Load-Sensitive Routing of Long-Lived IP Flows

Anees Shaikh
IBM Research

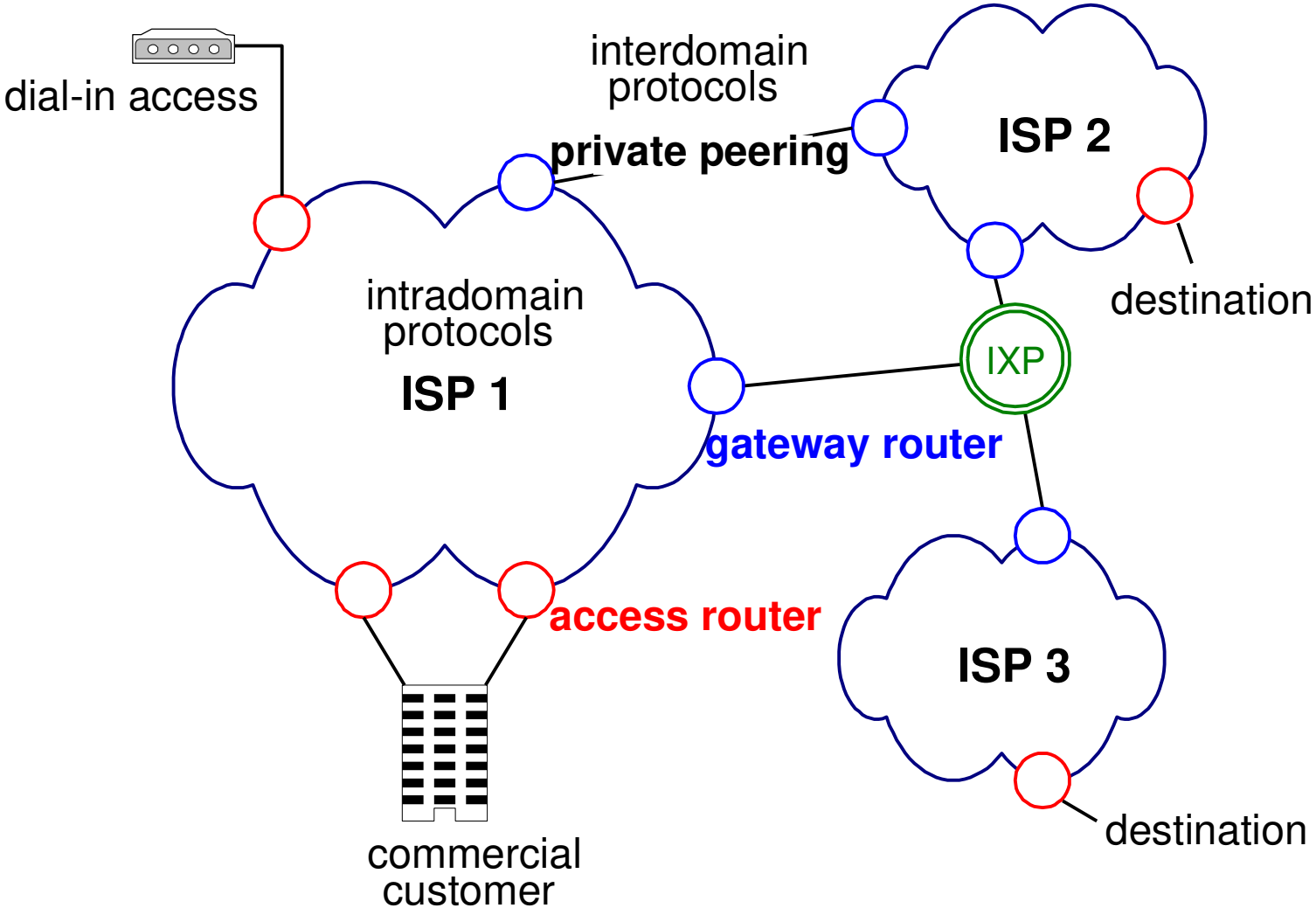
Jennifer Rexford
AT&T Labs--Research

Kang G. Shin
University of Michigan

Main Ideas

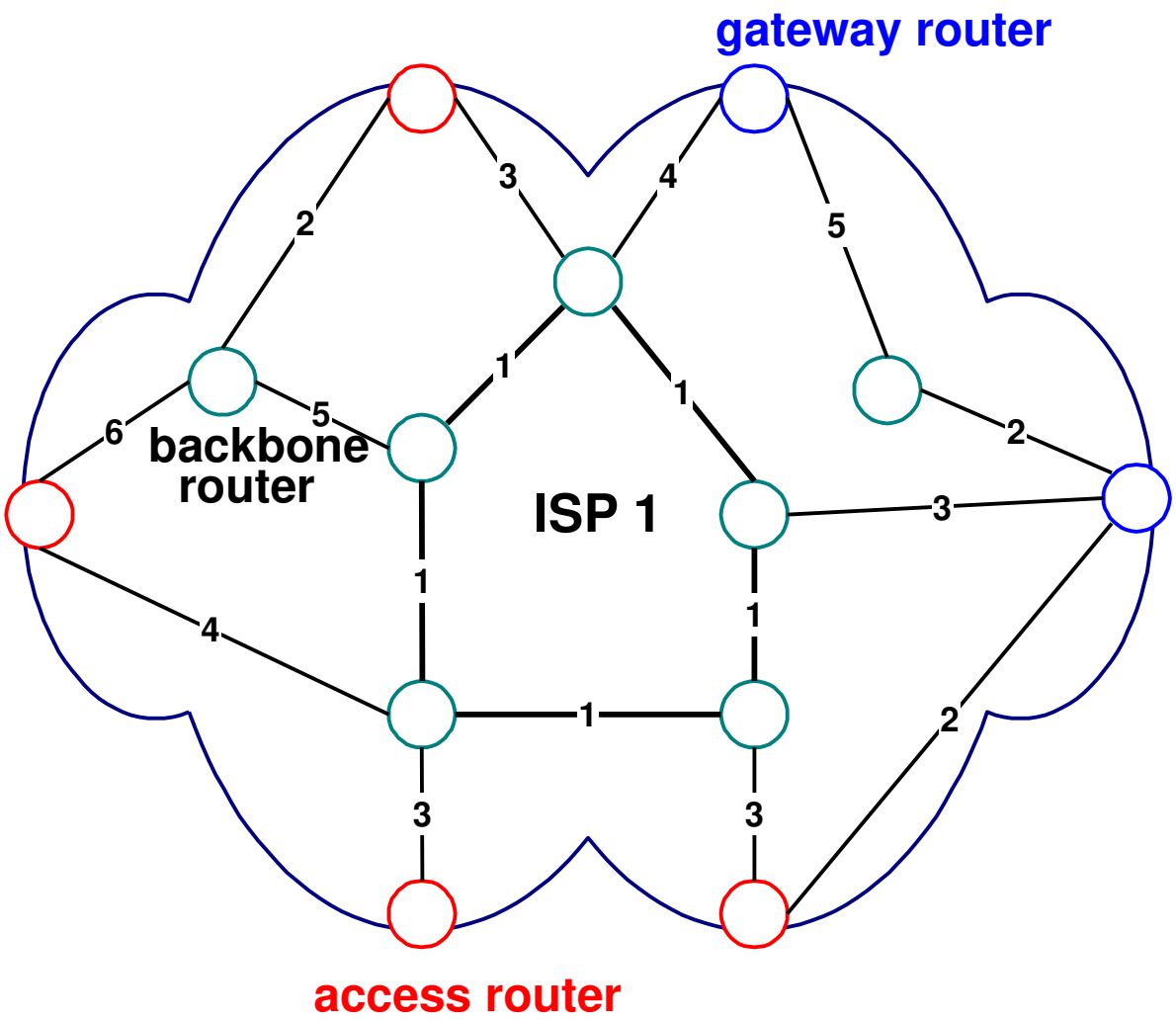
- Load-sensitive routing for traffic engineering
  - adapts routes to changing network conditions
  - suffers from stale load information
- IP flow length distribution
  - many short-lived flows (mice)
  - most volume in long flows (elephants)
- Our proposal
  - static mice and dynamic elephants
  - match flow lengths and link-state updates

Internet Routing Architecture



Routing Inside a Domain (OSPF)

- preconfigured link costs
- exchange cost information
- precompute route tables
- forward to next hop router
- ✓ simple, low overhead



Traffic Engineering Challenges

- Variability of network traffic
  - application variety on the backbone
  - fluctuations in traffic patterns
  - reconfiguration in other ISPs
- Lack of automated techniques
  - problem detection
  - problem diagnosis
  - network reconfiguration

Benefits of Load-Sensitive Routing

- ✓ Quickly react to changes in underlying load
  - collect information about network state
  - circumvent network congestion
  - ➡ optimize network resource usage
- ✓ Provide per-flow quality-of-service
  - choose routes with sufficient resources
  - ➡ improved performance for applications

Source-Directed QoS Routing

- Protocols
  - constraint-based routing in MPLS
  - QoS extensions to OSPF (explicit routing)
  - ATM PNNI protocol
- Operation
  - source router computes route for each flow
  - routes based on current link-state and flow requirements
  - all routers flood link-state updates

Stability Challenges

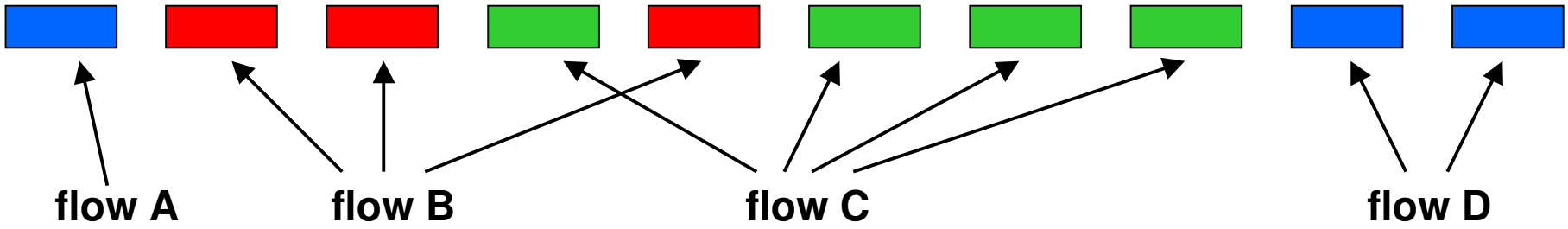
- Link-state staleness
  - route flapping for reasonable update rate
  - high processing and protocol overhead
- Existing techniques to address staleness
  - multi-path routing
  - choose randomly from “near-best” paths
  - probabilistic (safety-aware) routing

➡Ultimately flapping still arises

Elephants Matter

- Bad mice
  - IP transfers are primarily short-lived
  - need frequent updates to “keep up”
- Good elephants
  - IP packets are primarily on long flows
  - residual lifetime is very long
- Possible solutions
  - make link-state update rate higher?
  - make the network smaller (more areas)?
  - make the flows longer-lived?

Background on IP Flows



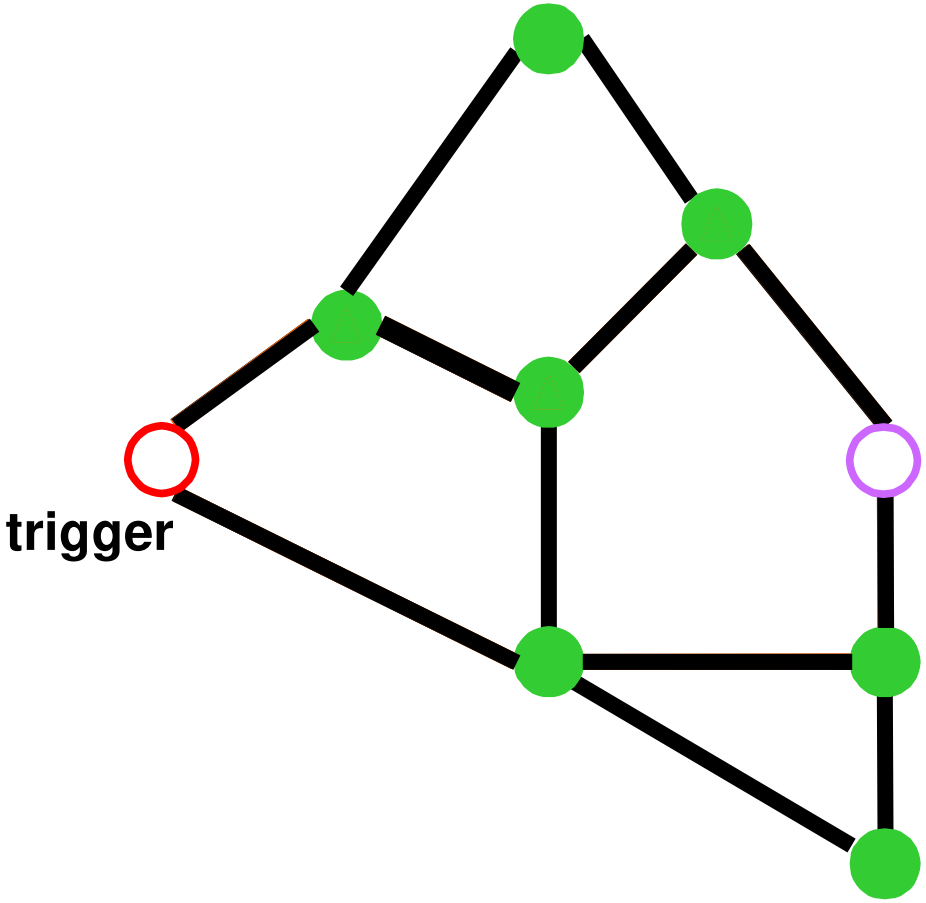
- Group related packets
  - TCP/UDP port, host, subnet, net aggregation
  - based on TOS bits
- Apply a flow timeout to determine end
- Flow requirements may be implicit or explicitly signaled

Hybrid Routing of IP Flows

- Forward packets on static default paths
- Detect long-lived flows at network edge
- Dynamically route long-lived flows

- Reduce overhead and improve stability
  - ✓ lower link-state update frequency
  - ✓ fewer route computations
  - ✓ lower signaling overhead, carry most bytes

Life of a Flow



- forward on default path
- trigger route computation
- pin dynamic path
- forward on new path
- timeout flow and unpin

How to Find Elephants

- Long flows detected using a flow trigger
  - edge routers classify packets into flows
  - trigger may be packets, bytes, or seconds
  - activates dynamic route computation
- Choosing the flow trigger
  - too small: route flapping
  - too large: limited routing flexibility
  - wide range of “good” triggers

Design of Hybrid Routing

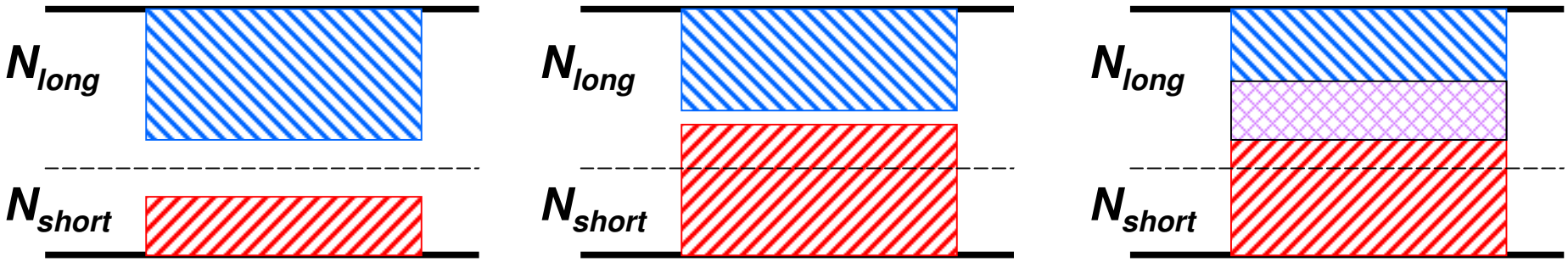
- Link-state based on reserved resources
  - reserved resources change on flow timescale
- Link-state only for long-lived flows
  - no reservation state for short flows
- Allocate resources for short-lived flows
  - trunk reservation to protect short flows
  - allocation based on flow size distribution

Other Design Decisions

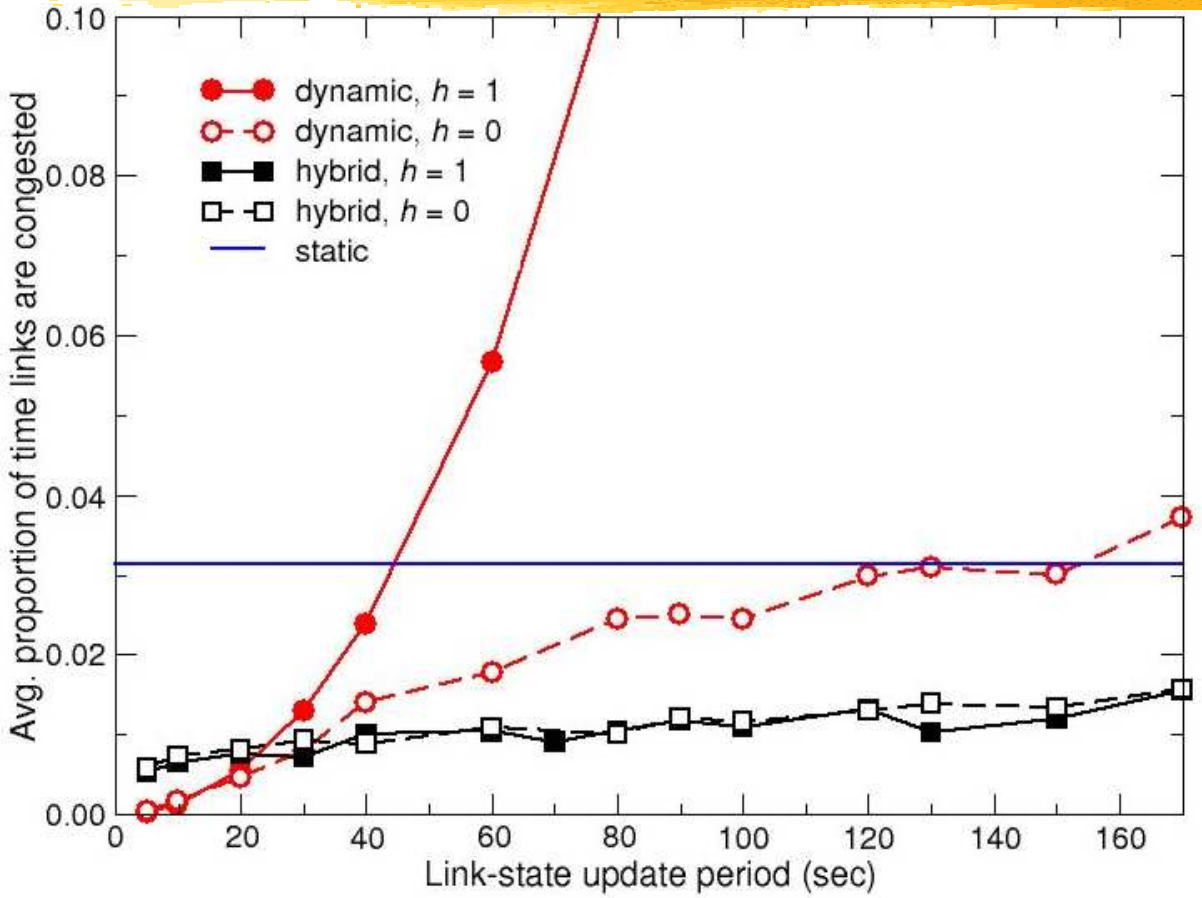
- No blocking
  - better suited to adaptive Internet applications
- “Widest-shortest” routing algorithm
  - shortest: limit resource consumption
  - widest: pick paths with most excess bw
- Resource sharing
  - short flows can consume excess capacity
  - short vs. long ≠ best-effort vs. QoS

Simulation Setup

- Flow-level simulations
  - empirical duration distributions from ISP trace
  - large, well-connected, provisioned topology
  - Poisson arrivals, uniformly dist. bandwidth
- Performance metric: time links are congested

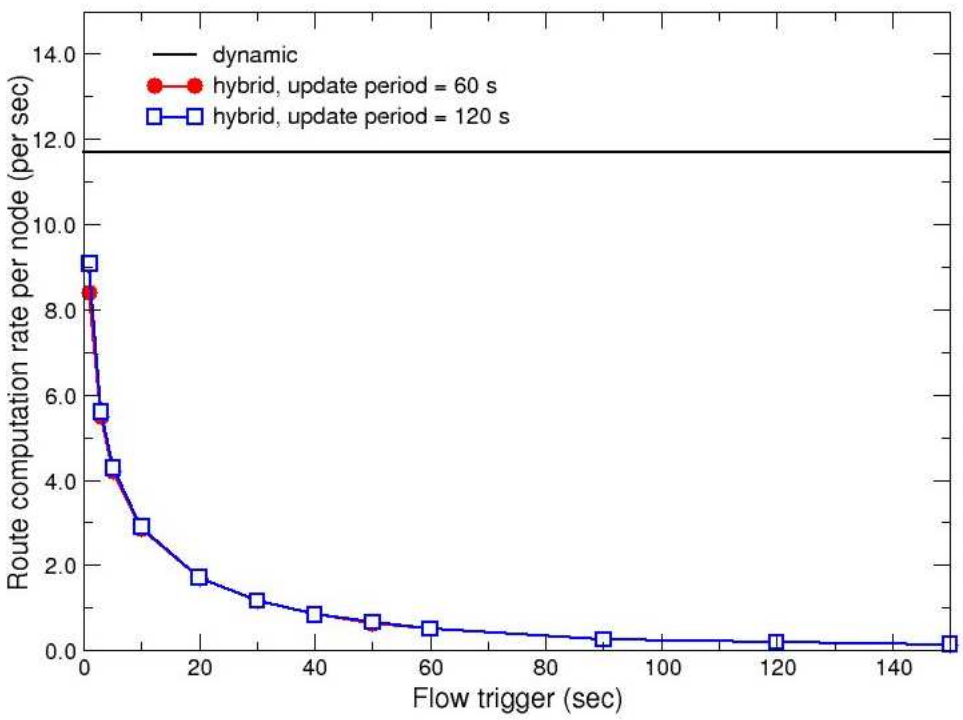


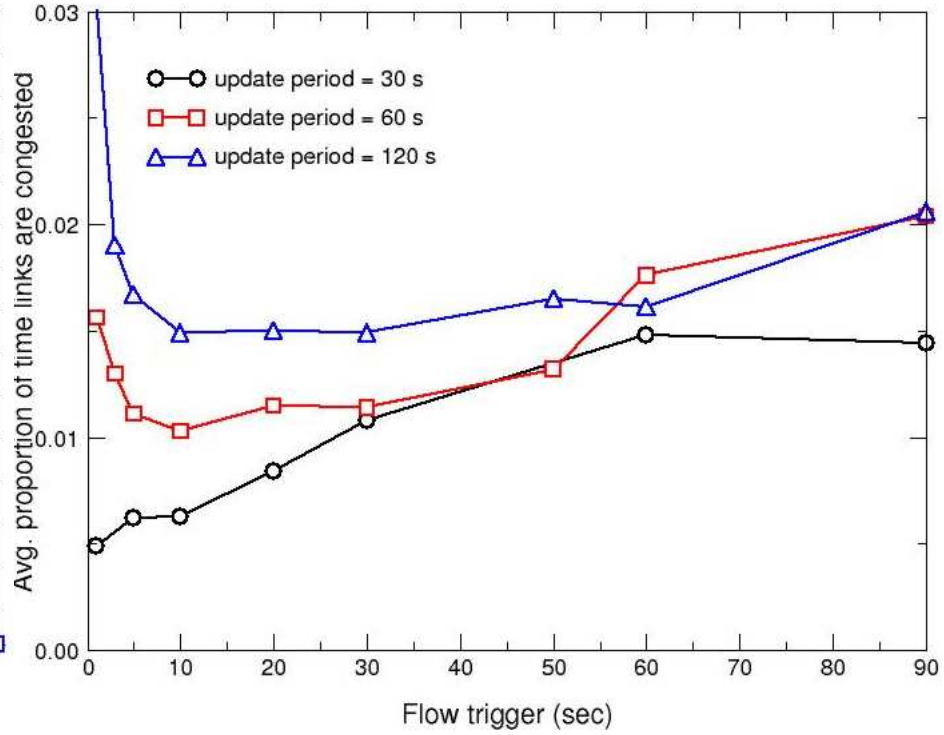
Effects of Stale Link-State



- traditional dynamic routing suffers from staleness
- hybrid performs well even under large update periods
- nonminimal routing does not degrade hybrid routing

Choosing the Flow Trigger





- Flow trigger directly affects computational overhead
- Small trigger increases sensitivity to link-state staleness
- Large trigger limits adaptability to load fluctuations

Summary and Open Questions

- Hybrid routing
  - detect and dynamically route long-lived flows
  - achieve stability under reasonable protocol overhead
- What about …
  - impact of packet-level dynamics (e.g., TCP)
  - using measured link state
  - different ways to select the flow trigger
  - integration with MPLS