Optimizing IGP Link Costs for Improving IP-level Resilience

Gábor Rétvári, Levente Csikor, János Tapolcai

High Speed Networks Laboratory

Department of Telecommunications and Media Informatics

Budapest University of Technology and Economics

Email: {retvari, csikor, tapolcai}@tmit.bme.hu

Gábor Enyedi, András Császár

TrafficLab

Ericsson Telecommunications Hungary

Email: {gabor.sandor.enyedi,andras.csaszar}@ericsson.com

Backgrounds

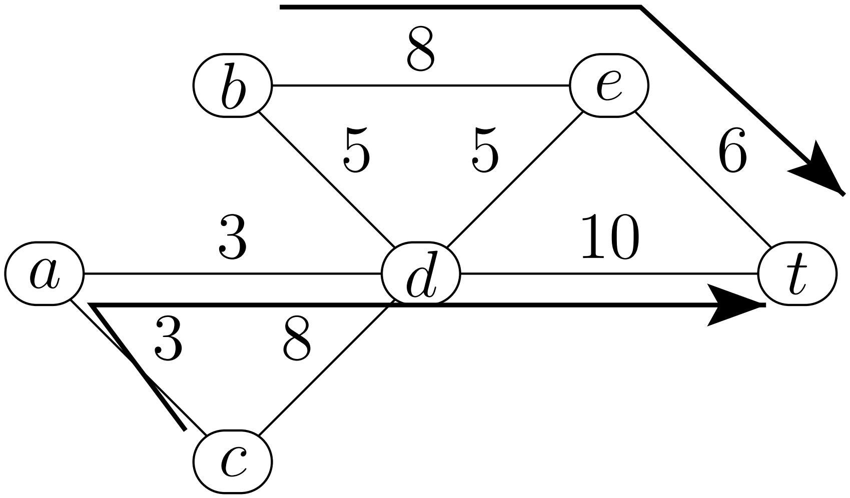
- Many operators provide commercial telecom services over pure IP
- Legacy IP failure recovery is slow (>150 ms)
- For <50 ms resilience, IP-level protection is the way to go
- There is only one IP fast-resilience scheme available in off-the-shelf routers: Loop Free Alternates (LFA)
- But with LFA certain failure cases are impossible to repair
- Can we improve?
- **Not** by changing LFA!

IP Fast ReRoute

- A framework for fast protection implemented in pure IP
  - instant failure detection (e.g., BFD, layer 2)
  - switch to precomputed detours
  - locally route around the failure
  - then get packet back to shortest path
  - let the IGP converge in the background
  - recompute detours
- Benefits both pure IP and MPLS-LDP

Basic IPFRR: Loop Free Alternates

- Piggy-back IPFRR on a standard link-state IP shortest path routing protocol (OSPF, IS-IS)
- When next-hop goes away, pass packet on to a neighbor that still has an intact route to the destination
- Basically any neighbor that will not send it back
- Enough to ensure that the alternate is not upstream
- So it will not loop the packet back



Alternatives of LFA

- IPFRR is hard: destination-based forwarding does not play well with local rerouting
- For full protection, packets on detour must be distinguished from packets on default paths
- Alter destination-based forwarding (FIR & co.)

  S. Nelakuditi et al. „Fast local rerouting for handling transient link failures", INFOCOM'04.

  - consider packet's incoming interface in forwarding
  - full protection, but per-interface FIB is not supported
- Explicit failure signaling (e.g., remote LFAPs)

  I. Hokelek et al.„Loop-free IP Fast Reroute using local and remote LFAPs" Internet Draft, Feb 2008.

  - standalone signaling mechanism for IPFRR
  - operators reluctant to deploy

Alternatives of LFA

- In-band signaling (MRC, SafeGuard, IP redundant trees)

  A. Kvalbein et al. „Fast IP Network Recovery Using Multiple Routing Configurations”, INFOCOM’06.

  - e.g., mark detours in the IP header
  - could never be pushed through IETF
- Tunneling (near-side/far-side tunneling, Not-via)

  S. Bryant et al. „IP fast reroute using Not-via addresses”, Internet Draft, March 2007.

  - „lightweight in-band signaling”: mark packets in destination address
  - wire-speed tunneling not reachable everywhere
  - MTU issues can cause debug nightmare
- Various combinations

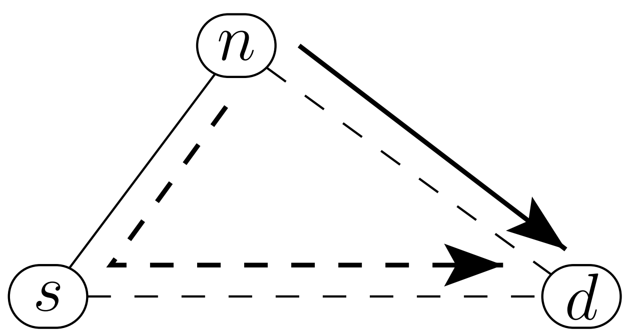
  M. Menth et al. „Loop-free alternates and not-via addresses: A proper combination for IP fast reroute?”, Comput. Netw., 54/8 pp. 1300–1315, 2010.

Alternatives LFA

- Alternatives are too complex
  - extra-management burden, added complexity and non-trivial infrastructure upgrade: deployment barrier
- In contrast, LFA is unobtrusive and incrementally deployable
  - standardized and commercially available
  - Cisco IOS Release 3.7, JUNOS 9.6
  - remains the only IPFRR technique widely implemented
  - but it does not provide complete protection!
- Before deployment of LFA, some questions must be answered
  1. To what extent LFA can protect real networks?
  2. Which topologies are good for LFA, and which are bad?
  3. If LFA turns out inefficient in a particular case, how can we improve?

Link-protecting LFAs: some definitions

- p2p links, no LANs, no ECMP, no SRLGs, only link failures
- Some neighbor n of s is a **link-protecting LFA** for s to d if
  (i) n is not the default (shortest-path) next-hop of s to d
  (ii) $\mathrm{dist}(n, d) < \mathrm{dist}(n, s) + \mathrm{dist}(s, d)$



- **LFA coverage metric** $\eta(G, c)$: characterize networks based on their amenability to LFA

$$\eta(G, c) = \frac{\text{\#LFA protected } (s, d) \text{ pairs}}{\text{\#all } (s, d) \text{ pairs}}$$

Graph theoretical LFA coverage analysis

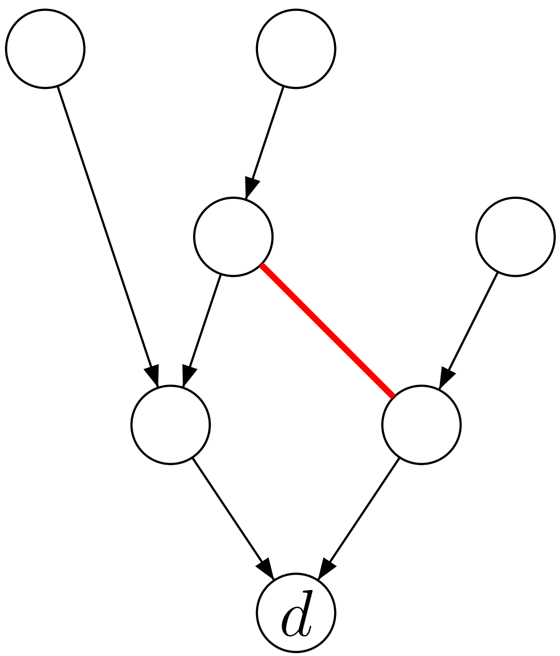
- **Theorem:** for any connected simple graph G with $n > 2$,

$$\eta(G,c) \leq \frac{n}{n-1}(\Delta - 2) + \frac{2}{n-1}$$

  where Δ is the average node degree
  - non-trivial for $\frac{2(n-1)}{n} \leq \Delta < 3$
  - tight for trees and uniform cost odd rings

Graph theoretical LFA coverage analysis

- The shortest path tree of any d contains $n-1$ edges
- The remaining $m-n+1$ edges provide at most 2 LFAs per node
- At most $n(\Delta-2)+2$ nodes have LFA to d



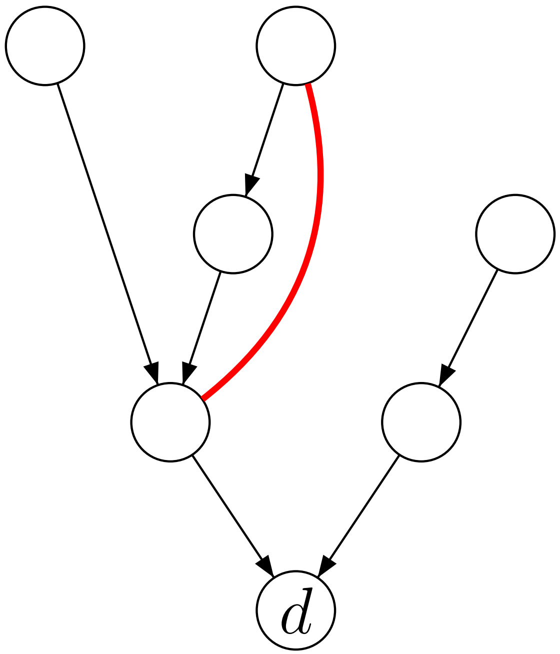
Graph theoretical LFA coverage analysis

- **Theorem:** for any connected simple graph G with $n > 2$,

$$\eta(G,c) \geq \frac{n}{n-1}\frac{\frac{\Delta}{2}-1}{\Delta_{\max}-1}+\frac{1}{(n-1)(\Delta_{\max}-1)}$$

where $\Delta_{\max}$ is the maximum node degree

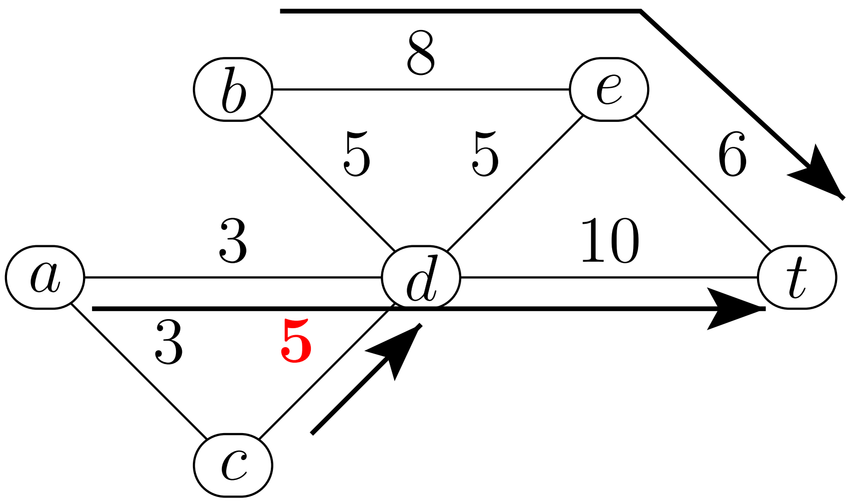
- A non-shortest-path edge provides at least one LFA to d
- When edge is inside a branch



What if some nodes do not have LFA?

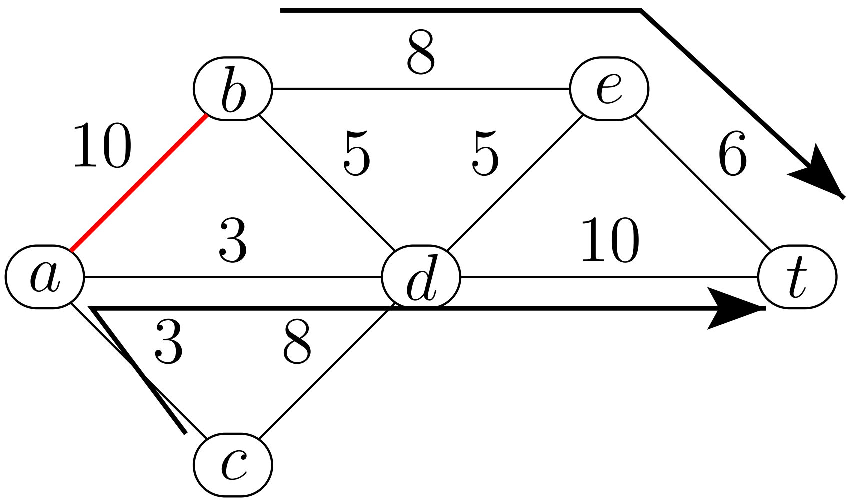
1.) Change link costs

- cheap but alters shortest paths
- might be too much of a price for improved LFA coverage



2.) Alter the topology by adding new links

- can be costly
- but leaves shortest paths intact
- at least, if new links are of sufficiently high cost



LFA cost optimization

- Find the link costs that maximize LFA coverage
- **Def.:** given a graph $G(V, E)$, is there a cost setting c so that $\eta(G, c) = 1$?
- Earlier papers treated the combined the cost optimization & traffic engineering problems

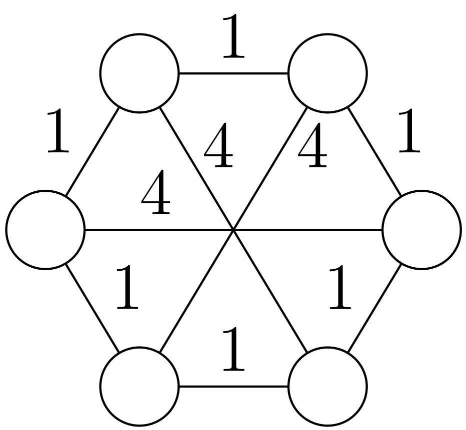
  M. Menth, M. Hartmann, and D. Hock, „Routing optimization with IP Fast Reroute,” Internet Draft, July 2010.

  H. T. Viet, P. Francois, Y. Deville, and O. Bonaventure, „Implementation of a traffic engineering technique that preserves IP Fast Reroute in COMET,” in *Algotel*, 2009.

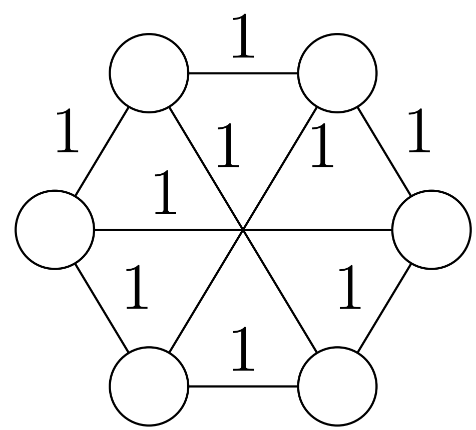
- We treat LFA cost optimization separately
  - not everyone is interested in TE
  - determine individual computational complexity
  - tells to what extent LFA can protect a network and how much we can improve

LFA cost optimization: potential

- A careful selection of costs can increase LFA coverage by more than 50%



$$\eta(G, c) = 1$$



$$\eta(G, c) = \frac{4}{9} < \frac{1}{2}$$

Complexity and algorithms

- **Theorem:** LFA cost optimization is NP-complete
- For a single destination, equivalent to the protection routing problem

  K.-W. Kwong, L. Gao, R. Guerin, and Z.-L. Zhang, „On the feasibility and efficacy of protection routing in IP networks," in *INFOCOM*, 2010.

- Gave an ILP and a simulated annealing-based heuristics
  - start with a random cost setting c and initialize temperature T to a high value
  - in each step, choose the „neighboring" cost setting that maximizes LFA coverage
  - neighboring means that differs at only one edge with 1
  - accept if LFA coverage improves
  - also accept if reduces, with probability decreasing with T
  - decrease T

Numerical results

- Ran the ILP and the heuristics on artificial and real topologies
- Real link costs wherever available, random costs otherwise
- The ILP can only be solved in small networks ($n \leq 8$)
- For such networks, the heuristics finds the optimum in 18 out of 20 cases
- For larger topologies, 5-20% improvement depending on the density

Name	n	m	Δ	$\eta(G, c)$	$\eta(G, c^*)$
AS1221	7	9	2.57	0.809	0.833
AS1755	18	33	3.66	0.872	0.98
AS3967	21	36	3.42	0.785	0.967
M_{30}	30	45	3	0.482	0.89

Conclusions

- IPFRR is under wide-scale deployment
  - LFA is the only commercially implemented technique
  - simple, but no protection for all failure scenarios
- In this paper: theoretical and practical studies on how to actually deploy LFA
  - general bounds on LFA coverage
  - introduced the LFA cost optimization problem
  - computationally hard, but efficiently approximable
  - significant improvement in real topologies