eecs 589

Advanced Computer Networks

Khanna and Zinky, "The Revised ARPANET Routing Metric," *Proc. of ACM SIGCOMM '89*, 19(4):45–46, Sep. 1989

Routing Metric

Distributed route computation is a function of link cost (routing metric)

Link costs are propagated hop by hop throughout the network

Problem: how to determine the cost of a link?
- static: hop count, link capacity, reliability
- dynamic: current load

Consequences: in determining routing, link cost also determines load distribution on the network

ARPANET Routing Algorithms

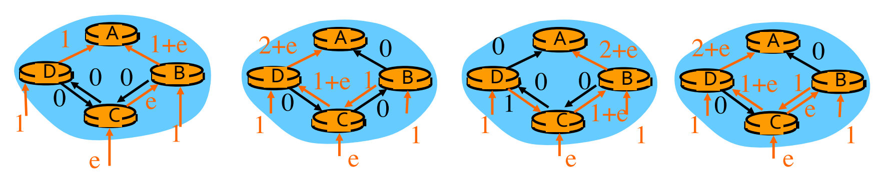
1969: GGP (Bellman-Ford, distance-vector)
- routing loops
- instantaneous queue length used as link metric

5/79: Delay-Shortest Path First (D-SPF)
- link cost is avg. delay over 10 secs.

Problem with D-SPF

Computing link cost as a function of link load leads to routing instability (oscillation) under heavy load

Example: traffic destined for A



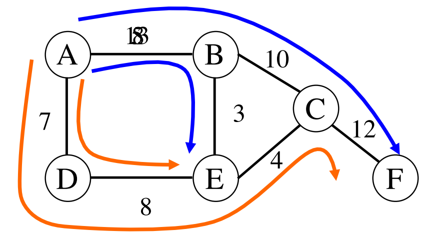
initially | recompute routing | recompute | recompute

Routing Instability

Caused by three "design" choices:

1. delay range is too wide
   ⇒ some links can look so bad
   that they are completely avoided
2. delay variation in successive updates are not limited
   ⇒ frequent link cost changes led to route oscillation
3. all nodes adjust routes simultaneously
   ⇒ synchronized route oscillation

Route Oscillation



D-SPF: AB congested link

Path	delay (t_1)	delay (t_2)	delay (t_3)
AB	5	13	8
ABE	8	16	11
ADE	15	15	15
ABCF	27	35	30
ADECF	31	31	31

Why is Routing Instability Bad?

Leads to unused bandwidth

Causes congestion to spread (to longer paths using the congested link)

Route flapping causes packet reordering

Extra routing traffic wastes bandwidth

Extra route updates wastes router CPU cycles

7/87: Hop-Normalized SPF

Range too wide: delay normalized to hop count (delay/delay at congested link), e.g.,
link delay $= 30$ ms ≈ 1 hop ⇒ delay of 60 ms ≈ 2 hops

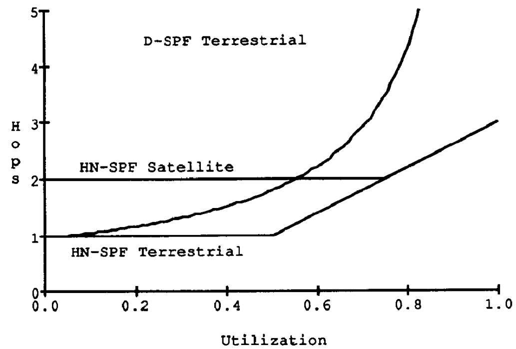
Delay variation not limited:

- averaged over the last 2 routing periods
- changes in half hops (hysteresis), e.g., delay of 30, 31, 32 are all considered 1 hop, 60 and 62 are both 2 hops
- max metric difference: 2 hops ⇒ max link cost is 3 hops

7/87: Hop-Normalized SPF

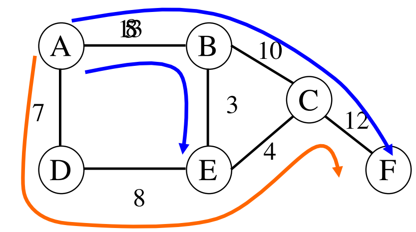
New link starts at max hop

After equilibrium, link cost kept at a preset value until utilization reaches a threshold (why?), then it is allowed to increase linearly as a function of link bandwidth



No Route Oscillation

Benefit: routes with slightly longer paths are shed from congested links first



HN-SPF: AB congested link

Link	delay [hop] (t_1)	delay [hop] (t_2)	delay [hop] (t_3)
AB	5 [1]	13 [3]	8 [2]
ABE	8 [2]	16 [3]	11 [2]
ADE	15 [3]	15 [3]	15 [3]
ABCF	27 [5]	35 [7]	30 [6]
ADECF	31 [6]	31 [6]	31 [6]

Study Dynamic Behavior by Simulation

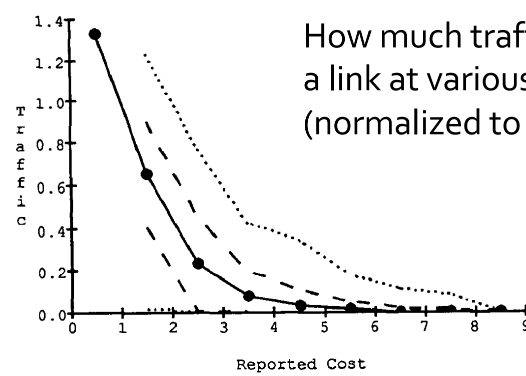
Given:

- actual 7/87 ARPANET topology and
- peak hour traffic matrix: who sends to whom, by how much, for how long

Assume:

- routing is by static cost min-hop SPF
- except for the link under study, whose cost is controlled, all other links report fixed cost

Network Response Map



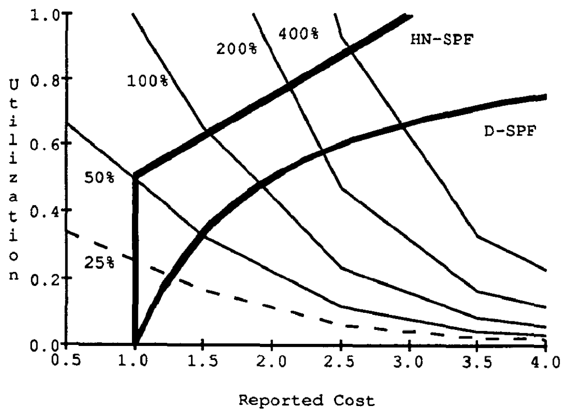
How much traffic is routed through a link at various reported cost (normalized to 1 for cost of 1 hop)

A small change in reported cost causes big change in offered load: compare link utilization at reported cost of 0.5 and 1.5

At reported cost 4, 90% of traffic is shed

Network Response Map

At other traffic load levels–superimposed with D-SPF and HN-SPF link metric reporting curves

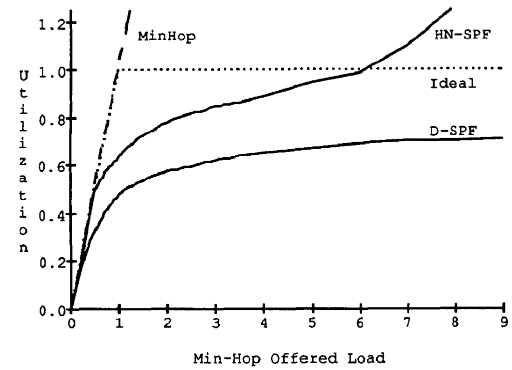


Equilibrium

Traffic level is a function of reported cost

Equilibrium (max utilization) is reached when reported cost corresponding to a certain level of utilization doesn't cause any change in offered load, which would lead to a change in reported cost

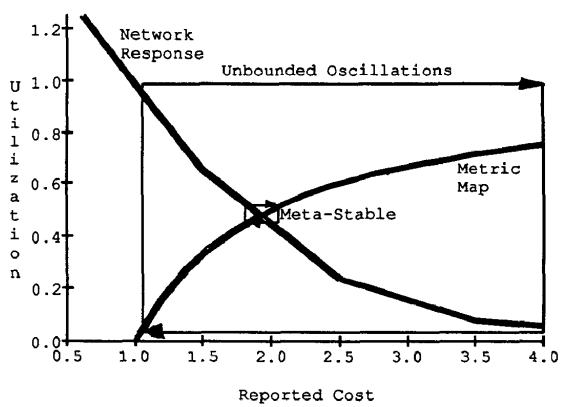
Max utilization for D-SPF is lower than for HN-SPF



D-SPF Dynamic Behavior

D-SPF is unstable and causes a link to oscillate between being oversubscribed and idle

Equilibrium is meta-stable: a slight perturbation can knock the system off its equilibrium into instability



Problem with Delay Metric

Queueing delay stays minimal when traffic is light, so transmission and propagation delays continue to dominate

By reporting current delay and using it to compute routes, assume future delay will stay the same ⇒ not true for heavy traffic

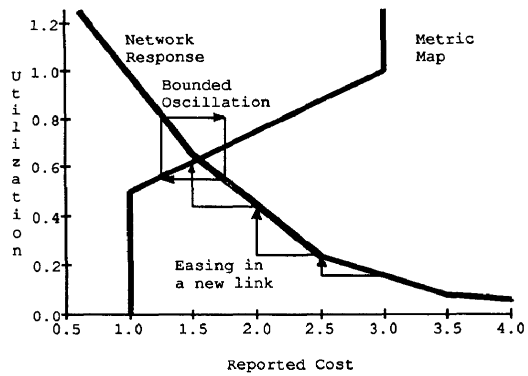
Queueing dominates when traffic becomes heavy

HN-SPF Dynamic Behavior

HN-SPF converges to equilibrium and may oscillate around it with a bounded amplitude

Recall: max change bounded by ½ hop

Averaging cost over the last 2 routing periods:
- slows down the frequency of oscillation
- gently eases in new line



Discussions

On a globally distributed system, local behavior can have unintended consequences

When such arise, is it a "bug"? Or is it (bad) design decision?

Application: Review protocols and design decisions and their implementations to look out for unintended global dynamic behavior

Hop-Normalized SPF

Acts like a delay-based metric under light loads, but like a capacity-based metric under heavy loads

⇒ no shortest-delay guarantee under light traffic, but vastly improved performance under heavy traffic