Routing in Congested Networks: [Shaikh00a]

(got to slide 19 on Feb. 2)

CSci551: Computer Networks
SP2006 Thursday Section

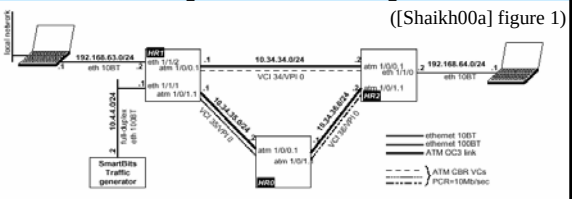
John Heidemann

Key Idea

- experiments:
  - set up toy network
  - inject traffic
  - causes congestion and lost routing messages
  - this causes routing to think the link is down
- do some analysis and modeling of protocol behavior
- (possible) conclusion: should isolate routing from data traffic?
  - we consider control traffic more important than data traffic

Experiment Setup

([Shaikh00a] figure 1)



- traffic generator
- loads from 125-500% of capacity

- link buffer 4 or 16MB
- drop-from-front queueing
- packet-aware ATM links

Topologies

2- vs. 3-nodes: how problem resolves differs

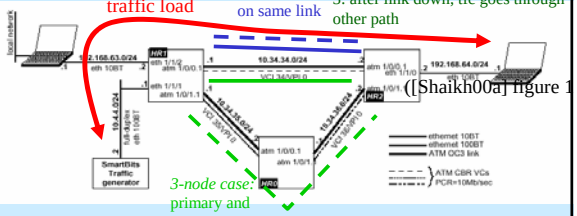
2 node case: stop sending new tfc after link is declared dead

3: after link down, tfc goes through other path

2-node case: primary and backup on same link

traffic load

([Shaikh00a] figure 1)



3-node case: primary and backup on different links

Measure

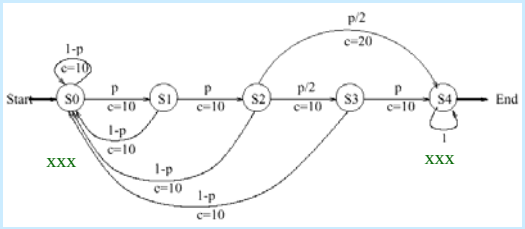
- u2d and d2u
  - route failure and restore times
- for OSPF and BGP
- as load goes from 125-500% of capacity (drop prob *p* from .2-.8)

(and *model* what you expect to check understanding)

Failure Situations

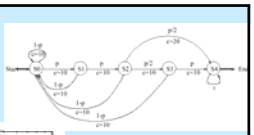
- OSPF
  - HELLO sent every 10s over UDP
  - expect one every 40s or link down
- BGP
  - sends keepalives over its TCP connection every 60s
  - expect at least one every 180s or link down

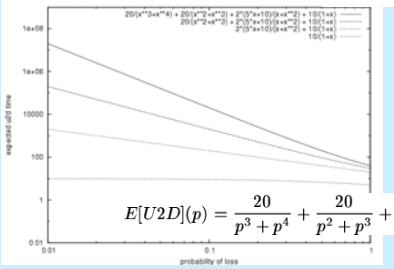
Markov Model of OSFP Failure



For details about how to solve Markov models see EE465.

Solving the OSPF Markov Model

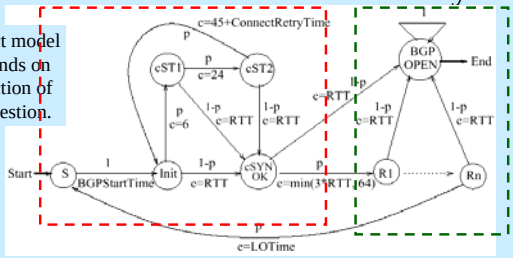




$$E[U2D](p) = \frac{20}{p^3+p^4} + \frac{20}{p^2+p^3} + \frac{2(5p+10)}{p+p^2} + \frac{10}{1+p}$$

Markov Model of BGP Recovery

Exact model depends on direction of congestion.



Routing Performance

- we saw the models,
- how do the protocols affect routing performance?
  - UDP connectionless, so no set up time => don't have to go through TCP setup time
  - u2d(BGP) > u2d (OSPF)?
    - at 100% load, u2d(BGP) = 1000s, while u2d(OSPF) = 200s
    - OSPF and UDP… send pkt, lost or not; try again at next hello message (in 10s)
    - BGP and TCP… send pkt, if lost, TCP tries to resend after RTO
      - in effect, TCP resending the keepalive message gives you "extra chances"

Experimental Results

- OSPF model is right on
  - 2-node: Figures 8 and 9
  - 3-node: Figures 10 and 11
  - **notice short absolute fail times**
- BGP results
  - 2-node: Figures 13 and 14
  - **much longer fail times** *(why?)*

Other Comments?

- xxx