Enhancing Repair Coverage of Loop-Free Alternates

S Sae Lor and M Rio

University College London

Abstract: Network reliability is an important measure for deployability of sensitive applications. A forwarding discontinuation due to link or node failures can be very damaging. Loop-Free Alternates (LFAs) are the simplest techniques used for IP fast re-route. Although LFAs incur low overheads, their repair coverage heavily depends on the underlying topology. This paper proposes the Enhanced Loop-Free Alternates (E-LFAs), which employs a simple recursive method on existing LFAs. Our simulation results show that the repair coverage is significantly improved and becomes near optimal using E-LFAs.

1 Introduction.

Network reliability problems involve the minimisation of packet loss in the presence of failures. Several emerging services cannot afford to rely on traditional routing paradigm due to unavoidable damage caused during network re-convergence. This increases the demand for a highly reliable network.

Several approaches such as multi-path and multi-homing routing [1], [2], [3] and overlay networks have been proposed to alleviate this problem. In this paper, we focus on fast re-route and recovery approach defined in IP Fast Re-Route (IPFRR) framework [5]. Basically, it specifies two main components for providing a disruption-free forwarding, which are fast failure detection and repair paths for fast re-route mechanisms. Achieving fast failure detection can be done by tweaking the protocol parameters [6]. That is, setting an appropriate Hello interval. Nonetheless, the amount of packets being dropped from the time the actual failure occurs until it is detected by a router is unavoidable. Thus, it is important to employ a mechanism that permits a router to immediately re-route the traffic for affected destinations via other paths. Several analyses [7], [8] show that most failures are transient (*i.e.* short-lived) and more than 50% last less than a minute. Consequently, most IPFRR techniques [9], [10], [11] focus on handling transient failures.

Recently, many techniques such as Loop-Free Alternates (LFAs) [10], U-turns alternates [12], tunnel [13], not-via addresses [11], and Failure Insensitive Routing (FIR) [9] have been introduced. We believe that LFAs are the most feasible solutions for resilient routing in IP networks due to their minimal requirements. Nevertheless, their repair coverage depends heavily on the underlying network topology, which can be as low as 60-70% of all protectable elements. This paper aims to show that by applying a simple recursive method in order to obtain the Enhanced Loop-Free Alternates (E-LFAs), the repair coverage of a network becomes near optimal without using any mechanism that may degrade the router performance.

2. Enhanced Loop-Free Alternates

Our algorithm employs similar conditions used to find LFAs. Normal alternate next hops are used in our technique wherever possible. However, for destinations without any eligible candidates, we compute E-LFAs using a simple recursive method to enhance the performance.

According to the basic specification for IPFRR with LFAs [10], the neighbour nodes can be classified by their abilities as alternate next hops as follows:

- Loop-Free Condition (LFC): a neighbour that satisfies this condition can be used as an LFA for link protection.
- Node-Protection Condition (NPC): a neighbour that satisfies this condition can be used as an LFA for node protection.

- Downstream Condition (DSC): a neighbour that satisfies this condition can be used as an LFA for a loop-free protection in case of multiple failures.
- Equal-Cost Alternates (ECA): routing via a neighbour that satisfies this condition offers an equal-cost path.

The notion of having DSC is that network operators may want to avoid short period of forwarding loops in the presence of multiple failures. Nevertheless, E-LFAs focus only on whether a neighbour satisfies LFC or NPC without having to worry that a forwarding loop may occur and lavishing the network capacity. Routing a packet from s to d with a primary next hop n_h, a neighbour of s, node n_i can be used as an LFC if it satisfies (1) and NPC if it satisfies (2).

$$cost(n_i, d) < cost(n_i, s) + cost(s, d) \tag{1}$$

$$cost(n_i, d) < cost(n_i, n_h) + cost(n_h, d) \tag{2}$$

If neither of these conditions is met, there is no LFAs from s to d. Similar to U-turn [12], we believe that although in several cases the detecting node may have no LFAs, one of its neighbours might have. U-turn employs an interface-specific mechanism and considers only the immediate neighbours. In contrast, our algorithm does not involve interface-specific mechanism or limit LFAs finding at the adjacent nodes.

Let $G = (V, E)$ be the graph with vertices V and edges E representing the network. Each source node s in V computes an LFA for each destination normally. First, we use the primary next hop as LFA if it does not exist. For each destination d in V without an original LFA, s runs the algorithm to find an appropriate E-LFA. Denote N as a set of neighbours of s with an LFA to d, their alternate next hops become E-LFA candidates. If the candidate satisfies LFA condition (*i.e.* LFC or NPC), it can be used as E-LFA. In many cases, LFAs of the neighbour nodes do not satisfy the LFA condition. We repeat our algorithm by considering LFA of LFA of the neighbours if it exists. This process iterates until either an E-LFA for d is found or the node being considered has been previously determined.

In the normal case, a router forwards the packet via the shortest path. When a failure occurs, the detecting node forwards the packet with existing LFAs. However, for destinations without original LFAs but E-LFAs, a packet must be marked with the number of recursions it has to be forwarded using local LFAs. When a node receives a re-routed packet, it decrements the number of recursions and forwards it using its local LFA. Once the number of recursions reaches zero, the packet can be forwarded using the normal path.

Since our routing technique considers either LFC or NPC condition, an extra bit must be marked to indicate a re-routed packet which will be dropped if it encounters two or more failures. To enable routing using LFAs, a router does not have to store any additional routing table entries. However, each entry for existing destinations must be enhanced with additional information about the next hop (E-LFA) and the number of recursions. As the failures are not limited to single type (*i.e.* it can be either link or node failures), it might be worth to enhance the routing table entries with information for each type of failures. This depends on the network operator decision.

3. Evaluation Methodology

We have developed a Java-based software to evaluate the performance of E-LFAs. Throughout our simulations, we use the original LFAs and best possible paths as benchmarks to ensure an unbiased comparison. It is important to note that, the best possible paths used for comparison are based on the corresponding scheme (*i.e.* avoiding links or avoiding nodes). The repair coverage of the algorithm and the stretch required for routing via alternate next hops are used as evaluation metrics. In addition, we estimate the number of bits required in the header to permit forwarding under our routing technique.

We run our simulations on different types of topologies to show that our algorithm can perform better than LFAs in an arbitrary network. This includes Abilene [14] and GEANT [15] which are real topologies available to the public. In addition, we use the inferred backbone topologies of Abovenet, Sprintlink, and Tiscali provided by the Rocketfuel [16] and synthetic topologies generated by BRITE

[17] based on different models. Our evaluation is separated into two cases: *a)* with LFC and *b)* with NPC. Following this section, we present and analyse the simulation results.

4. Results

The first important result is the repair coverage of the protection scheme. Table 1 shows the percentage of destinations being protected under each resilient technique in relative to a 100% of recoverable destinations. We define that a destination is recoverable if it can be reached after a link or node failure upon completion of network re-convergence.

Table 1: Repair coverage of different topologies under LFAs and E-LFAs

Topology	Degree	Link Protection		Node Protection	
		LFAs (%)	E-LFAs (%)	LFAs (%)	E-LFAs (%)
Abilene	1.273	65.455	89.091	58.537	85.366
GEANT	1.609	91.107	99.209	81.250	88.889
Abovenet	2.696	97.549	99.558	81.884	94.823
Sprintlink	3.086	96.242	98.815	77.733	95.417
Tiscali	2.037	88.063	97.205	78.792	93.391
Waxman	2.000	91.404	99.677	87.432	97.989
BA	1.970	92.707	99.869	81.201	95.361
BA-2	3.830	98.802	99.969	95.917	99.886

It can be clearly seen that the repair coverage of LFAs has been improved in all topologies regardless of protection condition. In average, 7.758% more links and 13.547% more nodes can be protected with E-LFAs. It is important to note that, a router still employs paths via LFAs if they exist. Performing the recursive method repeatedly can further increase the repair coverage in certain cases. However, the difference in performance is negligible while it requires more memory and increases the complexity of the forwarding plane.

By averaging the stretch across all topologies, we found that the average stretch of all paths employing E-LFAs is less than 1.257 for link protection and 1.252 for node protection which are only 0.02 and 0.04 higher than the optimal paths. Note that, we only calculate the stretch of node pairs with existing LFAs or E-LFAs. More importantly, routing packets through a longer path is often encouraged in IP traffic engineering to avoid congestion.

E-LFAs are calculated using the same conditions as normal LFAs; therefore, it guarantees a loop-free environment. By using an extra bit to indicate a re-routed packet, our routing scheme does not create a forwarding loop even if there are multiple failures. The simulation results show that more than 99.586% of E-LFAs require only 1 bit and at most 3 bits for link protection, and more than 99.943% of E-LFAs require only 2 bits and at most 3 bits for node protection. Thus, we require 4 bits in the packet header to permit re-routing via E-LFAs. We propose to use parts of ToS field for IPv4 and parts of Traffic Class field for IPv6.

5. Conclusion

The major cause of network reliability problems is the lack of resilient mechanism for handling failures. This is intolerable for emerging services and applications due to their sensitive nature. When a failure occurs, packets are being dropped continuously until the re-convergence process completes or the routing becomes consistent for affected destinations.

This paper proposed a technique known as Enhanced Loop-Free Alternates E-LFAs) to elevate the performance of the original LFAs without jeopardising the simplicity of traditional IP routing. Our solution makes use of a simple recursive method on the normal LFAs without breaching the conditions used in alternate next hops finding. The simulation results showed that by employing E-LFAs the repair coverage can be improved significantly in an arbitrary topology. We believe that E-LFAs are good alternatives for network operators who want to elevate the network reliability without sacrificing the performance of a router or increase the complexity in the management plane.

References

[1] M. Motiwala, M. Elmore, N. Feamster, and S. Vempala, "Path Splicing," in *Proc. ACM SIGCOMM*, Seattle, WA, Aug 2008, pp. 27-38.

[2] W. Xu and J. Rexford, "MIRO: Multi-path Interdomain ROuting," in *Proc. ACM SIGCOMM*, Pisa, Italy, Sep 2006, pp. 171-182.

[3] X. Yang and D. Wetherall, "Source Selectable Path Diversity via Routing Deflections," in *Proc.ACM SIGCOMM*, Pisa, Italy, Sep 2006, pp. 159-170.

[4] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris, "Resilient Overlay Network," in *Proc. ACM SOSP*, Banff, Canada, Oct 2001, pp. 131-145.

[5] M. Shand and S. Bryant. (2009, Feb) IP Fast Reroute Framework. IETF Internet draft. [Online]. Available: http://tools.ietf.org/html/draft-ietf-rtgwg-ipfrr-framework-10.

[6] M. R. Goyal and K. K. W. Feng, "Achieving Faster Failure Detection in OSPF Networks," in *Proc. IEEE ICC*, Anchorage, AK, May 2003, pp. 296-300.

[7] G. Iannaccone, C.-N. Chuah, R. Mortier, S. Bhattacharyya, and C. Diot, "Analysis of Link Failures in an IP Backbone," in *Proc. ACM IMW*, Marseille, France, Nov 2002, pp. 237-242.

[8] A. Markopulu, G. Iannaccone, S. Bhattacharya, C.-N. Chuah, and C. Diot, "Characterization of Failures in an IP Backbone," in *Proc. IEEE INFOCOM*, Hong Kong, Mar 2004, pp. 2307-2317.

[9] S. Nelakuditi, S. Lee, Yi Yu, Z.-L. Zhang, and C.-N. Chuah, "Fast Local Rerouting for Handling Transient Link Failures," *IEEE/ACM Transactions on Networking*, vol. 15, no. 2, pp.359-372, 2007.

[10] A. Atlas and A. Zinin. (2008, Sep) Basic Specification for IP Fast Reroute Loop-Free Alternates. RFC 5286. [Online]. Available: http://tools.ietf.org/html/rfc5286.

[11] S. Bryant, M. Shand, and S. Previdi. (2010, Mar). IP Fast Reroute Using Not-Via Addresses. IETF Internet Draft. [Online]. Available: http://tools.ietf.org/html/draft-iet-rtgwg-ipfrr-notvia-addresses-05.

[12] A. Atlas. (2006, Feb) U-turn Alternates for IP/LDP Fast Reroute. IETF Internet Draft. [Online]. Available: http://tools.ietf.org/html/draft-atlas-ip-local-protect-uturn-03.

[13] S. Bryant, C. Filsfils, S. Previdi, and M. Shand. (2007, Nov) IP Fast Reroute Using Tunnels. IETF Internet Draft. [Online]. Available: http://tools.ietf.org/html/draft-bryant-ipfrr-tunnels-03.

[14] Y. Zhang. (2004, Dec) The Abilene Topology and Traffic Mattrices. [Online]. Available: http://www.cs.utxas.edu/~yzhang/research/AbileneTM/.

[15] GEANT. (2004, Dec) The GEANT Topology. [Online].

Available: http://www.geant.net/upload/pdf/GEANT_Topology_12-2004.pdf.

[16] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, "Measuring ISP Topologies with Rocketfuel," *IEEE/ACM Transactions on Networking*, vol. 12, no. 1, pp. 2-16, 2004.

[17] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An Approach to Universal Topology Generation," in *Proc. IEEE MASCOTS*, Cincinnati, OH, Aug 2001, pp. 346-353.