

Flow optimization in IP networks with fast proactive recovery

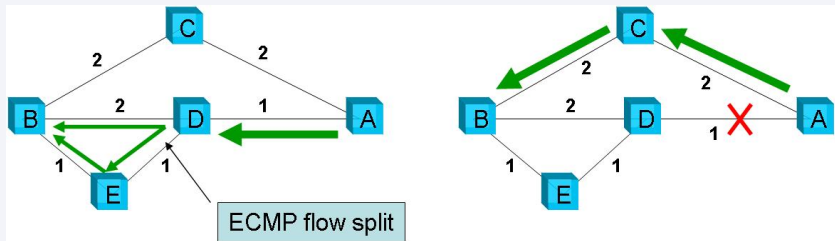
M. Dzida¹, M. Zagożdżon¹, M. Żotkiewicz¹, M. Pióro^{1,2}

¹Warsaw University of Technology, Poland

²Lund University, Sweden

Networks, 28 Sep - 2 Oct, 2008, Budapest





- shortest path routing (SPR)
- SPR protocols require seconds to reconverge after a failure
 - not suited for network applications requiring fast post-failure restoration
 - transient loops can cause packet losses

- a restoration mechanism with faster convergence is required
- a candidate: fast (local) rerouting (IPFR) [1]:
 - Equal-Cost Multiple Paths (ECMP)
 - Loop Free Alternate (LFA)
 - Multi-Hop Repair Path (MHRP)

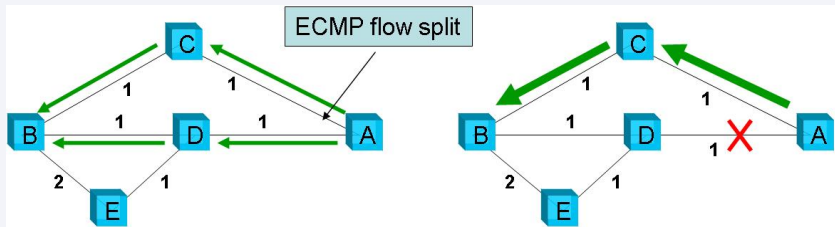


[1] M. Shand and S. Bryant.

IP Fast Reroute framework.

Technical Report, 2007, Internet Draft, work in progress.

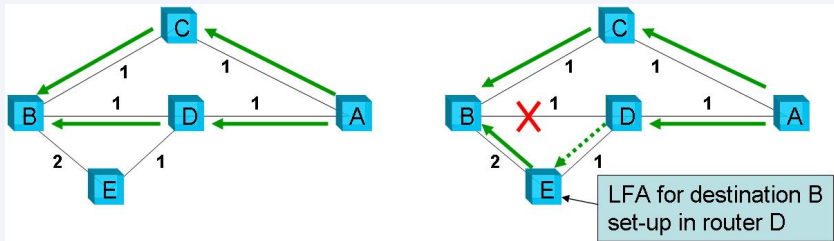
- ECMP split
- traffic overflow on the surviving shortest paths



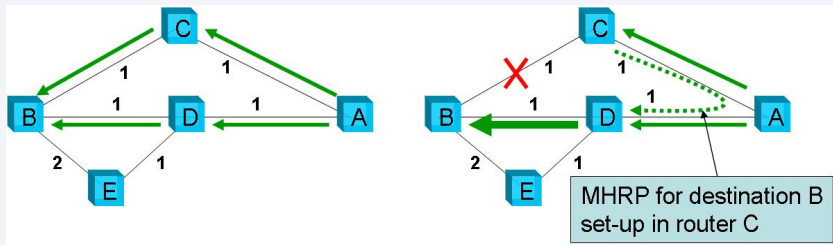
Loop-free alternate (LFA)



- no ECMP split
- neighbor routers proved not to form loops can be used as next-hop



- in 80% cases, ECMP and LFA are enough
- remaining 20% require ≥ 2 -hop loop-less paths, e.g., MPLS tunnels





- to formulate an optimization problem related to IPFR
- to develop an efficient resolution method
- to validate the method with respect to “rule of thumb” approaches

Problem

For given:

- network graph
- set of traffic demands
- link capacities

find

- an IGP weight system and the corresponding ECMP flows
- a set of LFA routers
- a set of MHRP paths

such that

- link loads are minimized
- the number of MHRP paths is minimized



- the problem is hard in the integer programming sense
- the direct approach based on resolving a related MIP is inefficient
- the proposed iterative (partially heuristic) method seems to be a reasonable approach

- Step 1:** compute an IGP link weight system and ECMP flows
- MIP maximizing # ECMP splits
- Step 2:** compute a set of LFA nodes for the links not protected by ECMP
- MIP identifying all feasible LFA nodes
- Step 3:** compute a set of MHRP paths for the remaining unprotected links
- MIP identifying feasible MHRP paths of minimal total length
- Step 4:** go back to Step 1 and try to improve the link weight system

- local search to find a feasible IGP system
- for each system solve Step 2 and Step 3
- evaluate the obtained systems with respect to the following objective function:
 - a) minimize the maximal link overload in the first place
 - b) if no overloaded link occurs, then use the minimal number of MHRP paths
 - c) the used MHRP paths are supposed to be as short as possible
 - d) finally, minimize the maximal link utilization (residual capacity)

- 3 versions of the first step: local search, uniform weights, and weights proportional to the inverse link capacity

Net	local search			$w_e = 1$			$w_e \sim \frac{1}{c_e}$		
	z^*	#	L	z^*	#	L	z^*	#	L
polska (12)	2.49	14	43	2.74	26	80	3.37	29	81
cost293 (11)	0.71	0	0	1.08	2	4	1.08	2	4
geant (19)	0.93	46	98	0.82	124	248	0.92	75	152
labnet (21)	0.99	1	2	0.95	10	20	0.95	10	20

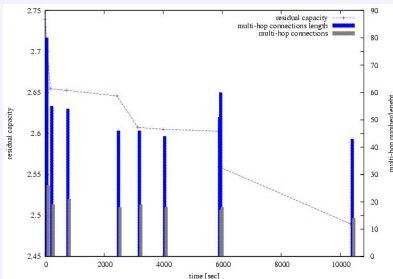
z^* maximal utilization level

number of MHRP paths used

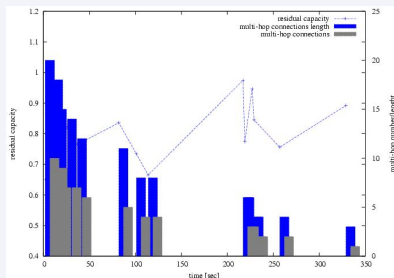
L total length of MHRP paths

Nets: <http://sndlib.zib.de>

- Three step heuristic algorithm iterations:



network_12 (12 nodes, 36 links)



labnet (21 nodes, 106 links)

- the main purpose of the optimization model is to exhibit the efficiency of IPFR
- an optimization model supporting IPFR
- an iterative approach based on heuristic partitioning of the problem into three phases
- each phase solved by the state-of-the-art method
- time-effective
- reasonable solutions
- in many cases we are able to eliminate almost all (disadvantageous) MHRP paths



- How to make the IPFR solution consistent with the stable SPR flows (after reconvergence)?
 - then we could stop using IPFR after a certain timeout
- How to adjust IPFR for multiple failures?
- Find a way to validate the quality of the solutions (to be able to estimate the optimality gap).