S-A-B-D is pre-computed at S to protect against the failure of link E-D (this is a remote link failure from router S point of view and the path S-A-B-D is a remote LFAP from link E-D point of view).

2.2 Handling Multiple Simultaneous Failures

```
S-----------A-----------B-----------C
|                                   |
|                                   |
|                                   |
|                                   |
E-----------------------F-----------D
|                       |
|                       |
|                       |
|                       |
G-----------------------H
```

Figure 2: A simple topology with unit link costs demonstrating rLFAP's capability for handling multiple simultaneous failures (links E-F and H-F fail at the same time)

Figure 2 shows a scenario where two link failures (links E-F and H-F) occur. Using IPFRR, E switches over the shortest alternate path E-G-H-F-D as soon as it detects the failure of link E-F. However, after the switchover, H sends all packets coming from the source E back to G since link H-F also failed. The micro-loop

between E and H should be resolved. After resolving the micro-loop between E and H, another micro-loop between E and S should also be resolved to enable successful reroute. If S is notified about these two link failures, then the best LFAP S-A-B-C-D can be utilized quickly (X=3 is needed in this scenario). The number of LFAPs needed for the multiple link failures will not be scalable if they are required to be pre-computed by each router for any combination of failures within the entire network. However, rLFAP only needs to protect failures within its MNBs (explained in Section 3.1); therefore rLFAP significantly enhances the IPFRR scalability in resolving micro-loops in case of multiple failures (e.g., compared to the NOT-VIA mechanism [6]).

2.3 Distinguishing Link and Node Failures

```
      A------------B-------------C
      |            |             |
      |            |             |
      |            |             |
      |            |             |
      S------------E-------------D
      \           /
       \         /
        \       /
         \     /
          \   /
           \ /
            F
```

Figure 3: rLFAP's ability to distinguish link and node failures (either link S-E or router E fail)

Another important rLFAP feature is the ability to distinguish between link and node failures. Figure 3 shows an example topology with unit link costs for demonstrating the rLFAP's ability to distinguish link and node failures. The primary path from S to D is S-E-D before any failure. Initially, a router assumes that it is a link failure whenever its communication to a neighboring node is disrupted (e.g., S detects that it can't reach node E through link S-E). In this case, the pre-computed LFAP S-F-E-D will be activated immediately without waiting for distinguishing whether it is a link or node failure. However, if it is a router failure (e.g., node E fails), the failure notification concept of rLFAP makes it possible for routers to receive the failure notification from other interfaces of the failed router (e.g., node S receives failure notifications of links B-E and F-E using 2-hop (i.e., X=2) failure notification mechanism). Upon detection of router's failure, the pre-computed LFAP S-A-B-C-D is activated. Hence, in rLFAP, routers can distinguish between link and node failures. The significance of this feature is supported by the recent work by Gjoka et. al. [7], which shows that the failure coverage of the fast reroute mechanisms

is different for link and node failure cases.

3. Design Details

rLFAP achieves loop-free convergence by introducing two additional mechanisms on top of IPFRR: multi-hop failure notification and remote alternate paths (remote LFAPs or SAPs) for protecting against failures at multi-hop routers. Apart from these mechanisms, rLFAP implements all its functionalities using the IPFRR framework [1]. In this section, we first describe multi-hop neighborhoods (MNBHs) which limit the number of the required remote LFAPs/SAPs for scalability. And then, two fast failure notification mechanisms and LFAP calculations for local and remote failures within MNBH are presented. It is crucial for all fast reroute mechanisms that the underlying dynamic routing protocol should converge back to its optimum routes without causing micro-loops once the topology change is disseminated to all routers in the network. rLFAP provides a fast re-convergence from LFAPs to the optimum new routes by utilizing new routes shortly after they are calculated.

3.1 Multi-hop Neighborhood (MNBH)

```
N11----N12----N13----N14----N15----N16----N17
 |      |      |      |      |      |      |
 |      |      |      |      |      |      |
N21----N22----N23----N24----N25----N26----N27
 |      |      |      |      |      |      |
 |      |      |      |      |      |      |
N31----N32----N33----N34----N35----N36----N37
 |      |      |      |      |      |      |
 |      |      |      |      |      |      |
N41----N42----N43----N44----N45----N46----N47
 |      |      |      |      |      |      |
 |      |      |      |      |      |      |
N51----N52----N53----N54----N55----N56----N57
 |      |      |      |      |      |      |
 |      |      |      |      |      |      |
N61----N62----N63----N64----N65----N66----N67
 |      |      |      |      |      |      |
 |      |      |      |      |      |      |
N71----N72----N73----N74----N75----N76----N77
```

Figure 4: An example network

Figure 4 shows an example network consists of 49 nodes. A node on the boundary has two neighbors if it is located on one of four corner positions (e.g., N11); otherwise three neighbors (e.g., N12). A non-boundary node has four neighbors (e.g., N22) irrespective of its location. The issue is to decide the structure of two adjacent MNBHs: overlapping or non-overlapping. Inconsistencies or micro-loops may arise on boundaries of adjacent MNBHs if they are

non-overlapping since each MNBH has only a partial view of the global topology (e.g., failures close to the boundary of adjacent MNBHs). Also, an additional mechanism is needed to explicitly maintain the boundaries if multi-hop NHs are non-overlapping. Therefore, rLFAP uses overlapping MNBHs.

```
                         |
                         |
                    ----N24----
                  |      |      |
                  |      |      |
             ----N33----N34----N35----
           |      |      |      |      |
           |      |      |      |      |
      ----N42----N43----N44----N45----N46----
           |      |      |      |      |
           |      |      |      |      |
             ----N53----N54----N55----
                  |      |      |
                  |      |      |
                    ----N64----
                         |
                         |
```

Figure 5: 2-hop MNBH for N44

The MNBH for each node only defines a local scope within which to propagate failure notifications. For example, 2-hop (i.e., X-hop where X=2) MNBH of N11 consists of nodes together with their all links which are at most 2-hop away from N11. These nodes include N12, N13, N21, N22, and N31; and hence, there are 5 nodes and 11 links within 2-hop MNBH of N11. However, N44's 2-hop MNBH includes 12 nodes and 36 links as shown in Figure 5. N44 has to calculate separate LFAPs/SAPs for each destination in the network to protect against link/node failures within its MNBH. Since MNBHs are overlapping and define only a local scope for each node, no additional mechanism is needed to explicitly maintain the MNBHs in the network (e.g., a simple flooding mechanism similar to OSPF LSAs but limited to X-hop away routers is sufficient for maintaining MNBHs). Each node takes its best fast reroute action independently in case of a failure within its MNBH. Minimal inconsistencies (hence micro-loops) are expected as a result of each node's independent decision since two overlapping MNBHs' partial topologies have a lot common information.

3.2 Alternate Path Calculation

We describe the alternate path calculation methodology for the node N44 in Figure 5. Other nodes in the network repeat the same steps but using their own MNBHs. For simplicity, we assume that only a single link failure occurs.

N44 has four local links. For each local link LL_k (k=1,2,3,4), the following calculations are done per each destination Nij (i=1,2,3,5,6,7 and j=1,2,3,5,6,7):

- Remove LL_k from the topology to anticipate the failure
- Calculate the shortest path to Nij using the new topology. If the new shortest path to Nij (after the failure) is the same as the primary path to Nij (before the failure) then break (do not perform the following three steps). Do not store any alternate path to Nij for the failure of LL_k. If the primary path to Nij before the failure has equal cost multipaths (ECMPs) and at least one of these ECMPs is affected from the failure, then the following three steps should also be performed.
- Calculate three local shortest alternate paths (SAPs) via immediate neighbors to Nij (there are three immediate neighbors after the link failure and therefore three SAPs need to be calculated). Here SAP is defined as the shortest distance path to Nij via an immediate neighbor calculated by the Djikstra algorithm after removing LL_k from the topology
- Store the shortest local LFAP among these three local SAPs (if LFAP exists) and modify the entry, corresponding to this source-destination pair, in the path safety matrix. This matrix keeps track of loop-free source-destination pairs which will be used during the remote LFAP calculation

A similar method will be used for calculating remote alternate paths. N44 has 32 remote links within its MNBH. For each remote link RL_k (k=5,6,...,32), the following calculations are done per each destination Nij (i=1,2,3,5,6,7 and j=1,2,3,5,6,7):

- Remove RL_k from the topology to anticipate the failure
- Calculate the shortest path to Nij using the new topology. If the shortest path to Nij is the same as the primary path to Nij then break (do not perform the following three steps). Do not store any alternate path to Nij for the failure of RL_k
- Calculate four remote shortest alternate paths (SAPs) via immediate neighbors to Nij (since RL_k is not a local interface there are four immediate neighbors for this node).
- Store the shortest remote LFAP among these four remote SAPs and update the path safety matrix if any LFAP exists. If there is no LFAP, then check the path safety matrix if there is any neighbor which has a loop-free path to this destination indicated by the path safety matrix. If there is/are such neighbor(s), then use the shortest one as remote LFAP and update the corresponding entry in the path safety matrix.

The above algorithms make sure that a local or remote alternate path will be stored only if the primary path does not function anymore after the failure. Therefore, rLFAP stores only the required alternate paths and is scalable. Note that routers only store alternate next-hops (not the alternate path itself).

3.3 Fast Failure Notification Mechanism

The objectives of a multi-hop failure notification mechanism are as follows:

- Routers should be notified about failures as fast as possible to minimize the reroute delay from the primary path to an LFAP
- Failure notification should create minimal overhead in terms of bandwidth consumption. For example, generating too many LSA packets in OSPF consumes the available bandwidth and may cause disruption in other parts of the network beyond the failure point
- A solution which does not require modifications to the underlying routing protocol is preferable
- A failure notification should not cause the network instability under some worst-case scenarios (e.g., sending too many OSPF LSA messages for transient failures (link flapping) may cause the network instability)

There are two options for the multi-hop failure notification in rLFAP when a router detects a local failure:

1) Configuring routing protocol's parameters for the fast failure propagation by relying on periodic link state updates (e.g., LSAs for OSPF and LSPs for IS-IS)
2) Implementing a new efficient fast failure notification mechanism within the MNBH

3.3.1 Configuring Routing Protocol's Parameters

rLFAP is proposed initially for link state IGPs, where link state update packets are LSAs in OSPF and LSPs in IS-IS. For simplicity, we describe the concept for OSPF; all the procedures are applicable to IS-IS as well. This option does not introduce a new signaling mechanism but optimizes the existing link state update mechanisms for rLFAP's performance efficiency.

The flooding procedure by which OSPF distributes LSAs is reliable. A router packages its new LSA within a link state update packet (may contain multiple distinct LSAs) and transmits it on each of its interfaces which are in the same OSPF area impacted by the LSA. Each recipient acknowledges the router from which the LSA was received, repackages the LSA within a new link state update packet and sends it out on each of its interfaces except for the interface on which the LSA was received.

When the content of an LSA changes, a new LSA is originated [8]. However, two instances of the same LSA may not be originated within the time period MinLSInterval. This may require that the generation of the next instance may be delayed by up to inLSInterval. Although MinLSInterval is an architectural constant (default is 5 secs), implementations could make this interval configurable in order to speed up the failure propagation [12].

Ref. [9] studies how to achieve sub-second IGP convergence in large IP networks by configuring the routing protocol parameters. Ref. [10] shows that minLSInterval around 20-30 ms does not generate much overhead while providing fast failure propagation to multi-hop routers. Based on these studies, we suggest that minLSInterval which is around 20-30 ms will provide rLFAP with fast failure notification mechanism due to the MNBH, where the maximum number of hops between two nodes in which one node includes another within its MNBH is limited to X hops. Note also that this interval limits the successive generations of the same LSA. The maximum delay (i.e., 20-30 ms) is realized only if there are two successive topology changes in which the second failure occurs just after an LSA for the first failure is generated.

3.3.2 An Efficient Failure Flooding Mechanism within MNBH

For the stable wired backbone networks, configuring routing protocols' parameters for fast failure notification will neither create much additional signaling overhead nor network instability since transient failures (i.e., link flapping) are rarely occured in these networks. However, for wireless mobile ad-hoc or backbone networks, the drawback of configuring the routing protocol's parameters for fast failure propagation is that the routing protocol overhead will be enormous in the case where frequent transient failures (e.g., link flapping) occur. This overhead is due to too many LSA updates which are generated for each transient topology change and flooded to the entire network. The frequent transient failures may also cause the network instability since the routing protocol may repeat shortest path calculations and FIB updates too frequently for each topology change without allowing a common convergence.

The LSA flooding scope is more explicit in OSPF IPv6 and appears in the LS type field [13]. There are three separate flooding scopes for LSAs:

- Link-local scope: LSA is flooded only on the local link and no further.
- Area scope: LSA is flooded only throughout a single OSPF area.
- AS scope: LSA is flooded throughout the routing domain.

However, rLFAP needs LSA's scope to be configurable to its MNBH and none of these scopes satisfies this requirement. Moreover, an LSA is needed by the routing protocol and should be at least flooded within the same OSPF area.

Therefore, another option for the fast failure notification mechanism in rLFAP is to implement a new flooding procedure within MNBH which minimizes routing protocol instability and overhead. The new flooding mechanism defines a new link update packet (LUP) which is similar to LSA in OSPF but includes two new fields: Time-To-Live (TTL) and

Stop-Flooding (SF) bit. TTL indicates the maximum number of hops a new LUP will be transmitted. The transmission is stopped if TTL field is zero and SF is one (i.e., true). SF decides whether the flooding of LUP will continue beyond the MNBH. A router continues flooding an LUP within the MNBH irrespective of its SF value. However, if TTL is zero, then a router continues the flooding procedure only if the LUP's SF value is zero (i.e., false). An intermediate router changes SF value to one if LFAPs, which protect against this particular failure described in LUP, are found for all destinations in the network. An intermediate router is not allowed to change SF value from one to zero because the SF value 1 indicates that LFAPs are found for all destinations (i.e., full coverage).

LUP has its own timers for controlling its new packet generation and flooding mechanism. This is another reason for introducing a new packet format since the timers for controlling LSA flooding can not be set independently for each LSA. The rLFAP flooding procedure will be the same as OSPF's flooding procedure but with the following modifications:

- Parameters (e.g., timers) for the LUP flooding are set to aggressive values for fast failure notifications while parameters for LSAs are set to relatively conservative for minimizing the routing protocol instability and overhead.
- Each router sets TTL field to the number of maximum hops defined by the MNBH (i.e., parameter X in X-hop MNBH) and SF bit to 0 upon generation of a new LUP packet.
- Each recipient of a LUP packet decrements TTL and sets SF bit to 1 if all required LFAPs are found and then transmits it on each of its interfaces except for the interface on which the LUP was received only if TTL field is nonzero.
- Each recipient which observes zero TTL field continue the flooding procedure if SF bit is still zero; otherwise stops the flooding procedure.

There are two advantages in this method: i) minimal routing overhead since flooding LUP update messages are limited to MNBH, ii) fast failure notification since the parameters of the flooding procedure (e.g., timers) can be set independently from the routing protocol's flooding mechanism's parameters. This efficient flooding mechanism is expected to substantially decrease the amount of additional overhead and routing protocol instability, which are observed during the transient failures, while fast failure notifications are still provided.

3.4 Remote LFAPs and Scalability

In rLFAP, remote LFAPs are calculated only to protect against node and link failures within the MNBH. Therefore, the number of LFAPs for a single failure scenario will not be an important issue in terms of scalability.

However, in case of protecting against multiple failures, the number of pre-computed LFAPs to protect against some combinations of these failures might not be scalable in rLFAP depending on the depth of the MNBH (i.e., parameter X) and the average number of links for each node. Our design utilizes the concept of KEYLINKS introduced in Ref. [14] for the purpose of handling this scalability problem. The main idea behind KEYLINKS is that LFAP for a certain destination is needed only if the primary path fails. Therefore, each node maintains what nodes/links in its MNBH are used for reaching a certain destination and calculates an LFAP only for the links on its primary path (i.e., LFAP, which is needed only if at least one of the links on the primary path fails, should not use any of multiple failures). Maintaining key links and nodes adds additional complexity while the gain is obtained in terms of scalability (i.e., fewer LFAPs need to be pre-computed and stored). For example, N44's 2-hop MNBH includes 12 nodes and 36 links as shown in Figure 5. For a certain destination, assume that 2 nodes and 3 links from the 2-hop MNBH of N44 are on the primary path. Therefore, LFAPs are needed only to protect against 5 members (5 LFAPs) instead of 48 members (48 LFAPs) of the 2-hop MNBH for a single failure scenario. This reduction for a single destination is significant since each node needs to pre-compute LFAPs for all possible destinations in the network.

3.5 Routing Convergence from LFAPs to New Primary Paths

It is crucial for all fast reroute mechanisms that the underlying dynamic routing protocol should converge to its new optimum routes once the topology change is disseminated to all routers in the network. The loops can still form in this phase if the FIB updates are not done in the right order.

In rLFAP, all nodes within the MNBH have pre-computed alternate paths protecting against the failures within the MNBH. This makes sure that all nodes within the MNBH are aware of these failures and therefore their current routes do not include the failed resources. This feature provides a flexibility of timely switching back to new primary paths when new paths are calculated by the routing protocol. rLFAP switches back to new primary paths after waiting for a constant delay representing the rLFAP convergence time within the MNBH (this delay starts after new paths are found by the routing protocol).

Another important feature of rLFAP is that the minimum number of next hop changes is expected during switching from LFAPs to new primary paths of the dynamic routing protocol. The reason is that the next hop change will occur only if the next hop in the new primary path is different than the next hop in the alternate path. Due to the MNBH concept, it is conjectured that the next hops for LFAPs and the new primary paths will be the same with a high probability and the minimum number of FIB updates is expected

during switching from LFAPs to new primary paths. Our initial analysis shows that

3.6 Applicability of rLFAP Mechanism to Support Fast PIM-SM Tree Repair

Protocol Independent Multicast - Sparse Mode (PIM-SM) [16] is a widely available multicast protocol that achieves efficient distribution of multicast content through on-demand construction of shared trees and shortest path trees. Key to its efficiency is its use of Join messages to build the multicast tree from the multicast group members towards the Rendezvous Point (RP) or the content source for the cases of shared trees and shortest path trees (SPTs), respectively.

Unfortunately, PIM-SM reacts to link failure events even more slowly than the underlying routing protocol (e.g., OSPF, IS-IS, etc.). Specifically, not only must the underlying unicast routing protocol converge to reflect the degraded network topology, but the appropriate PIM Join messages must be sent, received, and processed before multicast tree repair is completed. Furthermore, depending on the router implementation, PIM-SM may not even recognize a change in the underlying unicast paths for several seconds when time-based events are used to trigger the PIM-SM process to compare the current routing information for changes that impact the reverse path forwarding (RPF) to RPs and multicast sources.

The rLFAP features can help speed up PIM-SM tree repair by accelerating the convergence to correct RPF information about a failure at affected nodes in the LUP TTL MNBH. To illustrate potential benefits of the rLFAP mechanism in the context of PIM-SM, the topology of Figure 2 is considered again. Here, however, Node S is supposed to denote the source node for packets to be delivered by an SPT to the multicast group Z comprising Nodes D and G. The SPT constructed by PIM-SM (assuming unit cost links) is shown in Figure 6.

```
S           A           B           C
|
|
|
|
E-----------------------F-----------D
|
|
|
|
G                       H
```

Figure 6: SPT constructed by PIM-SM for multicast group Z = (D,G)

with source node S based on the topology of Figure 2 (links not part of the SPT are omitted from the figure).

Now, per the scenario of Figure 2, it is assumed that links E-F and F-H of Figure 2 fail at the same time. However, when the LUP originated by Node F arrives at Node D, D will recognize that the RPF to S has changed and D will use the pre-computed LFAP to S via the link D-C as the link for the new RPF to S. The processing described thus far is native to the rLFAP mechanisms already presented herein. In order to complete the speed-up of PIM-SM tree repair, routers employing rLFAP mechanisms MUST trigger the PIM-SM process to check the impact on RPF information of those multicast groups for which PIM-SM state is maintained locally whenever a LUP is received and processed. In doing so, the PIM-SM process will learn of the new correct RPF information for affected multicast groups within milliseconds of the received LUP being processed. The PIM-SM implementation MUST then immediately issue the appropriate (*,G) and/or (S,G) Join messages. For the scenario of Figure 6, Node D will issue a (S,G) Join message to its PIM neighbor Node C via the link D-C which, in turn, sends a Join message to Node B, and so forth. The resulting repaired PIM-SM SPT for Group Z is shown in Figure 7.

```
S-----------A-----------B-----------C
|                                   |
|                                   |
|                                   |
|                                   |
E                       F           D
|
|
|
|
G                       H
```

Figure 7: Reconstructed by PIM-SM for multicast group Z = (D,G) following fast tree repair based on LUPs originated by Node F (links not part of the SPT are omitted from the figure).

A few additional points regarding the illustrative scenario of Figures 6 and 7 are worth noting. First, the routing instance at Node F does not originate a new Join message for Group Z as it is no longer "upstream" to Group Z members. That is, a router applying the rLFAP mechanism to speed up PIM-SM tree repair SHOULD suppress sending PIM Join messages if the neighbors for which it maintained Join state are no longer "downstream" from it with respect to the source. Second, the illustrative scenario given here applies without loss of generality to RP-based (i.e., shared) trees: The only difference being (in the example here) that Node D would issue a (*,G) Join message to repair its branch of the multicast tree. Last,

the SPT branch going to multicast group member G was not affected by the link failures and, therefore, LUPs originated by Nodes E and H received at Node G (correctly) did not initiate tree repair.

4. Experimental Results

4.1 LFAP Coverage Analysis

We performed the coverage analysis of the fast reroute mechanism presented here on realistic topologies, which were generated by the BRITE topology generator in bottom-up mode [17]. The LFAP coverage percentage is defined here as the percentage of the number of LFAPs for protecting the primary paths which are failed because of link failures to the number of all failed primary paths. Only local LFAPs are considered in the coverage calculation for the neighborhood depth of 0 (i.e., X=0) while both local and remote LFAPs are taken into account when the neighborhood depth is set to a value greater than 0 (i.e., X>0).

The realistic topologies include AT&T and DFN using pre-determined BRITE parameter values from Ref.[17] and various random topologies with different number of nodes and varying network connectivity. For example, the number of nodes for AT&T and DFN are 154 and 30, respectively, while the number of nodes for other random topologies is varied from 20 to 100. The BRITE parameters which are used in our topology generation process are summarized in Figure 10 (see Ref.[17] for the details of each parameter). In summary, m represents the average number of edges per node and is set to either 2 or 3. A uniform bandwidth distribution in the range 100-1024 Mbps is selected and the link cost is obtained deterministically from the link bandwidth (i.e., inversely proportional to the link bandwidth as used by many vendors). Since the values for p(add) and beta determine the number of edges in the generated topologies, their values are varied to obtain network topologies with varying connectivity (e.g., sparse and dense).

	Bottom up
Grouping Model	Random pick
Model	GLP
Node Placement	Random
Growth Type	Incremental
Preferential Connectivity	On
BW Distribution	Uniform
Minimum BW	100
Maximum BW	1024
m	2-3
Number of Nodes (N)	20,30,50,100,154
p(add)	0.01,0.05,0.10,0.42
beta	0.01,0.05,0.15,0.62

```
|----------------------------|-----------------------|
```

Figure 10: BRITE topology generator parameters

The coverage percentage of our fast reroute method is reported for different network topologies (e.g., different number of nodes and varying network connectivity) using neighborhood depths of 0, 1, and 2. (i.e., X=0, 1, and 2). For a particular failure, LFAPs protecting the failed primary paths are calculated only by those nodes which are within the multi-hop neighborhood of this failure. Note that these nodes are determined by the parameter X as follows: For X=0, two nodes which are directly connected to the failed link, for X=1, two nodes which are directly connected to the failed link and also neighboring nodes which are adjacent to one of the outgoing links of these two nodes, and so on.

The LFAP coverage percentage for a certain topology is computed by the following formula: LFAP Coverage Percentage = N_lfaps*100/N_fpp where N_lfaps is the number of source-destination pairs whose primary paths are failed because of link failures and have LFAPs for protecting these failed paths, and N_fpp is the number of source-destination pairs whose primary paths are failed because of link failures. The source-destination pairs, in which source and destination nodes do not have any physical connectivity after a failure, are excluded from N_fpp. Note that the coverage percentage includes a network-wide result which is calculated by averaging all coverage results obtained by individually failing all edges for a certain network topology.

Figure 11 shows the LFAP coverage percentage results for random topologies with different number of nodes (N) and network connectivity, and Figure 12 shows these results for AT&T and DFN topologies. In these figures, E_mean represents the average number of edges per node for a certain topology. Note that the average number of edges per node is determined by the parameters m, p(add), and beta. We observed that E_mean increases when p(add) and beta values increase. For each topology, LFAP coverage analysis is repeated for 10 topologies generated randomly by using the same BRITE parameters. E_mean and LFAP coverage percentage are obtained by averaging the results of these ten experiments.

Case	N	E_mean	X=0	X=1	X=2
p(add)=0.01	20	3.64	82.39	98.85	100.0
beta=0.01	50	3.86	82.10	98.69	100.0
	100	3.98	83.21	98.04	100.0
p(add)=0.05	20	3.70	85.60	99.14	100.0

beta=0.05	50	4.01	84.17	99.09	100.0
	100	4.08	83.35	98.01	100.0
p(add)=0.1	20	5.52	93.24	100.0	100.0
beta=0.15	50	6.21	91.46	99.87	100.0
	100	6.39	91.17	99.86	100.0

Figure 11: Coverage percentage results for random topologies

Case	N	E_mean	X=0	X=1	X=2
p(add)=0.42	154 (AT&T)	6.88	91.04	99.81	100.0
beta=0.62	30 (DFN)	8.32	93.76	100.0	100.0

Figure 12: Coverage percentage results for AT&T and DFN topologies

There are two main observations from these results:

1. As the neighborhood depth (X) increases the LFAP coverage percentage increases and the complete coverage is obtained using a low neighborhood depth value (i.e., X=2). This result is significant since failure notification message needs to be sent only to nodes which are two-hop away from the point of failure for the complete coverage. This result supports that our method provides fast convergence by introducing minimal signaling overhead within only the two-hop neighborhood.

2. The topologies with higher connectivity (i.e., higher E_mean values) have better LFAP coverage compared to the topologies with lower connectivity (i.e., lower E_mean values). This is an intuitive result since the number of possible alternate hops in dense network topologies is higher than the number of possible alternate hops in sparse topologies. This phenomenon increases the likelihood of finding LFAPs, and therefore the LFAP coverage percentage.

4.2 Convergence Analysis Based on Testbed Experiments

```
F-----------E------------A------------|
|           |            |            |
|           |            |            |
|           |            |            |
|           |            |            |
G-----------D------------|            B
            |                         |
            |                         |
            |                         |
```

```
               |                        |
               C------------------------|
```

Figure 13: 7-node network topology for local LFAP convergence experiments

```
   F-----------E------------A------------|
   |           |                         |
   |           |                         |
   |           |                         |
   |           |                         |
   G-----------D                         B
               |                         |
               |                         |
               |                         |
               |                         |
               C-------------------------|
```

Figure 14: 7-node network topology for remote LFAP convergence experiments

We performed the convergence analysis of our new fast reroute presented here using 7-node network consisting of 2600 and 3600 series Cisco routers as shown in Figures 13 and 14. Link costs are set to the same value for all links. A dedicated computer (e.g., an agent), which is running our fast reroute technology, is connected to each router through an Ethernet cable. These agents obtain the network topology from the routers in real-time by issuing an SNMP request. Using the retrieved network topology information, a set of LFAPs, which protects the failed primary paths when a real failure occurs, is pre-computed and stored. For both networks, the link between routers A and E is failed for all experiments.

The convergence time on the alternate path is measured by running a session, similar to a ping application, between routers A and E. When the link between routers A and E is failed on the topology as shown in Figure 13, the primary path A-E (and hence the session between them) fails. Once the failure is detected, the router A (E) reroutes its traffic over the alternate path A-D-E (E-D-A) by installing its pre-computed local LFAP. However, when the same scenario is applied to the topology in Figure 14, there is no local LFAP in the router A (E) to protect the primary path A-E (E-A) for this failure. For the above scenario, our fast reroute technology propagates the failure information to router B (D) which is already pre-computed a set of remote LFAPs for this particular failure. As a result, the session is rerouted through the alternate path A-B-C-D-E (E-D-C-B-A).

For both local and remote LFAP scenarios, the experiments are repeated for 10 times. The mean convergence time for the local LFAP

scenario is 602 milliseconds while the mean convergence time for the remote LFAP scenario is 612 milliseconds. Note that LFAPs are stored in the agents which are external to the routers. These agents issue an IOS command to install the right set of LFAPs when a failure is detected. In reality, the agent technology should run in the router as a GPP and LFAPs should be installed beforehand and waiting for a failure signal to be activated. Therefore, the results should be used to compare the difference between local and remote LFAPs rather than their absolute values.

The convergence times do not include the failure detection time since our primary objective in these experiments is to compare local and remote LFAP rather than proposing a new failure detection mechanism. For the sake of experiments, we implemented a heuristic based failure detection mechanism using periodic light-weight hearbeat messages similar to the Bidirectional Forwarding Detection. Note that the alternate path between A and E in the remote LFAP scenario is longer (A-D-E vs. A-B-C-D-E). The failure information is reached to one-hop neighbor within a few milliseconds since the round trip time between two neighboring agents is measured around 1-2 milliseconds. These results indicate that the convergence time of the remote LFAP mechanism is only slightly higher compared to the only local LFAP mechanism due to the failure notification. This increase is bounded by the neighborhood depth times a few milliseconds. However, the remote LFAP significantly increases the alternate path coverage since there is no local LFAP to protect the session between routers A and E when the link between routers A and E fails in Figure 14.

5. Scope and Applicability

This work is proposed initially for link state IGPs (i.e., OSPF and IS-IS). Further study is needed for extending its applicability to non-link state IGPs or BGP.

6. Acknowledgements

The authors would like to thank John Scudder, the co-chair of the IETF RTGWG, for his valuable comments and suggestions on the preliminary version of this draft.

7. References

Internet-drafts are works in progress available from <http://www.ietf.org/internet-drafts/>

[1] M. Shand and S. Bryant, "IP Fast Reroute Framework", draft-ietf-rtgwg-ipfrr-framework-06.txt, Oct. 2006 (work in progress).

[2] S. Bryant and M. Shand, "A Framework for Loop-free Convergence", draft-bryant-shand-lf-conv-frmwk-03.txt, Oct. 2006 (work in progress).

[3] Alex Zinin, "Analysis and Minimization of Microloops in Link-state Routing Protocols", draft-ietf-rtgwg-microloop-analysis-01.txt, Oct. 2005 (work in progress).

[4] Francois et. al., "Loop-free convergence using ordered FIB updates", <draft-francois-ordered-fib-02.txt>, Oct. 2006 (work in progress).

[5] S. Bryant, M. Shand, "IP Fast Reroute using tunnels", <draft-bryant-ipfrr-tunnels-02.txt>, Apr 2005 (work in progress).

[6] S. Bryant, M. Shand, and S. Previdi, "IP Fast Reroute Using Not-via Addresses", <draft-bryant-shand-ipfrr-notvia-addresses-03.txt>, Oct. 2006 (Work in progress).

[7] M. Gjoka, V. Ram, and X. Yang, "Evaluation of IP Fast Reroute Proposals", to appear in IEEE/Create-Net/ICST COMSWARE 2007.

[8] J. Moy, "OSPF Version 2", RFC 2328, April 1998.

[9] P. Francois, C. Filsfils, J. Evans, and O. Bonaventure, "Achieving sub-second IGP convergence in large IP network", SIGCOMM Comput. Commun. Rev., 35(3):35-44, 2005.

[10] M. Pitkanen and M. Luoma, "OSPF Flooding Process Optimization", Workshop on High Performance Switching and Routing (HPSR), pp.448-452, May 2005.

[11] G. Iannaccone et. al., "Analysis of link failures in an IP backbone", in Proc. ACM Sigcomm Internet Measurement Workshop, Nov. 2002.

[12] Cisco Systems, Inc., Cisco IOS IP Routing Protocols Configuration Guide, Release 12.4.

[13] R. Coltun, D. Ferguson, and J. Moy, "OSPF for IPv6", RFC 2740, December 1999.

[14] S. Lee, Y. Yu, S. Nelakuditi, Z. Zhang, and C. Chuah, "Proactive vs. reactive approaches to failure resilient routing", Proc. INFOCOM 2004.

[15] A. Atlas and A. Zinin, "Basic Specification for IP Fast-Rerorute: Loop-free Alternates", <draft-ietf-rtgwg-ipfrr-spec-base-06.txt>, March 2007 (work in progress).

[16] B. Fenner, M. Handley, H. Holbrook and I. Kouvelas, "Protocol

Independent Multicast (PIM-SM): Protocol Specification,"RFC 4601, August 2006.

[17] Oliver Heckmann et al., "How to use topology generators to create realistic topologies", Technical Report, Dec 2002.

8. Authors' Addresses

Ibrahim Hokelek
Applied Research,
Telcordia Technologies, Inc.
RRC-1E313
One Telcordia Drive,
Piscataway, NJ 08854
United States.   Email: ihokelek@research.telcordia.com

Mariusz A. Fecko
Applied Research,
Telcordia Technologies, Inc.
RRC-1E326
One Telcordia Drive,
Piscataway, NJ 08854
United States.   Email: mfecko@research.telcordia.com

Provin Gurung
Applied Research,
Telcordia Technologies, Inc.
RRC-1D305
One Telcordia Drive,
Piscataway, NJ 08854
United States.   Email: pgurung@research.telcordia.com

Sunil Samtani
Applied Research,
Telcordia Technologies, Inc.
RRC-1P387
One Telcordia Drive,
Piscataway, NJ 08854
United States.   Email: ssamtani@research.telcordia.com

Selcuk Cevher
Applied Research,
Telcordia Technologies, Inc.
RRC-1A212
One Telcordia Drive,
Piscataway, NJ 08854
United States.   Email: cevhers@research.telcordia.com

John Sucec

Applied Research,
Telcordia Technologies, Inc.
RRC-1G313
One Telcordia Drive,
Piscataway, NJ 08854
United States.   Email: sucecj@research.telcordia.com

Full Copyright Statement

Intellectual Property