Slide 0

Performance Evaluation of Quality-of-Service Routing with Inaccurate Link-State Information

Jennifer Rexford
Networking and Distributed Systems
AT&T Labs – Research; Florham Park, NJ
`http://www.research.att.com/~jrex`

Anees Shaikh
Real-Time Computing Laboratory
University of Michigan; Ann Arbor, MI
`http://www.eecs.umich.edu/~ashaikh`

Slide 1

Routing in Integrated Services Networks

- Support the migration of voice, data, and multimedia services to an integrated backbone network
- Quality-of-service routing considers the call's traffic spec and performance requirements in path selection
  - Satisfy the QoS requirements of each call
  - Optimize usage of network resources
- QoS routing consumes bandwidth, CPU, and memory
  - Exchange recent information about link load
  - Compute suitable route for each incoming call

How does the link-state update policy affect this performance vs. overhead trade-off in large networks?

Slide 2

Outline

- Source-directed link-state routing
- Topology and traffic models
- Simulation results of sensitivity study
  - Periodic link-state updates
  - Triggered link-state updates
  - Interactions with network topology
- Conclusions and ongoing work
  - Proposed new routing and signalling policies
  - Extensions to simulation model

Slide 3

Performance Evaluation of QoS Routing

Source-directed link-state routing

- Source computes route based on topology database
- Link-state updates sent periodically or triggered by significant change
- Path selection based on call traffic parameters and available link capacity

Tunable parameters affecting network overheads

- Frequency of link-state update propagation
- Path-selection algorithm and link-cost parameters
- Size of network (number of nodes and links)
- Frequency of route computation (ongoing work)

$\Rightarrow$ sensitivity study of performance vs. overheads

Slide 4

Call Routing and Signalling Model

Route and signal a connection with bandwidth b:

- Source computes a route to the destination node:
  1. Prune infeasible links $(util(l) + b > 1)$
  2. Compute shortest paths to destination node
  3. Choose shortest path with lowest cost $\sum_l util(l)$

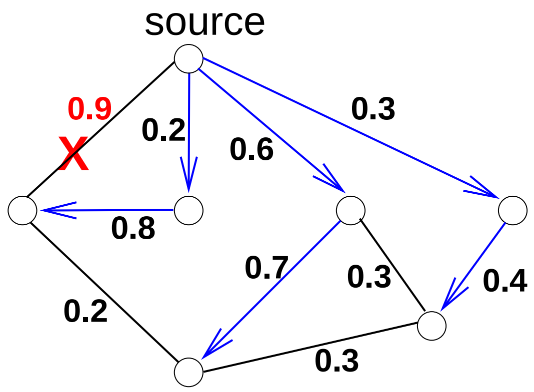
  $\Rightarrow$ cheapest, shortest, feasible path

- Call signalling across each link in the path:
  1. Test admissability $(util(l) + b \leq 1)$
  2. Reserve bandwidth $(util(l) = util(l) + b)$

  $\Rightarrow$ admission control and resource reservation

Slide 5

Routing Failure vs. Set-Up Failure



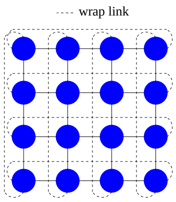
Routing failure: source cannot compute a feasible path

Set-up failure: path cannot reserve the resources

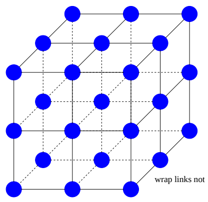
Slide 6

Backbone Topology Model

Homogeneous backbone topology (k-ary n-cube)



4-ary 2-cube



3-ary 3-cube

- Systematic study of network growth (# nodes, # links, #routes, node degree)
- Analytic expressions to validate simulation and predict performance in simple scenarios
- $N = k^n$, $L = 2nk^n$, degree$= 2n$, diam$= \lfloor k/2 \rfloor n$

Slide 7

Synthetic Traffic Model

Standard homogeneous traffic models

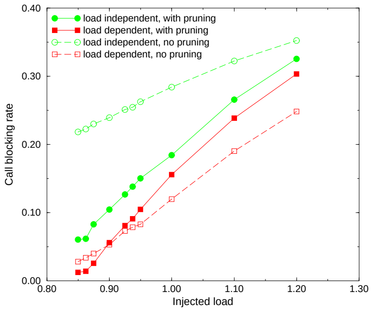
- Characterize basic performance trends
- Understand effects of stochastic traffic variations
- Avoid potential bias against static routing
- Focus on QoS routing in a well-provisioned backbone

Traffic parameters

- Exponential call interarrivals, $1/\lambda \sim \exp(1/\bar{\lambda})$
- Exponential call holding times, $\ell \sim \exp(\bar{\ell})$
- Uniform bandwidth requests, $b \sim U[\bar{b} - s/2, \bar{b} + s/2]$
- Source-destination pairs uniformly distributed

Slide 8

Call Blocking vs. Offered Load

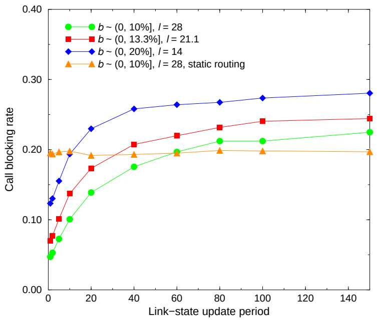


125 nodes, $b_{req} = 5\%$, fixed holding time ℓ, instant updates

- Load-dependent routing reduces call blocking
- Pruning results in some non-minimal routes
- Non-minimal routes help at low load, hurt in overload

Slide 9

Staleness Due to Periodic Updates

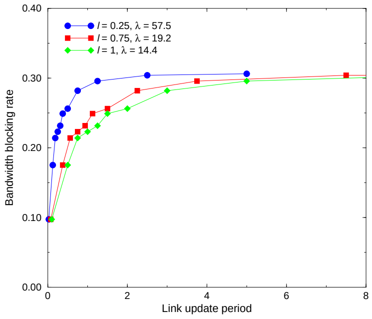


125 nodes, 85% offered load, arrival rate $\lambda = 1$

- More difficult to admit high bandwidth calls
- Low blocking requires very low update period x
- Static routing performs better under large periods

Slide 10

Holding Times and Periodic Updates



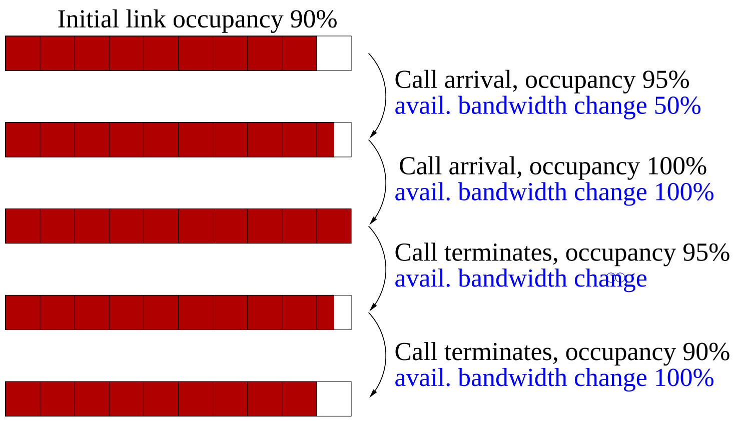
125 nodes, 87% offered load, $b = 10\%$

- Longer holding times permit use of higher periods
- Separate resources for short- and long-lived flows
- Detect long-lived flows and assign to QoS routes

Slide 11

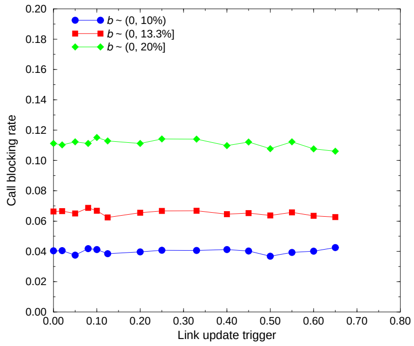
Triggered Link-State Updates

$$bwchange = \frac{|avail_{old} - avail_{new}|}{avail_{old}}$$



Slide 12

Blocking Insensitivity to Triggers

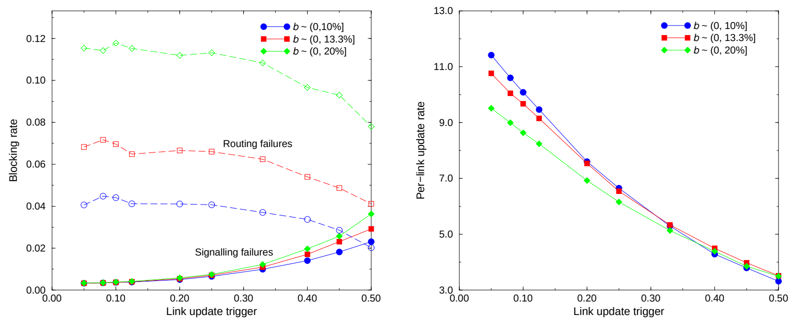


125 nodes, 85% offered load, fixed call arrival rate

- Update state when avail. bandwidth changes by x%
- Link-state staleness does *not* increase call blocking
- Trigger is activated when accuracy is most critical

Slide 13

Set-up Failures vs. Update Overheads



125 nodes, 85% offered load, fixed call arrival rate

- Coarse triggers also decrease link update rate
- Coarse triggers increase set-up failures
- Infeasible links look feasible and vice versa

Slide 14

Infeasible link that looks feasible

- Chosen route cannot support the call on "best" route
- Staleness causes call blocking during set-up
- Accurate information would have pruned this link
- Accurate info would probably result in routing failure

⇒ turns routing failure into a set-up failure

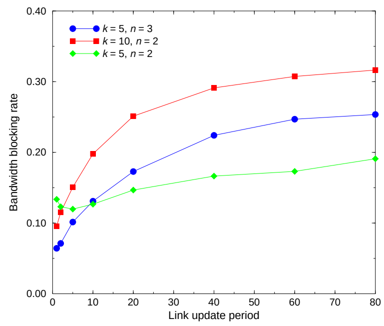
Feasible link that looks infeasible

- Stale information causes inadvertent pruning of link
- Excessive pruning can lead to a routing failure
- Unlikely due to frequent triggers for high-load links
- Unlikely in a topology with multiple short routes

⇒ no significant impact on routing failures

Slide 15

Interaction of Staleness with Topology



$b = 5\%$, 85% offered load, fixed call arrival rate

- Routing failures dominate at lower periods, set-up failures increase with longer periods
- Richer topology helps at low period due to large number of routes, but less helpful under large period

Slide 16

Conclusions and Ongoing Work

Routing and signalling under stale information

- Applying QoS routing only to long-lived calls
- Modest back-off before rerouting blocked calls
- Controlled sequencing through alternate routes
- Precomputed routes with on-demand pruning

Experiments with different network configurations

- Realistic traffic models (telephony, internet)
- Representative backbone topologies (edge-core)
- Range of link cost functions (exponential, discretized)

$\Rightarrow$ efficient link-state routing by controlling processor, bandwidth, and memory resources