Routing Area Working Group
Internet-Draft
Intended status: Informational
Expires: May 3, 2012

A. Atlas, Ed.
R. Kebler
M. Konstantynowicz
Juniper Networks
G. Enyedi
A. Csaszar
Ericsson
R. White
Cisco Systems
M. Shand

October 31, 2011

An Architecture for IP/LDP Fast-Reroute Using Maximally Redundant Trees

draft-atlas-rtgwg-mrt-frr-architecture-01

Abstract

As IP and LDP Fast-Reroute are increasingly deployed, the coverage limitations of Loop-Free Alternates are seen as a problem that requires a straightforward and consistent solution for IP and LDP, for unicast and multicast. This draft describes an architecture based on redundant backup trees where a single failure can cut a point-of-local-repair from the destination only on one of the pair of redundant trees.

One innovative algorithm to compute such topologies is maximally disjoint backup trees. Each router can compute its next-hops for each pair of maximally disjoint trees rooted at each node in the IGP area with computational complexity similar to that required by Dijkstra.

The additional state, address and computation requirements are believed to be significantly less than the Not-Via architecture requires.

Status of this Memo

time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 3, 2012.

Table of Contents

1. Introduction

There is still work required to completely provide IP and LDP Fast-Reroute[RFC5714] for unicast and multicast traffic. This draft proposes an architecture to provide 100% coverage.

Loop-free alternates (LFAs)[RFC5286] provide a useful mechanism for link and node protection but getting complete coverage is quite hard. [LFARevisited] defines sufficient conditions to determine if a network provides link-protecting LFAs and also proves that augmenting a network to provide better coverage is NP-hard. [I-D.ietf-rtgwg-lfa-applicability] discusses the applicability of LFA to different topologies with a focus on common PoP architectures.

While Not-Via [I-D.ietf-rtgwg-ipfrr-notvia-addresses] is defined as an architecture, in practice, it has proved too complicated and stateful to spark substantial interest in implementation or deployment. Academic implementations [LightweightNotVia] exist and have found the address management complexity high (but no standardization has been done to reduce this).

A different approach is needed and that is what is described here. It is based on the idea of using disjoint backup topologies as realized by Maximally Redundant Trees (described in [LightweightNotVia]); the general architecture could also apply to future improved redundant tree algorithms.

1.1. Goals for Extending IP Fast-Reroute coverage beyond LFA

Any scheme proposed for extending IPFRR network topology coverage beyond LFA, apart from attaining basic IPFRR properties, should also aim to achieve the following usability goals:

- ensure maximum physically feasible link and node disjointness regardless of topology,
- automatically compute backup next-hops based on the topology information distributed by link-state IGP,
- do not require any signaling in the case of failure and use pre-programmed backup next-hops for forwarding,
- introduce minimal amount of additional addressing and state on routers,
- enable gradual introduction of the new scheme and backward compatibility,

o and do not impose requirements for external computation.

2. Terminology

2-connected: A graph that has no cut-vertices. This is a graph that requires two nodes to be removed before the network is partitioned.

2-connected cluster: A maximal set of nodes that are 2-connected.

2-edge-connected: A network graph where at least two links must be removed to partition the network.

ADAG: Almost Directed Acyclic Graph - a graph that, if all links incoming to the root were removed, would be a DAG.

block: Either a 2-connected cluster, a cut-edge, or an isolated vertex.

cut-link: A link whose removal partitions the network. A cut-link by definition must be connected between two cut-vertices. If there are multiple parallel links, then they are referred to as cut-links in this document if removing the set of parallel links would partition the network.

cut-vertex: A vertex whose removal partitions the network.

DAG: Directed Acyclic Graph - a graph where all links are directed and there are no cycles in it.

GADAG: Generalized ADAG - a graph that is the combination of the ADAGs of all blocks.

Maximally Redundant Trees (MRT): A pair of trees where the path from any node X to the root R along the first tree and the path from the same node X to the root along the second tree share the minimum number of nodes and the minimum number of links. Each such shared node is a cut-vertex. Any shared links are cut-links. Any RT is an MRT but many MRTs are not RTs.

network graph: A graph that reflects the network topology where all links connect exactly two nodes and broadcast links have been transformed into the standard pseudo-node representation.

Redundant Trees (RT): A pair of trees where the path from any node X to the root R along the first tree is node-disjoint with the path from the same node X to the root along the second tree. These can be computed in 2-connected graphs.

3. Maximally Redundant Trees (MRT)

In the last few years, there's been substantial research on how to compute and use redundant trees. Redundant trees are directed spanning trees that provide disjoint paths towards their common root. These redundant trees only exist and provide link protection if the network is 2-edge-connected and node protection if the network is 2-connected. Such connectiveness may not be the case in real networks, either due to architecture or due to a previous failure. The work on maximally redundant trees has added two useful pieces that make them ready for use in a real network.

- Computable regardless of network topology: The maximally redundant trees are computed so that only the cut-edges or cut-vertices are shared between the multiple trees.

- Computationally practical algorithm is based on a common network topology database. Algorithm variants can compute in O(e) or O(e + n log n), as given in [I-D.enyedi-rtgwg-mrt-frr-algorithm].

There is, of course, significantly more in the literature related to redundant trees and even fast-reroute, but the formulation of the Maximally Redundant Trees (MRT) algorithm makes it very well suited to use in routers.

A known disadvantage of MRT, and redundant trees in general, is that the trees do not necessarily provide shortest detour paths. The use of the shortest-path-first algorithm in tree-building and including all links in the network as possibilities for one path or another should improve this. Modeling is underway to investigate and compare the MRT alternates to the optimal [I-D.enyedi-rtgwg-mrt-frr-algorithm]. Providing shortest detour paths would require failure-specific detour paths to the destinations, but the state-reduction advantage of MRT lies in the detour being established per destination (root) instead of per destination AND per failure.

The specific algorithm to compute MRTs as well as the logic behind that algorithm and alternative computational approaches are given in detail in [I-D.enyedi-rtgwg-mrt-frr-algorithm]. Those interested are highly recommended to read that document. This document describes how the MRTs can be used and not how to compute them.

The most important thing to understand about MRTs is that for each pair of destination-routed MRTs, there is a path from every node X to the destination D on the Blue MRT that is as disjoint as possible from the path on the Red MRT. The two paths along the two MRTs to a given destination-root of a 2-connected graph are node-disjoint, while in any non-2-connected graph, only the cut-vertices and cut-edges can be contained by both of the paths.

For example, in Figure 1, there is a network graph that is 2-connected in (a) and associated MRTs in (b) and (c). One can consider the paths from B to R; on the Blue MRT, the paths are B->F->D->E->R or B->F->C->E->R. On the Red MRT, the path is B->A->R. These are clearly link and node-disjoint. These MRTs are redundant trees because the paths are disjoint.

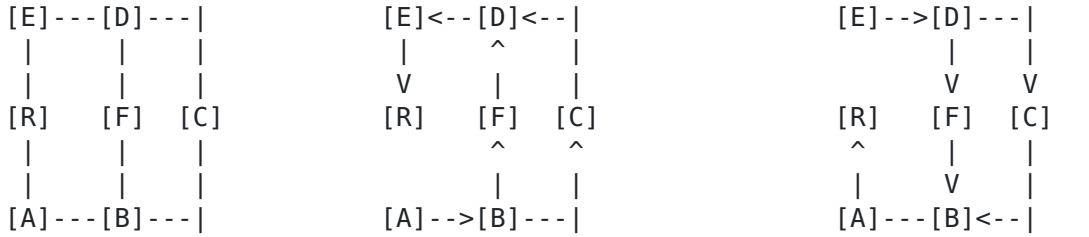


Figure 1: A 2-connected Network

By contrast, in Figure 2, the network in (a) is not 2-conneted. If F, G or the link F<->G failed, then the network would be partitioned. It is clearly impossible to have two link-disjoint or node-disjoint paths from G, I or J to R. The MRTs given in (b) and (c) offer paths that are as disjoint as possible. For instance, the paths from B to R are the same as in Figure 1 and the path from G to R on the Blue MRT is G->F->D->E->R and on the Red MRT is G->F->B->A->R.

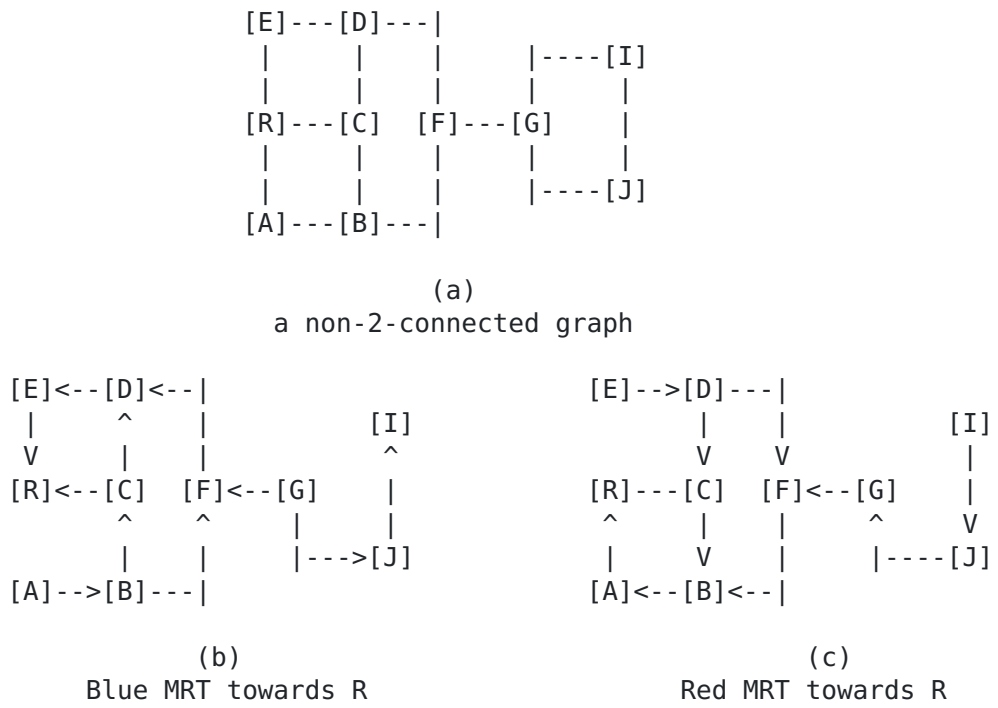


Figure 2: A non-2-connected network

4. Maximally Redundant Trees (MRT) and Fast-Reroute

In normal IGP routing, each router has its shortest-path-tree to all destinations. From the perspective of a particular destination, D, this looks like a reverse SPT (rSPT). To use maximally redundant trees, in addition, each destination D has two MRTs associated with it; by convention these will be called the blue and red MRTs.

MRTs are practical to maintain redundancy even after a single link or node failure. If a pair of MRTs is computed rooted at each destination, all the destinations remain reachable along one of the MRTs in the case of a single link or node failure.

When there is a link or node failure affecting the rSPT, each node will still have at least one path via one of the MRTs to reach the destination D. For example, in Figure 2, C would normally forward traffic to R across the C<->R link. If that C<->R link fails, then C could use either the Blue MRT path C->D->E->R or the Red MRT path C->B->A->R.

As is always the case with fast-reroute technologies, forwarding does not change until a local failure is detected. Packets are forwarded

along the shortest path. The appropriate alternate to use is pre-computed. [I-D.enyedi-rtgwg-mrt-frr-algorithm] describes exactly how to determine whether the Blue MRT next-hops or the Red MRT next-hops should be the MRT alternate next-hops for a particular primary next-hop N to a particular destination D.

MRT alternates are always available to use, unless the network has been partitioned. It is a local decision whether to use an MRT alternate, a Loop-Free Alternate or some other type of alternate. When a network needs to use a micro-loop prevention mechanism [RFC5715] such as Ordered FIB[I-D.ietf-rtgwg-ordered-fib] or Farside Tunneling[RFC5715], then the whole IGP area needs to have alternates available so that the micro-loop prevention mechanism, which requires slower network convergence, can take the necessary time without impacting traffic badly.

As described in [RFC5286], when a worse failure than is anticipated happens, using LFAs that are not downstream neighbors can cause micro-looping. An example is given of link-protecting alternates causing a loop on node failure. Even if a worse failure than anticipated happened, the use of MRT alternates will not cause looping. Therefore, while node-protecting LFAs may be prefered, there are advantages to using MRT alternates when such a node-protecting LFA is not a downstream path.

4.1. Multi-homed Prefixes

One advantage of LFAs that is necessary to preserve is the ability to protect multi-homed prefixes against ABR failure. For instance, if a prefix from the backbone is available via both ABR A and ABR B, if A fails, then the traffic should be redirected to B. This can also be done for backups via MRT.

This generalizes to any multi-homed prefix. A multi-homed prefix could be:

- An out-of-area prefix announced by more than one ABR,
- An AS-External route announced by 2 or more ASBRs,
- A prefix with iBGP multipath to different ASBRs,
- etc.

For each prefix, the two lowest total cost ABRs are selected and a proxy-node is created connected to those two ABRs. If there exist multiple multi-homed prefixes that share the same two best connectivity, then a single proxy-node can be used to represent the

set. An example of this is shown in Figure 3.

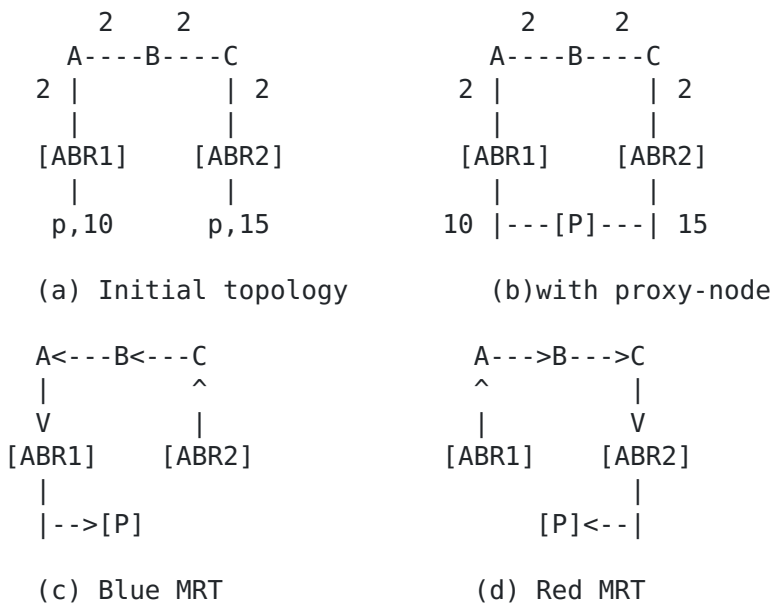


Figure 3: Prefixes Advertised by Multiple ABRs

The proxy-nodes and associated links are added to the network topology after all real links have been assigned to a direction and before the actual MRTs are computed. Proxy-nodes cannot be transited when computing the MRTs. In addition to computing the pair of MRTs associated with each router destination D in the area, a pair of MRTs can be computed for each such proxy-node to fully protect against ABR failure.

Each ABR or attaching router must remove the MRT marking[see Section 4.2] and then forward the traffic outside of the area (or island of MRT-fast-reroute-supporting routers).

When directing traffic along an MRT towards a multi-homed prefix, if a topology-identifier label[see Section 4.2.1] is not used, then the proxy-node must be named and either additional LDP labels or IP addresses associated with it.

4.2. Unicast Forwarding with MRT Fast-Reroute

With LFA, there is no need to tunnel unicast traffic, whether IP or LDP. The traffic is simply sent to an alternate. The behavior with MRT Fast-Reroute is different depending upon whether IP or LDP unicast traffic is considered.

Logically, one could use the same IP address or LDP FEC and then also use 2 bits to express the topology to use. The topology options are (00) IGP/SPT, (01) blue MRT, (10) red MRT. Unfortunately, there just aren't 2 spare bits available in the IPv4 or IPv6 header. This has different consequences for IP and LDP because LDP can just add a topology label on top or take 2 spare bits from the label space.

Once the MRTs are computed, the two sets of MRTs are seen by the forwarding plane as essentially two additional topologies. The same considerations apply for forwarding along the MRTs as for handling multiple topologies.

4.2.1. LDP Unicast Forwarding - Avoid Tunneling

For LDP, it is very desirable to avoid tunneling because, for at least node protection, tunneling requires knowledge of remote LDP label mappings and thus requires targeted LDP sessions and the associated management complexity. There are two different mechanisms that can be used.

1. Option A - Encode Topology in Labels: In addition to sending a single label for a FEC, a router would provide two additional labels with their associated MRT colors. This is simple, but reduces the label space for other uses. It also increases the memory to store the labels and the communication required by LDP.

2. Option B - Create Topology-Identification Labels: Use the label-stacking ability of MPLS and specify only two additional labels - one for each associated MRT color - by a new FEC type. When sending a packet onto an MTR, first swap the LDP label and then push the topology-identification label for that MTR color. When receiving a packet with a topology-identification label, pop it and use it to guide the next-hop selection in combination with the next label in the stack; then swap the remaining label, if appropriate, and push the topology-identification label for the next-hop. This has minimal usage of additional labels, memory and LDP communication. It does increase the size of packets and the complexity of the required label operations and look-ups. This can use the same mechanisms as are needed for context-aware label spaces.

Note that with LDP unicast forwarding, regardless of whether topology-identification label or encoding topology in label is used, no additional loopbacks per router are required as are required in the IP unicast forwarding case. This is because LDP labels are used on a hop-by-hop basis to identify MRT-blue and MRT-red forwarding trees.

For greatest hardware compatibility, routers should support Option B of encoding the topology in the labels.

4.2.1.1. Protocol Extensions and Considerations: LDP

This captures an initial understanding of what may need to be specified.

1. Specify Topology in Label: When sending a Label Mapping, have the ability to send a Label TLV and multiple Topology-Label TLVs. The Topology-Label TLV would specify MRT and the associated MRT color.

2. Topology-Identification Labels: Define a new FEC type that describes the topology for MRT and the associated MRT color.

4.2.2. IP Unicast Traffic

For IP, there is no currently practical alternative except tunneling. The tunnel egress could be the original destination in the area, the next-next-hop, etc.. If the tunnel egress is the original destination router, then the traffic remains on the redundant tree with sub-optimal routing. If the tunnel egress is the next-next-hop, then protection of multi-homed prefixes and node-failure for ABRs is not available. Selection of the tunnel egress is a router-local decision.

There are three options available for marking IP packets with which MRT it should be forwarded in.

1. Tunnel IP packets via an LDP LSP. This has the advantage that more installed routers can do line-rate encapsulation and decapsulation. Also, no additional IP addresses would need to be allocated or signaled.

    A. Option A - LDP Destination-Topology Label: Use a label that indicates both destination and MRT. This method allows easy tunneling to the next-next-hop as well as to the IGP-area destination. For multi-homed prefixes, this requires that additional labels be advertised for each proxy-node.

    B. Option B - LDP Topology Label: Use a Topology-Identifier label on top of the IP packet. This is very simple and doesn't require additional labels for proxy-nodes. If tunneling to a next-next-hop is desired, then a two-deep label stack can be used with [Topology-ID label, Next-Next-Hop Label].

2. Tunnel IP packets in IP. Each router supporting this option would announce two additional loopback addresses and their associated MRT color. Those addresses are used as destination addresses for MRT-blue and MRT-red IP tunnels respectively. They allow the transit nodes to identify the traffic as being forwarded along either MRT-blue or MRT-red tree topology to reach the tunnel destination. Announcements of these two additional loopback addresses per router with their MRT color requires IGP extensions.

For proxy-nodes associated with one or more multi-homed prefixes, the problem is harder because there is no router associated with the proxy-node, so its loopbacks can't be known or used. In this case, each router attached to the proxy-node could announce two common IP addresses with their associated MRT colors. This would require configuration as well as the previously mentioned IGP extensions. Similarly, in the LDP case, two additional FEC bindings could be announced.

4.2.2.1. Protocol Extensions and Considerations: OSPF and ISIS

This captures an initial understanding of what may need to be specified.

- Capabilities: Does a router support MRT? Does the router do MRT tunneling with LDP or IP or GRE or...?

- Topology Association: A router needs to advertise a loopback and associate it with an MRT whether blue or red. Additional flexibility for future uses would be good.

- Proxy-nodes for Multi-homed Prefixes: We need a way to advertise common addresses with MRT for multi-homed prefixes' proxy-nodes. Currently, those proxy-nodes aren't named or considered.

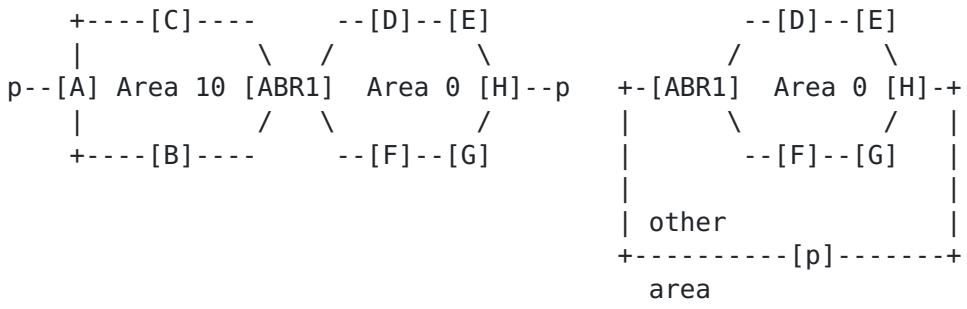
As with LFA, it is expected that OSPF Virtual Links will not be supported.

4.2.3. Inter-Area and ABR Forwarding Behavior

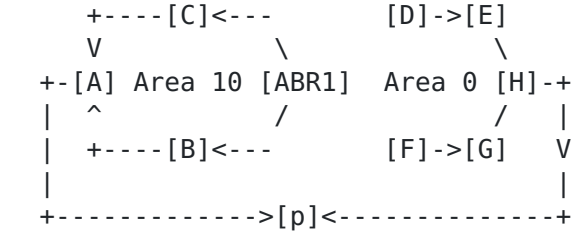
In regular forwarding, packets destined outside the area arrive at the ABR and the ABR forwards them into the other area because the next-hops from the area with the best route (according to tie-breaking rules) are used by the ABR. The question is then what to do with packets marked with an MRT that are received by the ABR.

The only option that doesn't require forwarding based upon incoming interface is to forward an MRT marked packet in the area with the

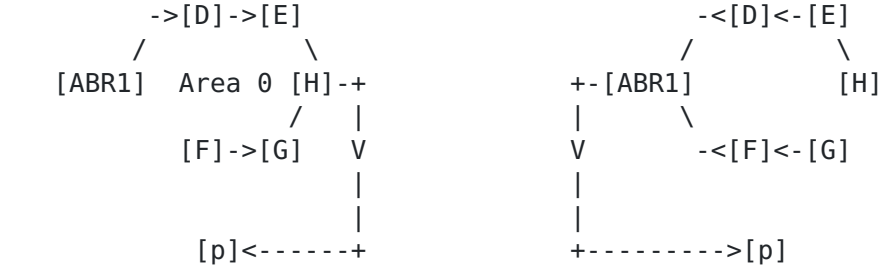
best route along its associated MRT. If the packet came from that area, this correctly avoids the failure. If the packet came from a different area, at least this gets the packet to the destination even though it is along an MRT rather than the shortest-path.



(a) Example topology (b) Proxy node view in Area 0 nodes



(c) rSPT towards destination p



(d) Blue MRT in Area 0 (e) Red MRT in Area 0

Figure 4: ABR Forwarding Behavior and MRTs

To avoid using an out-of-area MRT, special action can be taken by the penultimate router along the in-local-area MRT immediately before the ABR is reached. The penultimate router can determine that the ABR

will forward the packet out of area and, in that case, the penultimate router can remove the MRT marking but still forward the packet along the MRT next-hop to reach the ABR. For instance, in Figure 4, if node H fails, node E has to put traffic towards prefix p onto the red MRT. But since node D knows that ABR1 will use a best from another area, it is safe for D to remove the MRT marking and just send the packet to ABR1 still on the red MRT but unmarked. ABR1 will use the shortest path in Area 10.

In all cases for ISIS and most cases for OSPF, the penultimate router can determine what decision the adjacent ABR will make. The one case where it can't be determined is when two ASBRs are in different non-backbone areas attached to the same ABR, then the ASBR's Area ID may be needed for tie-breaking (prefer the route with the largest OPSF area ID) and the Area ID isn't announced as part of the ASBR link-state advertisement (LSA). In this one case, suboptimal forwarding along the MRT in the other area would happen. If this is a realistic deployment scenario, OSPF extensions could be considered.

4.2.4. Issues with Area Abstraction

MRT fast-reroute provides complete coverage in a area that is 2-connected. Where a failure would partition the network, of course, no alternate can protect against that failure. Similarly, there are ways of connecting multi-homed prefixes that make it impractical to protect them without excessive complexity.

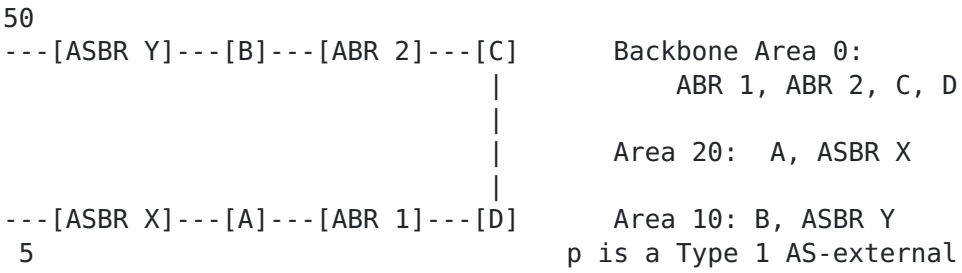


Figure 5: AS external prefixes in different areas

Consider the network in Figure 5 and assume there is a richer connective topology that isn't shown, where the same prefix is announced by ASBR X and ASBR Y which are in different non-backbone areas. If the link from A to ASBR X fails, then an MRT alternate could forward the packet to ABR 1 and ABR 1 could forward it to D, but then D would find the shortest route is back via ABR 1 to Area 20. The only real way to get it from A to ASBR Y is to explicitly tunnel it to ASBR Y.

Tunnelling to the backup ASBR is for future consideration. The previously proposed PHP approach needs to have an exception if BGP policies (e.g. BGP local preference) determines which ASBR to use. Consider the case in Figure 6. If the link between A and ASBR X (the preferred border router) fails, A can put the packets to p onto an MRT alternate, even tunnel it towards ASBR Y. Node B, however, must not remove the MRT marking in this case, as nodes in Area 0, including ASBR Y itself would not know that their preferred ASBR is down.

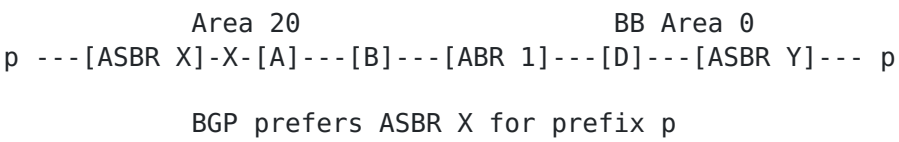


Figure 6: Failure of path towards ASBR preferred by BGP

The fine details of how to solve multi-area external prefix cases, or identifying certain cases as too unlikely and too complex to protect is for further consideration.

4.2.5. Partial Deployment and Islands of Compatible MRT FRR routers

A natural concern with new functionality is how to have it be useful when it is not deployed across an entire IGP area. In the case of MRT FRR, where it provides alternates when appropriate LFAs aren't available, there are also deployment scenarios where it may make sense to only enable some routers in an area with MRT FRR. A simple example of such a scenario would be a ring of 6 or more routers that is connected via two routers to the rest of the area.

First, a computing router S must determine its local island of compatible MRT fast-reroute routers. A router that has common forwarding mechanisms and common algorithm and is connected to either to S or to another router already determined to be in S's local island can be added to S's local island.

Destinations inside the local island can obviously use MRT alternates. Destinations outside the local island can be treated like a multi-homed prefix with caveats to avoid looping. For LDP labels including both destination and topology, the routers at the borders of the local island need to originate labels for the original FEC and the associated MRT-specific labels. Packets sent to an LDP label marked as blue or red MRT to a destination outside the local island will have the last router in the local island swap the label to one for the destination and forward the packet along the outgoing

interface on the MRT towards a router outside the local island that was represented by the proxy-node.

For IP in IP encapsulations, remote destinations may not be advertising additional IP loopback addresses for the MRTs. In that case, a router attached to a proxy-node, which represents destinations outside the local island, must advertise IP addresses associated with that proxy-node. Packets sent to an address associated with a proxy-node will have their outer IP header removed by the router attached to the proxy-node and be forwarded by the router along the outgoing interface on the MRT towards a router outside the local island that was represented by the proxy-node.

4.2.6. Network Convergence and Preparing for the Next Failure

After a failure, MRT detours ensure that packets reach their intended destination while the IGP has not reconverged onto the new topology. As link-state updates reach the routers, the IGP process calculates the new shortest paths. Two things need attention: micro-loop prevention and MRT re-calculation.

4.2.6.1. Micro-forwarding loop prevention and MRTs

As is well known[RFC5715], micro-loops can occur during IGP convergence; such loops can be local to the failure or remote from the failure. Managing micro-loops is an orthogonal issue to having alternates for local repair, such as MRT fast-reroute provides.

There are two possible micro-loop prevention mechanism discussed in [RFC5715]. The first is Ordered FIB [I-D.ietf-rtgwg-ordered-fib]. The second is Farside Tunneling which requires tunnels or an alternate topology to reach routers on the farside of the failure.

Since MRTs provide an alternate topology through which traffic can be sent and which can be manipulated separately from the SPT, it is possible that MRTs could be used to support Farside Tunneling. Details of how to do so are outside of this document.

4.2.6.2. MRT Recalculation

When a failure event happens, traffic is put by the PLRs onto the MRT topologies. After that, each router recomputes its shortest path tree (SPT) and moves traffic over to that. Only after all the PLRs have switched to using their SPTs and traffic has drained from the MRT topologies should each router install the recomputed MRTs into the FIBs.

At each router, therefore, the sequence is as follows:

1. Receive failure notification

2. Recompute SPT

3. Install new SPT

4. Recompute MRTs

5. Wait configured period for all routers to be using their SPTs and traffic to drain from the MRTs.

6. Install new MRTs.

While the recomputed MRTs are not installed in the FIB, protection coverage is lowered. Therefore, it is important to recalculate the MRTs and install them as quickly as possible.

It is for further study whether MRT re-calculation is possible in an incremental fashion, such that the sections of the MRT in use after a failure are not changed.

4.3. Multicast and MRT Fast-Reroute

There are several basic issues with doing Fast-Reroute for multicast traffic, whether the alternates used are LFA or MRT. They are given below:

1. The Point-of-Local-Repair (PLR) does not know the set of next-next-hops in the multicast tree.

2. A potential Merge Point(MP) does not know its previous-previous-hop in the multicast tree.

3. For mLDP, the PLR does not know the appropriate labels to use for the next-next-hops in the multicast tree.

4. The Merge Point (MP) does not know upon what interface to expect backup traffic. For LFAs, this is a particular issue since the LFA selected by a PLR is known only to that PLR.

Additionally, fast-reroute is to protect against a link failure, a node failure, or even local SRLG or general SRLG failures, but the mechanisms for such detection cannot distinguish easily between a link failure and a node failure (much less more complicated failures). In unicast forwarding, the assumption can be made that any failure is a node failure, unless the destination is the next-hop, and traffic is simply forwarded to the final destination avoiding the next-hop. For multicast, the final destination is not

useful - what matters is the set of next-hop routers and the set of next-next-hop routers reached via each of the next-hop routers on the relevant multicast tree.

In multicast, it is possible that traffic is required by the next-hop as well as the next-next-hop and beyond. Therefore, whenever a local failure is detected and node protection is configured, it may be necessary to send traffic to both the affected next-hop routers and the set of next-next-hops reached via those next-hop routers.

4.3.1. Traffic Handling

When the PLR detects a failure, it forwards the multicast traffic on the link-protecting alternates. If node-protection is desired, then the traffic is also replicated to the node-protecting alternates.

The PLR sends traffic on the alternates for a configurable time-out. There is no clean way for the next-hop routers and/or next-next-hop routers to indicate that the traffic is no longer needed.

Critically, the potential Merge Point can independently determine whether to accept alternate traffic. If the primary upstream link(s) have failed, then accept and forward alternate traffic. When traffic is received on a new primary upstream link, stop accepting and forwarding alternate traffic.

This MP behavior involves a new action on detecting a local failure. When the local failure is detected, if that was the last primary upstream link, then the associated FIB entry for the alternate traffic is updated from discard to forward.

The final question is can anything be done about traffic missed due to different latencies along new primary and alternate/old primary trees? Any such techniques are outside the scope of this document.

4.3.2. PLR Replication and Tunneled

The disadvantages of tunneling unicast traffic do not fully translate to those for multicast. With MRT fast-reroute, IP unicast traffic is tunneled. With mLDP, with the suggested extensions, along with learning the next-next-hops on the multicast tree, the associated labels can be learned so there is no need for targeted sessions. If multicast traffic weren't tunneled, then multicast state would need to be created ahead of the failure along the alternate paths.

In this approach, the PLR tunnels multicast traffic into the unicast alternates destined to each particular MP. This is simply PLR-replication. For node-protection, the PLR learns of the MPs and

their labels via protocol extensions[See Section 4.3.4.1 and Section 4.3.5].

The downside of PLR replication is that the same packets may appear multiple times on a link if they are tunneled to different destinations. The upside is that PLR replication avoids creating any alternate multicast state in the network.

4.3.3. Alternate Trees

To minimize replication of packets, it is possible to create alternate-trees. Each alternate-tree would be for a given PLR and neighbor - the alternate-tree would be failure-specific. It is not possible to merge alternate-trees for different PLRs or for different neighbors. This is shown in Figure 7 where G can't select an acceptable upstream node on the alternate tree that doesn't violate either the need to avoid C (for PLR A) or D (for PLR B).

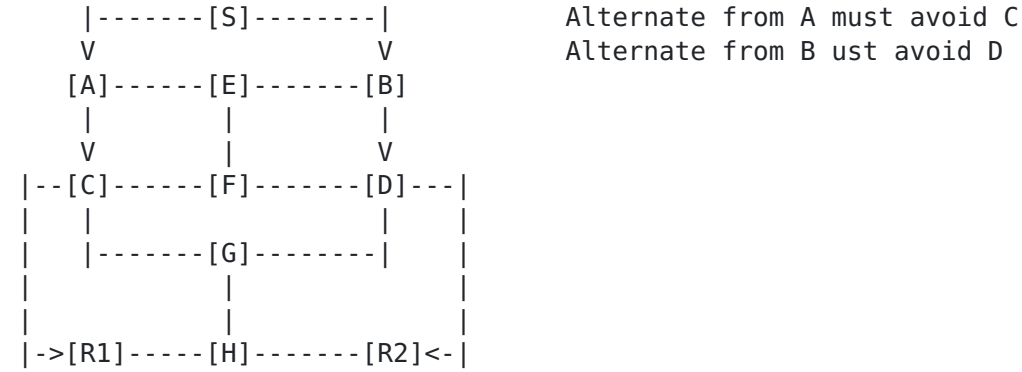


Figure 7: Alternate Trees from PLR A and B can't be merged

Backup Joins can be used to create the per-failure-point alternate trees. A Backup Join would indicate the PLR and the node to avoid. Each router that receives the Backup Join would determine which of the Blue MRT or Red MRT could offer an acceptable path and forward the traffic that way.

This method is still under investigation and consideration as its scaling properties are unfortunate.

4.3.3.1. Protocol Extensions

To create alternates from the potential Merge Points to the PLR and provide the MP and PLR with sufficient information, the following protocol extensions are needed.

- Extend PIM and mLDP to signal Backup Joins: A backup Join can be sent from the MP towards the PLR going hop-by-hop.

- Extend PIM and mLDP to send Join Confirmations with upstream router information. This provides the MP with information about the PLR for node protection scenarios.

4.3.4. PIM Forwarding

For node-protection, the merge points would be the next-next-hops in the tree. For a PLR to learn them, additional PIM Join Attributes [RFC5384] need to be defined to specify the set of next-hops from which the sending node has received Joins. For link-protection, of course a PLR knows the address of the neighbor.

PIM currently sends its JoinPrune messages periodically (60 seconds by default). Upon a change to the next-next hop list, the router can send a triggered JoinPrune with the updated Join Attribute, or it can wait for the next periodic refresh. It would be a tradeoff of increased control messages against a window of being unprotected.

Once the failure is detected, the PLR will send the traffic encapsulated to the list of downstream MPs. The PLR will send the encapsulated traffic for the duration of the protection-timeout. The protection-timer starts when the PLR detects a local failure. Once the timeout expires, the PLR can then prune upstream if there are no longer any receivers after the failure.

As is done today, the MP will forward traffic received on its normal incoming interface. If that interface fails, the MP will forward traffic if it is received with the correct encapsulation. After the incoming interface changes and new traffic arrives on the new incoming interface, received encapsulated traffic will not be forwarded until the protection-timer expires. This reduces sending of duplicate traffic at the cost of being briefly unprotected after a failure event.

4.3.4.1. Protocol Extensions and Considerations: PIM

This captures an initial understanding of what may need to be specified. This is focusing on PIM Sparse mode.

- Capabilities: New Hello Option Capabilities to indicate the ability to understand the new Join Attributes.

- Next-Hops: Need a new Join Attribute[RFC5384] to send the next-hops and the type of acceptable encapsulation to the PLR.

4.3.5. mLDP Forwarding

As in PIM, in mLDP[I-D.ietf-mpls-ldp-p2mp] a mechanism must be added so that the PLR can learn the next-next-hops. The PLR also needs to learn the associated label-bindings. This can be done via a new P2MP Child Data Object. This object would include the primary loopback of an LSR that has provided labels for the FEC to the sending LSR along with the label specified. Multiple P2MP Child Data Objects could be included in a P2MP Label Mapping; only those specified in the most recent P2MP Label Mapping should be stored and used.

This will provide the PLR with the MPs and their associated labels. The MPs will accept traffic received with that label from any interface, so no signaling is required before the alternates are used.

Traffic sent out each alternate will be tunneled with a destination of the MP.

4.4. Live-Live Multicast

In MoFRR [I-D.karan-mofrr], the idea of joining both a primary and a secondary tree is introduced with the requirement that the primary and secondary trees be link and node disjoint. This works well for networks where there are dual-planes, as explained in [I-D.karan-mofrr]. For other networks, it may still be desirable to have two disjoint multicast trees and allow a receiver to join both and make its own decision about what to do.

Using MRTs gives the ability to guarantee that the two trees are as disjoint as possible and to dynamically recompute the two MRTs whenever the topology changes.

Unlike for fast-reroute where the MRTs are rooted at the destination, with Live-Live Multicast, the MRTs would be routed at the multicast group source S. If the multicast source S is in a different area, then it could be represented via a proxy-node. If asymmetric link costs aren't a concern, then the same set of next-hops (previous-hops in this case) could be used as is used for MRT fast-reroute. A new P2MP FEC with Tree Identifier Element would need to be defined; it would include the topology to be used which could be IGP, MRT red, or MRT blue. For PIM, the existing PIM MT-ID Join

Attribute[I-D.ietf-pim-mtid] could be used to specify which MRT to use (blue or red).

For PIM, a different group could be used on the Blue MRT than on the Red MRT. Similarly, a different Opaque-Value could be used in mLDP for the Blue MRT and the Red MRT. Receiving routers would join both the blue MRT group and the red MRT group to receive traffic.

4.4.1. Forwarding Plane

If the two MRTs are not fully disjoint due to a network with a single point of failure, then traffic must self-identify as to which P2MP tree it belongs to. This means there must be a way to distinguish packets on the blue-MRT from the red-MRT. When different multicast groups are used, this is quite straightforward. For PIM, packets on the blue MRT would be destined to the group G-blue and packets on the red MRT would be destined to the group G-red. For mLDP, different labels will have been distributed for the Opaque-Value-blue and for the Opaque-Value-red.

RPF checks would still be enabled by the control plane. The control plane can program differnet forwarding entries on the G-blue incoming interface and on the G-red incoming interface. The other interfaces would still discard both G-blue and G-red traffic.

The receiver would still need to detect failures and handle traffic discarding as is specified in [I-D.karan-mofrr].

5. Acknowledgements

The authors would like to thank Hannes Gredler, Jeff Tantsura, Ted Qian, Kishore Tiruveedhula, Santosh Esale, Nitin Bahadur, Harish Sitaraman and Raveendra Torvi for their suggestions and review.

6. IANA Considerations

This doument includes no request to IANA.

7. Security Considerations

This architecture is not currently believed to introduce new security concerns.

8. References

8.1. Normative References

[I-D.enyedi-rtgwg-mrt-frr-algorithm]
Atlas, A., Envedi, G., and A. Csaszar, "Algorithms for computing Maximally Redundant Trees for IP/LDP Fast-Reroute", draft-enyedi-rtgwg-mrt-frr-algorithm-00 (work in progress), October 2011.

[I-D.ietf-mpls-ldp-p2mp]
Minei, I., Wijnands, I., Kompella, K., and B. Thomas, "Label Distribution Protocol Extensions for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths", draft-ietf-mpls-ldp-p2mp-15 (work in progress), August 2011.

[I-D.ietf-pim-mtid]
Cai, Y. and H. Ou, "PIM Multi-Topology ID (MT-ID) Join Attribute", draft-ietf-pim-mtid-10 (work in progress), September 2011.

[I-D.karan-mofrr]
Karan, A., Filsfils, C., Farinacci, D., Decraene, B., Leymann, N., and T. Telkamp, "Multicast only Fast Re-Route", draft-karan-mofrr-01 (work in progress), March 2011.

[RFC5286] Atlas, A. and A. Zinin, "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, September 2008.

[RFC5384] Boers, A., Wijnands, I., and E. Rosen, "The Protocol Independent Multicast (PIM) Join Attribute Format", RFC 5384, November 2008.

[RFC5714] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, January 2010.

8.2. Informative References

[I-D.ietf-rtgwg-ipfrr-notvia-addresses]
Shand, M., Bryant, S., and S. Previdi, "IP Fast Reroute Using Not-via Addresses", draft-ietf-rtgwg-ipfrr-notvia-addresses-07 (work in progress), April 2011.

[I-D.ietf-rtgwg-lfa-applicability]
Filsfils, C., Francois, P., Shand, M., Decraene, B., Uttaro, J., Leymann, N., and M. Horneffer, "LFA applicability in SP networks",

draft-ietf-rtgwg-lfa-applicability-03 (work in progress), August 2011.

[I-D.ietf-rtgwg-ordered-fib]
Shand, M., Bryant, S., Previdi, S., and C. Filsfils, "Loop-free convergence using oFIB", draft-ietf-rtgwg-ordered-fib-05 (work in progress), April 2011.

[LFARevisited]
Retvari, G., Tapolcai, J., Enyedi, G., and A. Csaszar, "IP Fast ReRoute: Loop Free Alternates Revisited", Proceedings of IEEE INFOCOM , 2011, <http://opti.tmit.bme.hu/~tapolcai/papers/retvari2011lfa_infocom.pdf>.

[LightweightNotVia]
Enyedi, G., Retvari, G., Szilagyi, P., and A. Csaszar, "IP Fast ReRoute: Lightweight Not-Via without Additional Addresses", Proceedings of IEEE INFOCOM , 2009, <http://mycite.omikk.bme.hu/doc/71691.pdf>.

[RFC5715] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, January 2010.

Authors' Addresses

Alia Atlas (editor)
Juniper Networks
10 Technology Park Drive
Westford, MA 01886
USA

Email: akatlas@juniper.net

Robert Kebler
Juniper Networks
10 Technology Park Drive
Westford, MA 01886
USA

Email: rkebler@juniper.net

Maciek Konstantynowicz
Juniper Networks

Email: maciek@juniper.net

Gabor Sandor Enyedi
Ericsson
Konyves Kalman krt 11.
Budapest 1097
Hungary

Email: Gabor.Sandor.Enyedi@ericsson.com

Andras Csaszar
Ericsson
Konyves Kalman krt 11
Budapest 1097
Hungary

Email: Andras.Csaszar@ericsson.com

Russ White
Cisco Systems

Email: russwh@cisco.com

Mike Shand

Email: mike@mshand.org.uk