Slide 0

A Scalable Architecture for Fair Leaky-Bucket Shaping

Jennifer Rexford
AT&T Labs Research
jrex@research.att.com

Flavio Bonomi
ZeitNet, Cabletron
flavio@zeitnet.com

Albert Greenberg
AT&T Labs Research
albert@research.att.com

Albert Wong
Lucent Technologies
albertwong@lucent.com

Slide 1

Outline

- Mechanisms for end-to-end quality of service
- Scalable shaper architecture for high-speed switches
- Integrating traffic shaping with fair scheduling
- Performance evaluation through simulation
- Conclusions and future work

Slide 2

Constraints in Emerging Networks

Network traffic sharing a single link

- Thousands of connections/sessions
- Diverse traffic parameters (video, audio, data)
- Diverse performance requirements (rate, delay, loss)

⇒ powerful algorithms for managing bandwidth & buffers

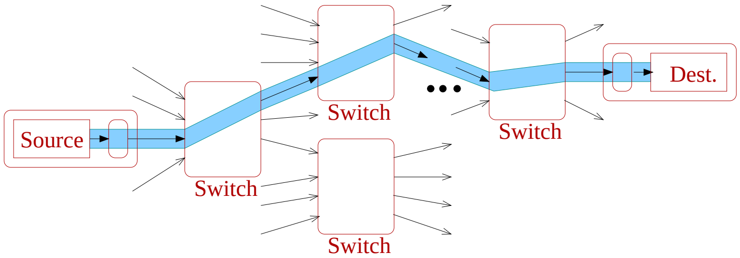
Network technology trends

- High-speed links (e.g., 155 Mbit/sec; 1 Gbit/sec)
- Small data transfer unit (e.g., 53-byte ATM cell)

⇒ scalable & efficient hardware switch designs

Slide 3

End-to-End Quality of Service



- Select a suitable multi-hop route to the destination
- Reserve buffer and bandwidth resources
- Enforce connection traffic contract (policing/shaping)
- Schedule traffic to satisfy performance requirements

Slide 4

Shaping a Large Number of Connections

Network edge:

- Large-scale web or video servers with multimedia data
- Shaping as a service for multiple smaller end systems
- Shaping to avoid overloading the receiving LAN/host

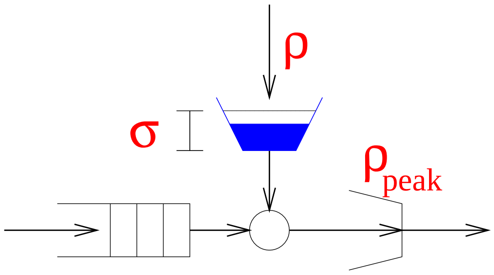
Network interior:

- Boundary between different providers in the backbone
- Schedulers that reshape connections to limit jitter
- Available-bit-rate connections with rate enforcement at "virtual sources" inside the network

$\Longrightarrow$ high-speed shaping for a large number of connections

Slide 5

Leaky-Bucket Traffic Shaping

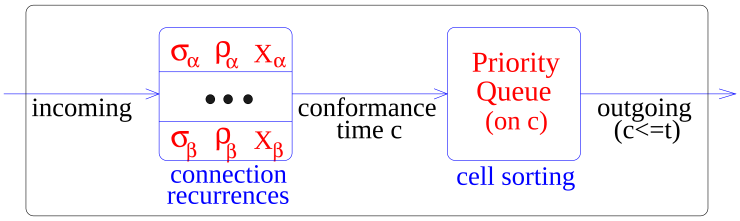


Connection traffic parameters:

- Burst size σ
- Sustainable bandwidth ρ
- Peak rate ρ_{peak}

Slide 6

Scalable Traffic Shaping



- **Traffic parameters**: burstiness σ, bandwidth ρ, and leaky-bucket state X
- **Recurrences**: compute conformance time c
- **Cell sorting**: transmit cell with smallest $c \leq t$

$\Rightarrow$ priority queue with 1000s connections and diverse c?
$\Rightarrow$ handling multiple cells with the same value of c?

Slide 7

Performance Scalability

Performance challenges:

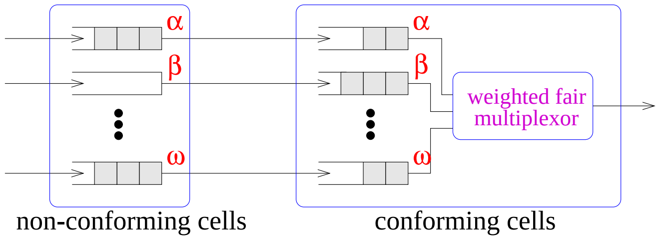
- Effects of ties (multiple cells with same c value)
- Large number of connections ganging up on others
- Bursty connections denying service to others
- Inaccuracies from approximations in implementation

Performance implications:

- Cell shaping delay ($t - c > 0$)
- Traffic distortions ($\sigma^{out} > \sigma$)
- Difficulty of provisioning downstream resources
- Cell loss (buffer overflow) at downstream switches

Slide 8

Fair Leaky-Bucket Traffic Shaping



Idealized model:

- Connections α with shaping parameters $(\sigma_\alpha, \rho_\alpha)$
- Set of connections C with conforming cells
- If $\alpha \in C$ service connection at rate $\geq \rho_\alpha$

$\Rightarrow$ weighted fair scheduling can ensure $\sigma_\alpha^{out} \approx \sigma_\alpha$

Slide 9

Trading Off Accuracy and Complexity

Implementation scalability challenges:

- Number of connections and cells in the priority queue
- Range of burst (σ) and bandwidth (ρ) parameters

$\Longrightarrow$ large number of cells with wide range of c values

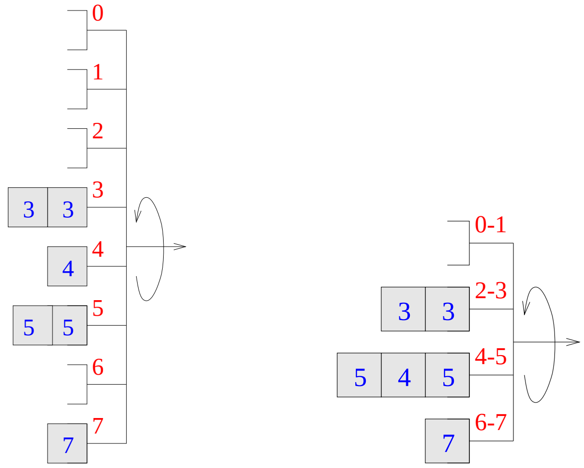
Calendar queue sorting:

- Collection of b FIFO sorting bins
- Grain of g time units per sorting bin
- Incoming cell joins the bin based on its c value

$\Longrightarrow$ tension between range bg and grain g of bins

Slide 10

Traffic Shaping With Calendar Queues



(a) $b = 8$ bins of grain $g = 1$ (b) $b = 4$ bins of grain $g = 2$

Slide 11

Scalability Challenges

Large number of bins comes from:

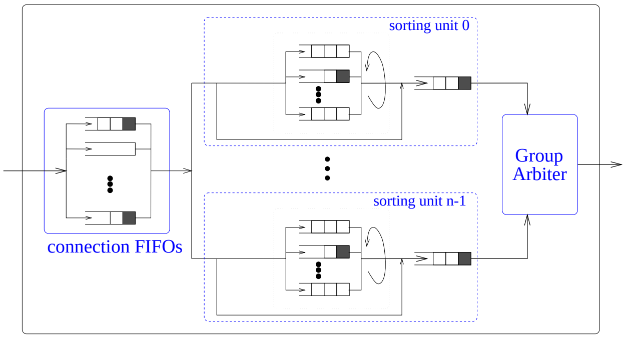
- Large c from low-rate traffic (e.g., $16 - 64$ kilobits/sec)
- Fine accuracy for high-rate traffic (e.g., $10 - 100$ megabits/sec)
- Multiple cells from same connection
- Transient backlog of conforming cells

Possible solutions:

- Large grain g of sorting bins for low-rate connections
- Small range bg of sorting bins for high-rate connections
- Only head-of-line cell in sorting bins (per-connection queueing)
- Avoid sorting by c for conforming, head-of-line cells

Slide 12

Hierarchical Architecture

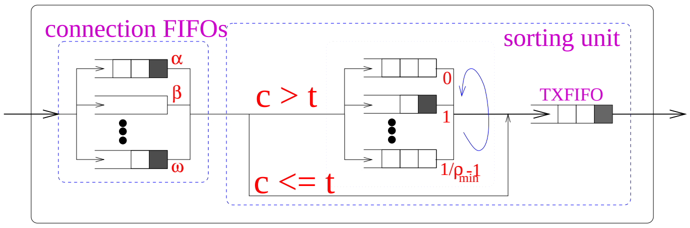


- Group connections with rates ρ within a factor m
- Tailor grain g and range bg to each group
- Fair arbitration between groups and within groups

$\Longrightarrow$ fair shaping with $n = \log_m\{\rho_{\max}/\rho_{\min}\}$ groups with b bins

Slide 13

Scalable Architecture



Where does a connection-α cell with conformance time c reside?

- Not a head-of-line cell $\Rightarrow$ Connection FIFO (α)
  (move to sorting unit after connection's previous cell departs)
- Head-of-line non-conforming cell $\Rightarrow$ Calendar FIFO (c mod b)
  (move to TXFIFO when bin c mod b reaches conformance)
- Head-of-line conforming cell $\Rightarrow$ Transmission FIFO
  (more to outgoing link upon reaching head of TXFIFO)

Slide 14

Weighted Fairness Within a Group

Initial scheme with per-connection queueing:

- Cell departure inserts next cell into the TXFIFO
- Round-robin service amongst conforming connections
- Favors connections with smaller ρ values
- Distort high-bandwidth (large ρ) connections

Minor change for weighted fairness:

- Allow α_i to have $\lfloor \rho_{\alpha_i} / \rho_{\min_i} \rfloor$ cells in the sorting unit
- Maintain counter of number of cells in the sorting unit
- Event-driven algorithm and no additional sorting bins

$\Rightarrow$ Low-cost approximation to weighted round robin

Slide 15

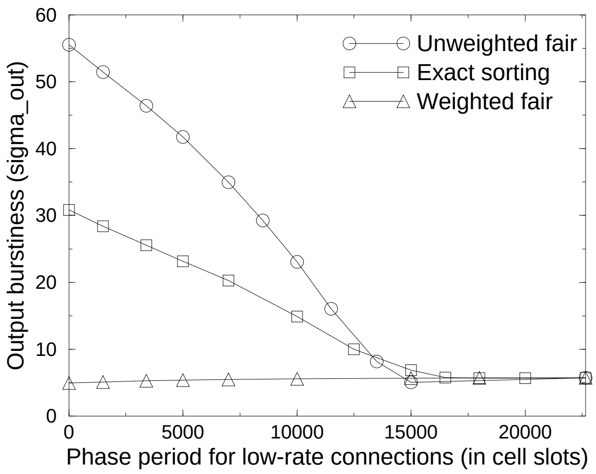
Performance Evaluation

- **Source pattern:** greedy on/off traffic with (σ, ρ)
- **Start times:** uniformly distributed in $[0, x]$
- **Congestion:** varied using start-time parameter x
- **Performance metric:** avg output burstiness $\overline{\sigma}^{out}$

	Arrival		Shaping	
N	σ	ρ	σ	ρ
320	50	0.0022	25	0.0025
10	50	0.0090	5	0.0100

Slide 16

Fair Shaping Within a Group



Slide 17

Implementation and performance scalability:

- Hierarchical arbitration (between and within groups)
- Tailoring range and grain of sorting bins in each group
- Integration of shaping and fair scheduling in group

⇒ scalability in the number of connections, range of rates, and range of burstiness

Future work:

- Performance evaluation under realistic traffic patterns
- Extensions to the hierarchical fair arbitration policies
- Schedulers with separate rate and delay parameters

⟹ balance between powerful algorithms and efficient architectures in high-speed switches