Improving Internet Congestion Control and Queue Management Algorithms

Wu-chang Feng

March 17, 1999

Final Oral Examination

Outline

- Motivation
- Congestion control and queue management today (TCP, Drop-tail, RED)
- Solutions for reducing packet loss in the Internet
  - ECN
  - Adaptive RED
  - SubTCP
  - Blue
  - Stochastic Fair Blue
- Providing scalable QoS over the Internet
- Conclusion

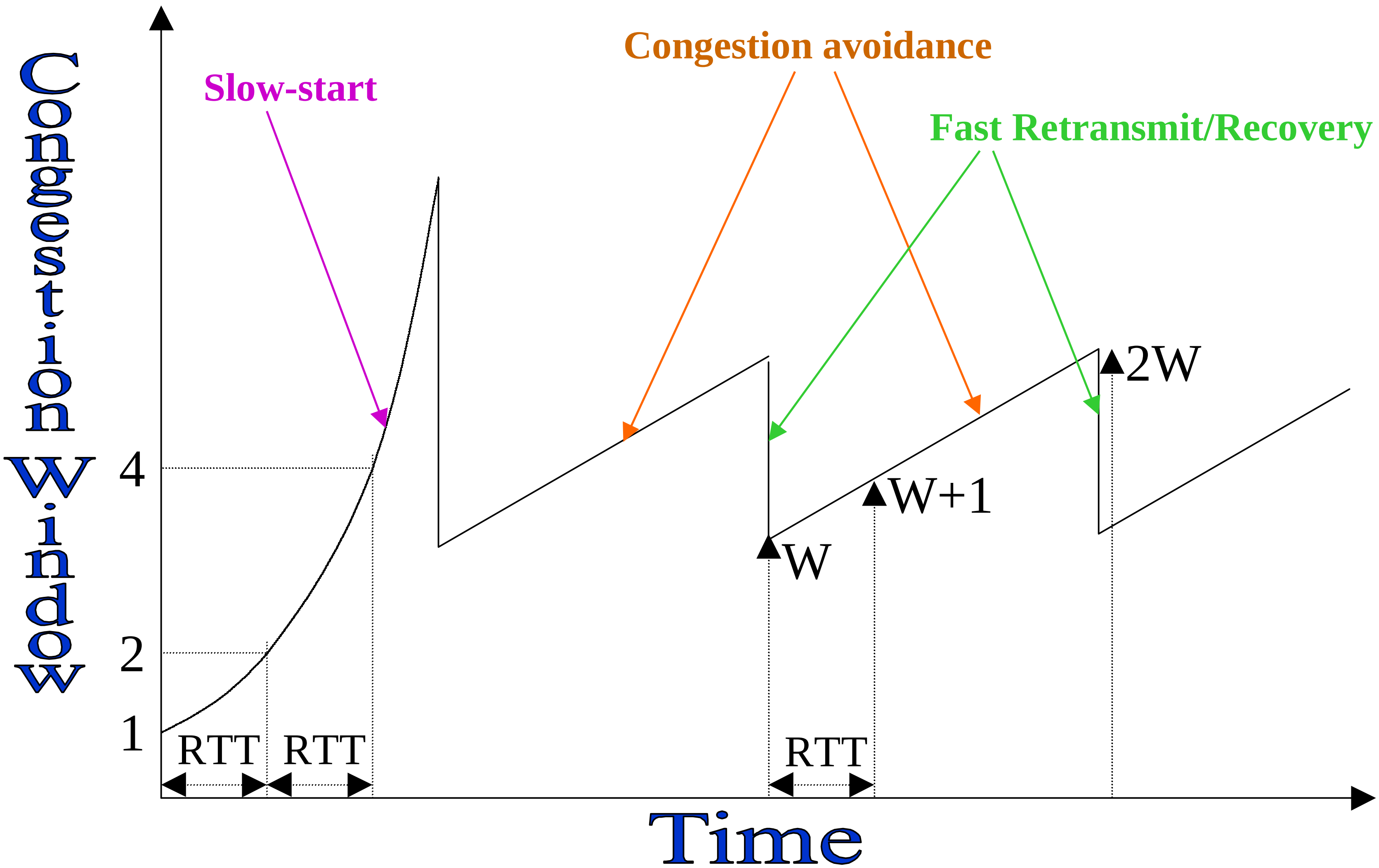
Motivation

- Exponential increase in network demand
  - Rising packet loss rates
    - 17% loss rates reported [Paxson97]
  - Low utilization and goodput
  - Potential for congestion collapse
- Goal of dissertation
  - Examine causes
  - Solutions for maximizing network efficiency in times of heavy congestion
  - 0% packet loss, 100% link utilization, low queuing delay

Congestion Control Today

- TCP
  - Instrumental in preventing congestion collapse
  - Limits transmission rate at the source
  - Window-based rate control
    - Increased and decreased based on network feedback
    - Implicit congestion signal based on packet loss
    - Slow-start
    - Fast-retransmit, Fast-recovery
    - Congestion avoidance

Example of TCP Windowing

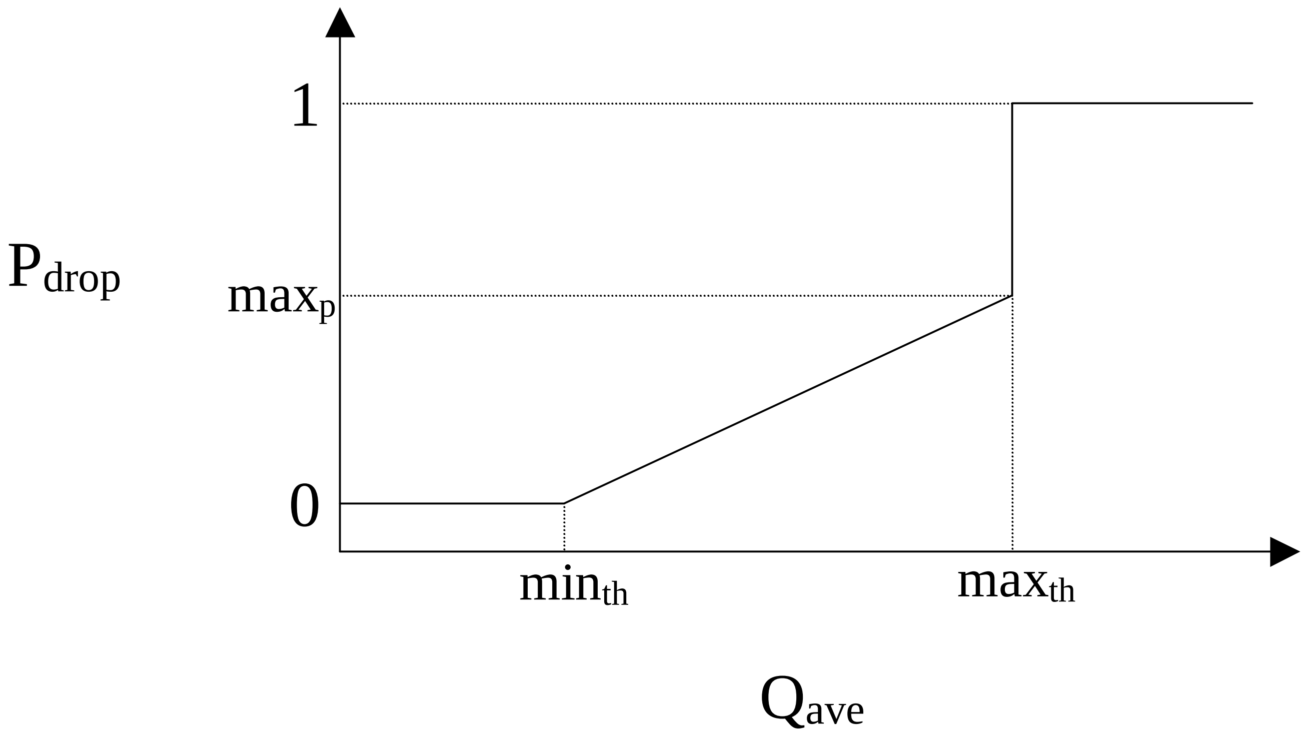


Drop-tail Queue Management

- Default queue management mechanism
- Packets dropped upon queue overflow
- Problems
  - Global synchrony (poor utilization)
  - Late congestion notification (packet loss)
- Solution
  - Randomize
  - Early detection of incipient congestion

RED Queue Management

- RED (Random Early Detection)
  - Keep EWMA of queue length (Q_{ave})
  - Increase in EWMA triggers random drops
- Basic algorithm



Question

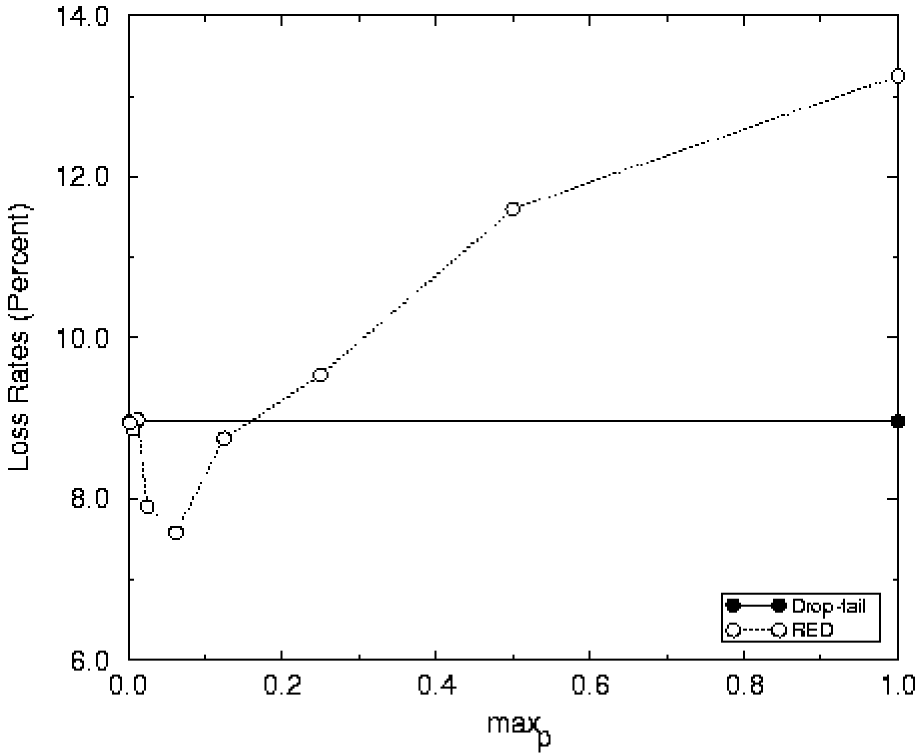
- If TCP and RED are so good, why is network efficiency so bad?
- Problems (and solutions)
  - Congestion notification through packet loss (ECN)
  - RED not adaptive to congestion (Adaptive RED)
  - TCP too aggressive at high loads (SubTCP)
  - RED depends on queue lengths (Blue)
  - Non-responsive flows (Stochastic Fair Blue)

Outline

- Motivation
- Congestion control and queue management today (TCP, Drop-tail, RED)
- Solutions for reducing packet loss in the Internet
  - ECN
  - Adaptive RED
  - SubTCP
  - Blue
  - Stochastic Fair Blue
- Providing scalable QoS over the Internet
- Conclusion

RED and Packet Loss

- Impact of RED on loss rates minimal
- Loss rates are a first order function of TCP



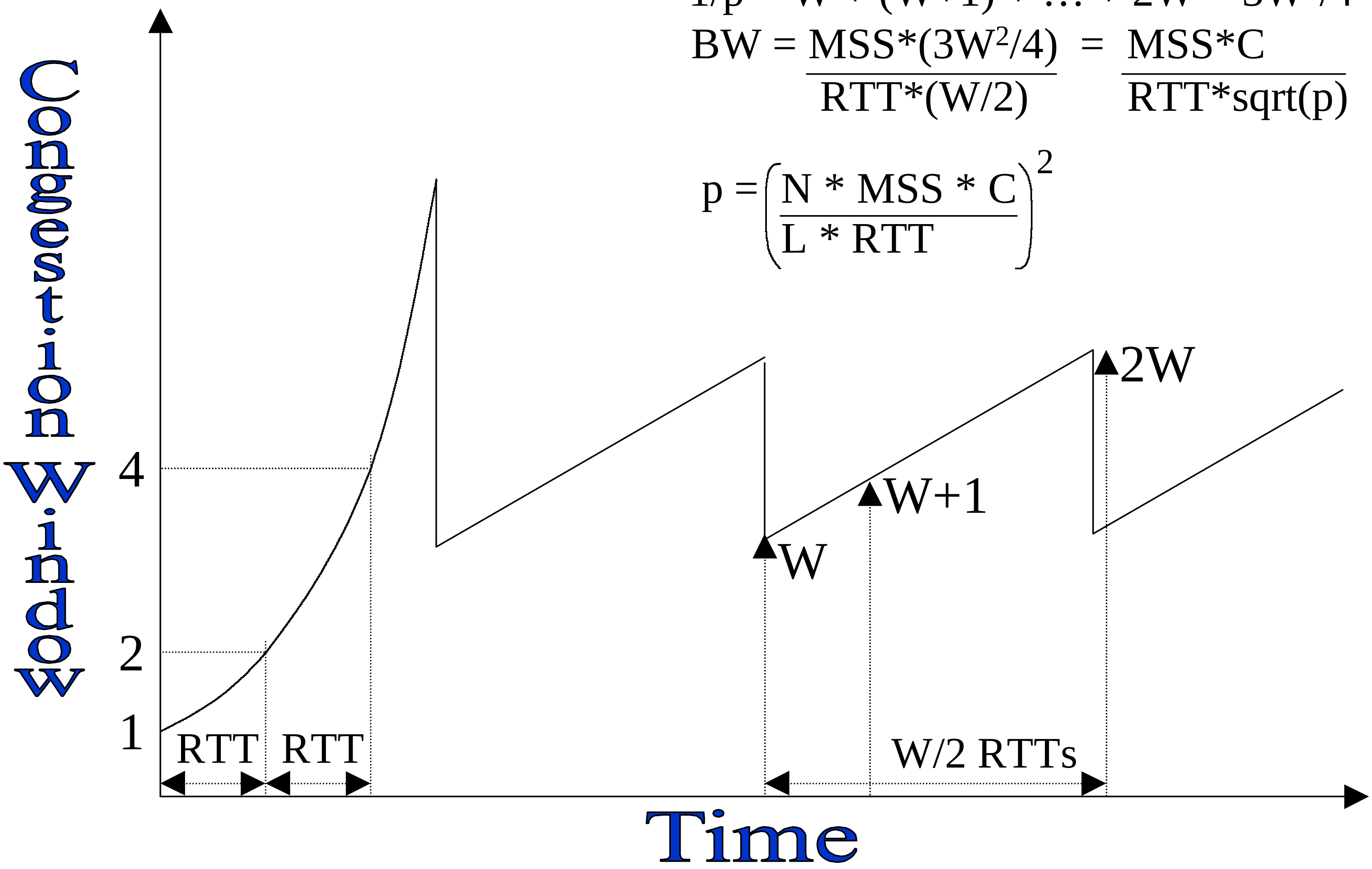
64 connections
10Mbs link

TCP Revisited

$$1/p = W + (W+1) + \ldots + 2W = 3W^2/4$$

$$BW = \frac{MSS*(3W^2/4)}{RTT*(W/2)} = \frac{MSS*C}{RTT*sqrt(p)} = \frac{L}{N}$$

$$p = \left(\frac{N * MSS * C}{L * RTT}\right)^2$$



Comments on Model

- Reducing N - [Balakrishnan98]
- Increasing RTT - [Villamizar94]
- Decreasing MSS - [Feng98]
- Loss rates as a function of N between linear and quadratic
  - Fair share assumption (L/N) - [Morris97]
  - No retransmission timeouts - [Padhye98]

$$p = \left(\frac{N * MSS * C}{L * RTT}\right)^2$$

ECN

- Without ECN, packet loss rates will remain high
- IETF ECN WG (1998)
- RFC 2481 - January 1999 (Experimental standard)
  - 2-bits in “DS Field” of IPv4/IPv6 headers (ECT, CE)
  - 2-bits in “TCP Flags” field of TCP (CWR, ECN Echo)

Outline

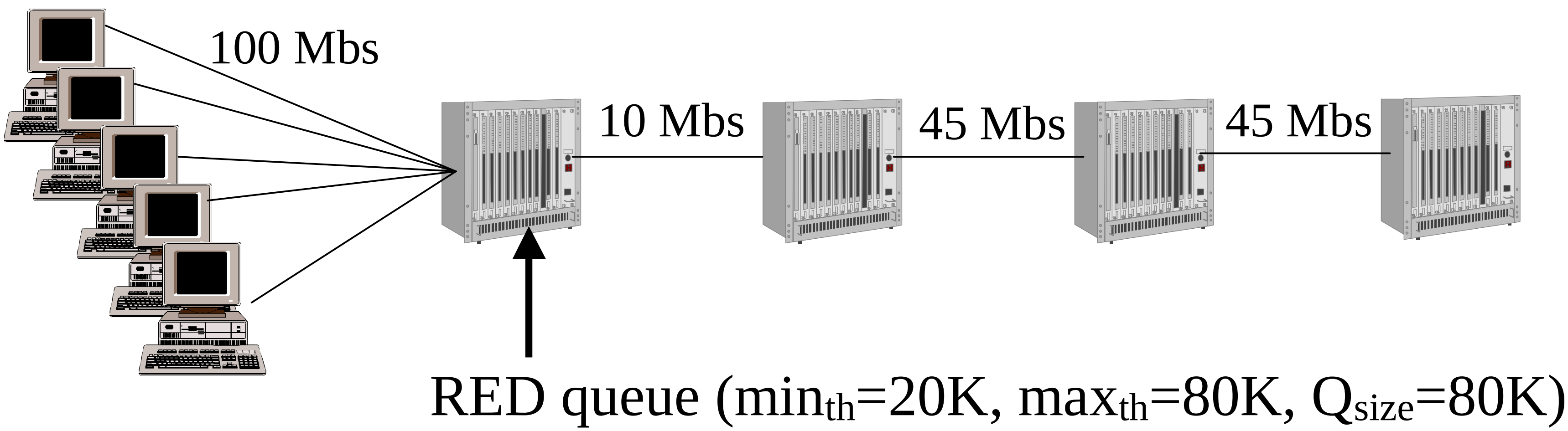
- Motivation
- Congestion control and queue management today (TCP, Drop-tail, RED)
- Solutions for reducing packet loss in the Internet
  - ECN
  - Adaptive RED
  - SubTCP
  - Blue
  - Stochastic Fair Blue
- Providing scalable QoS over the Internet
- Conclusion

RED and Packet Loss

- Even with ECN, RED does not eliminate packet loss
- Problem
  - RED is not adaptive to congestion level
  - max_p constant
- Congestion notification vs. number of connections
  - N = number of connections
  - Offered load reduced by [1 - (1/2N)] per notification

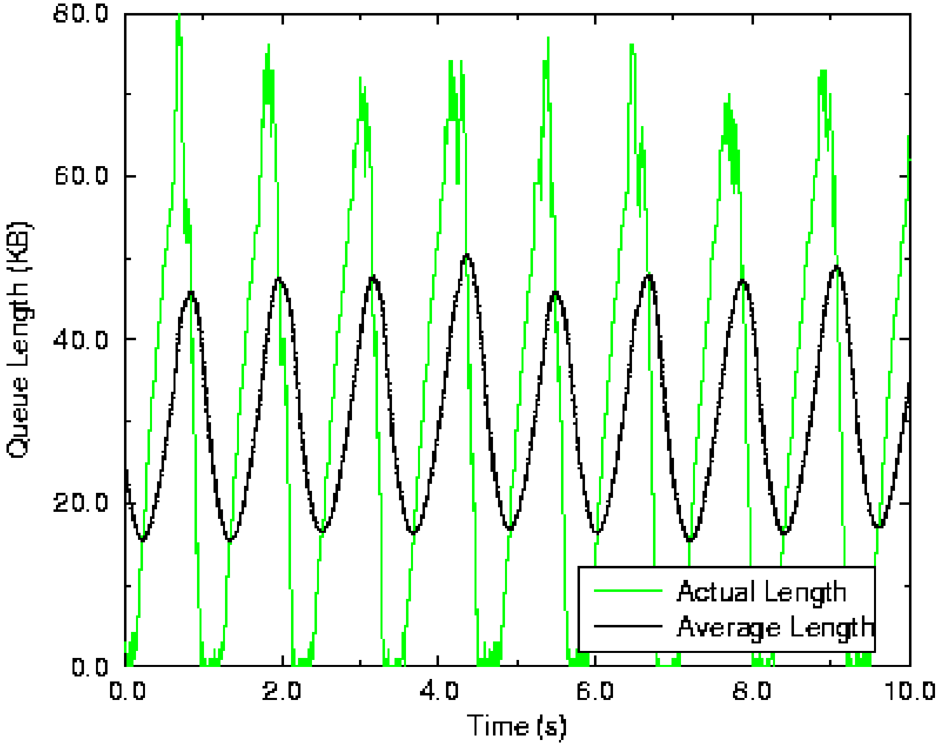
RED Experiments

- 8 or 32 TCP sources using ECN
- Conservative vs. aggressive early detection
- Simulated in ns
  - Aggressive detection: $max_p = 0.250$
  - Conservative detection: $max_p = 0.016$

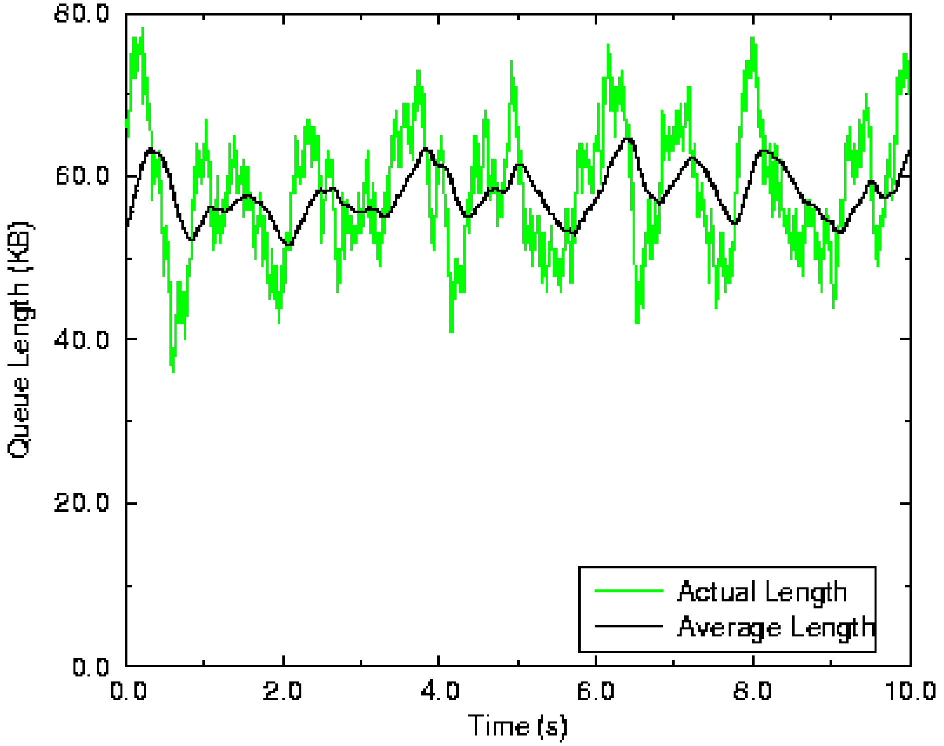


Aggressive Early Detection

8 sources

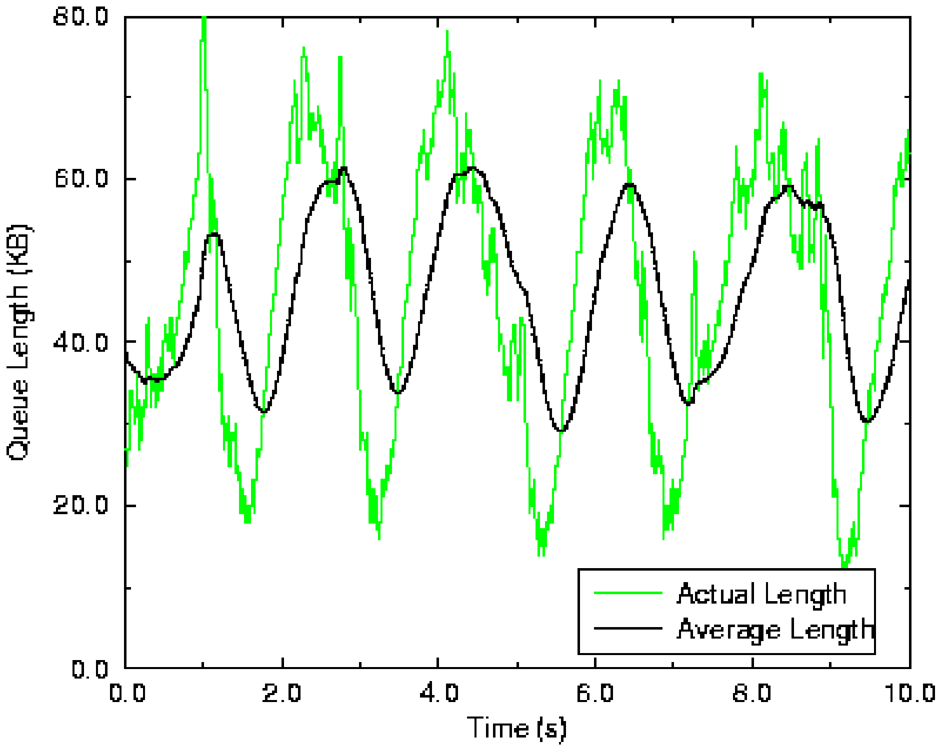


32 sources

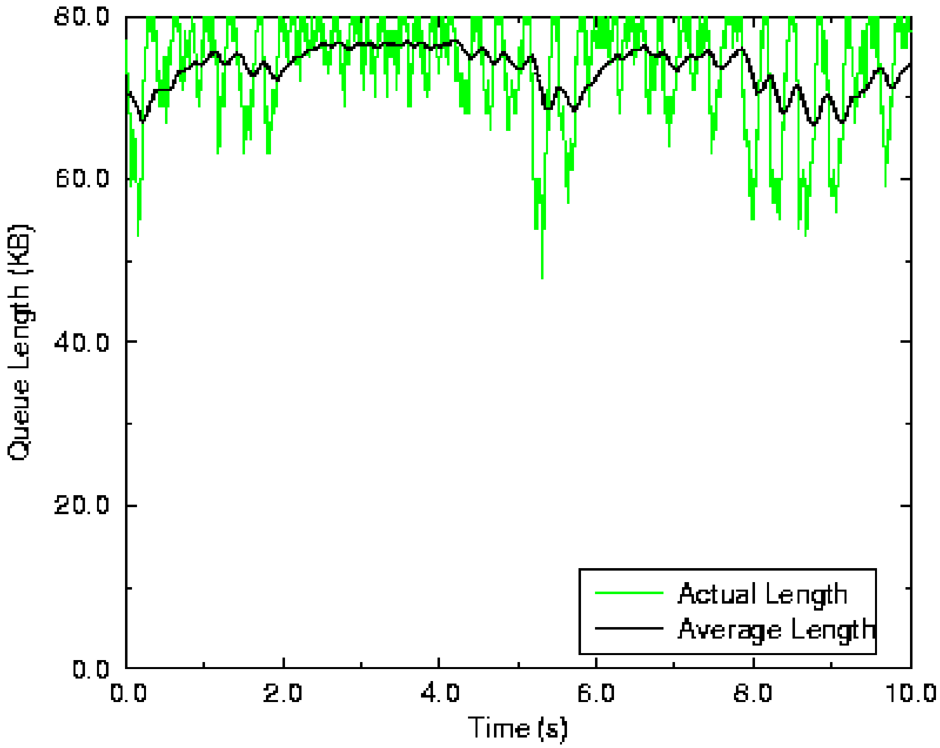


Conservative Early Detection

8 sources

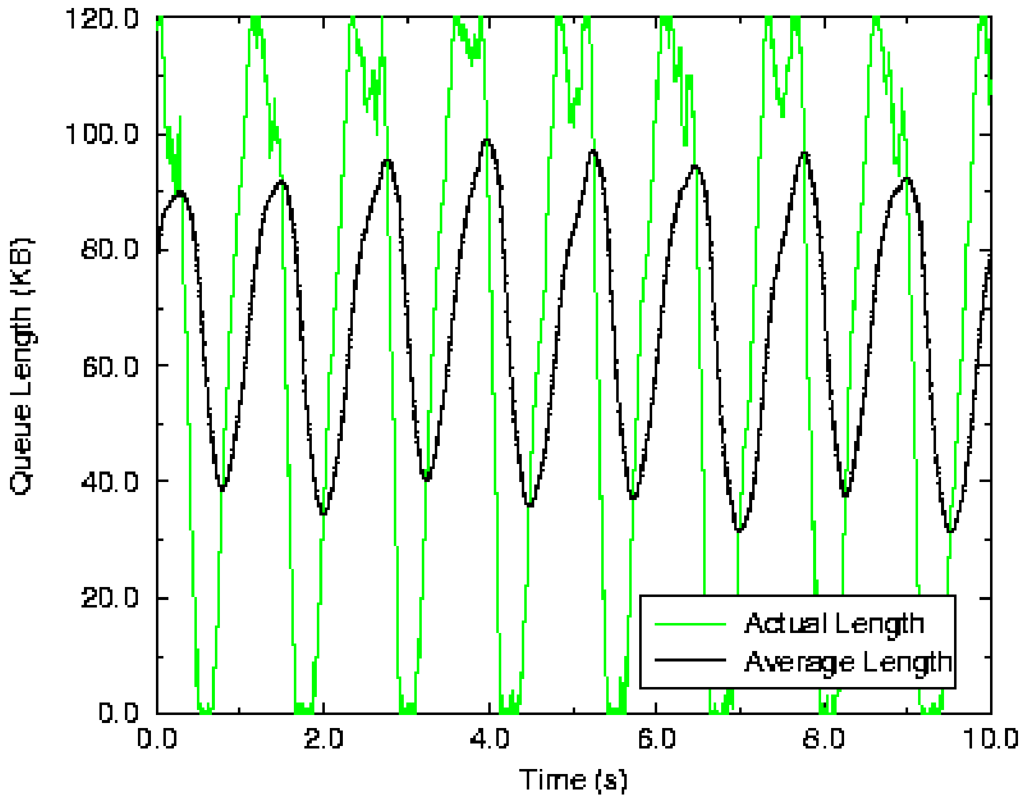


32 sources



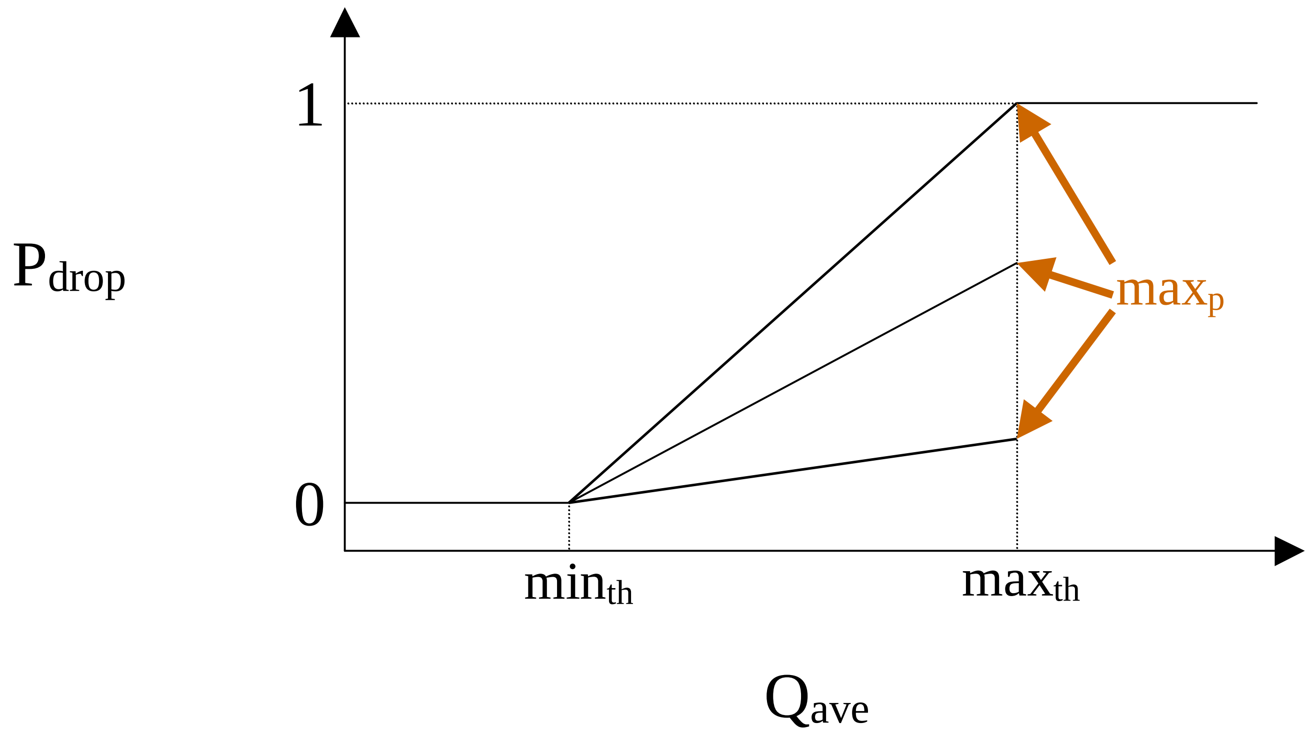
Conservative Early Detection

32 sources, Q_{len} = 120KB



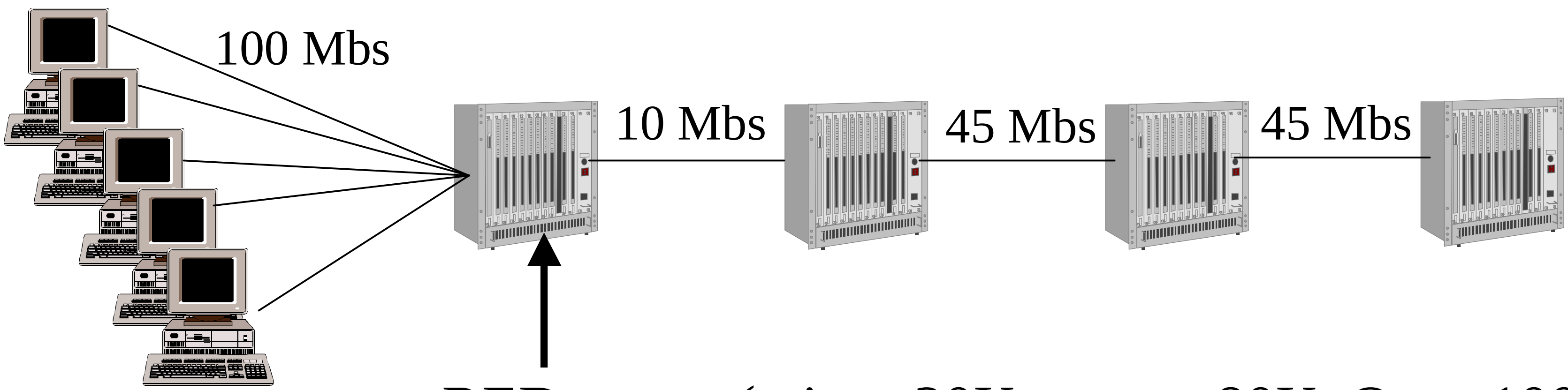
Adaptive RED

- Adapt max_p based on queue behavior
- Increase max_p when Q_{ave} crosses above max_{th}
- Decrease max_p when Q_{ave} crosses below min_{th}
- Freeze max_p after changes to prevent oscillations



Evaluation

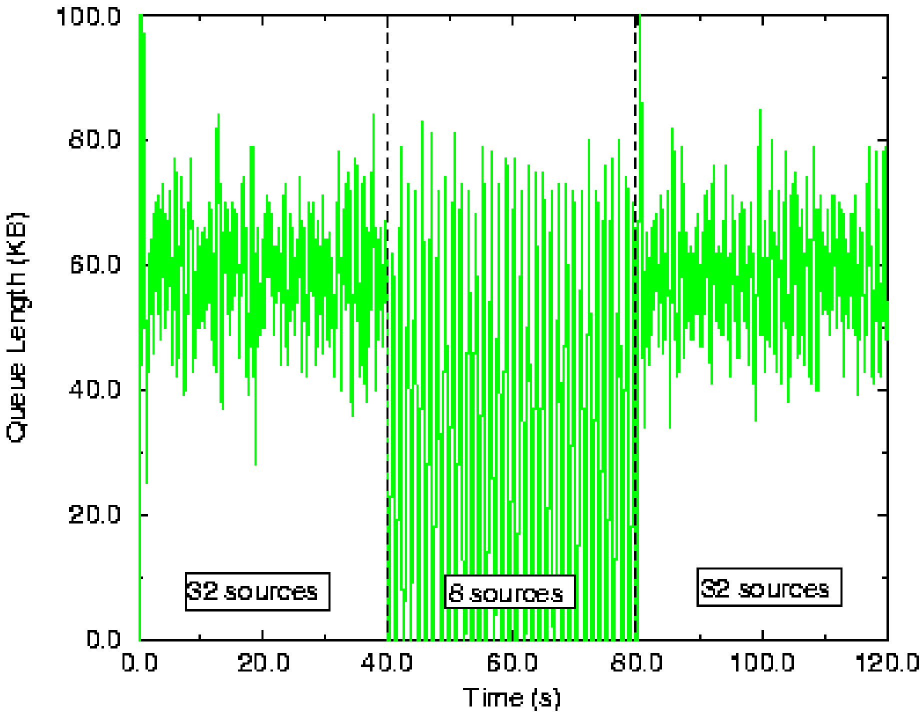
- Workload varied between 8 and 32 sources



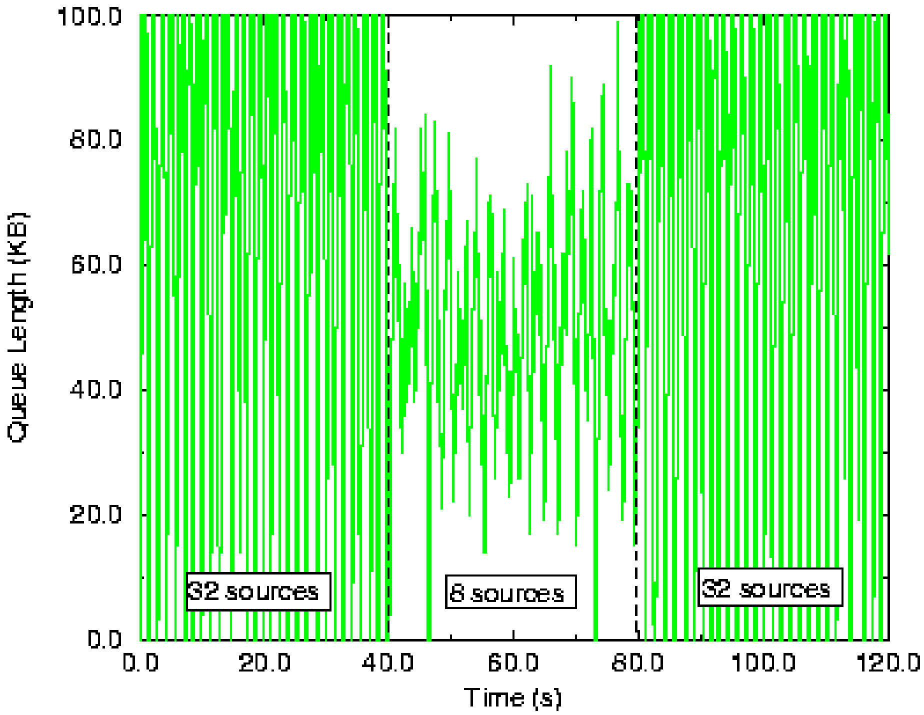
RED queue (min_{th}=20K, max_{th}=80K, Q_{size}=100K)

Static Early Detection

Aggressive

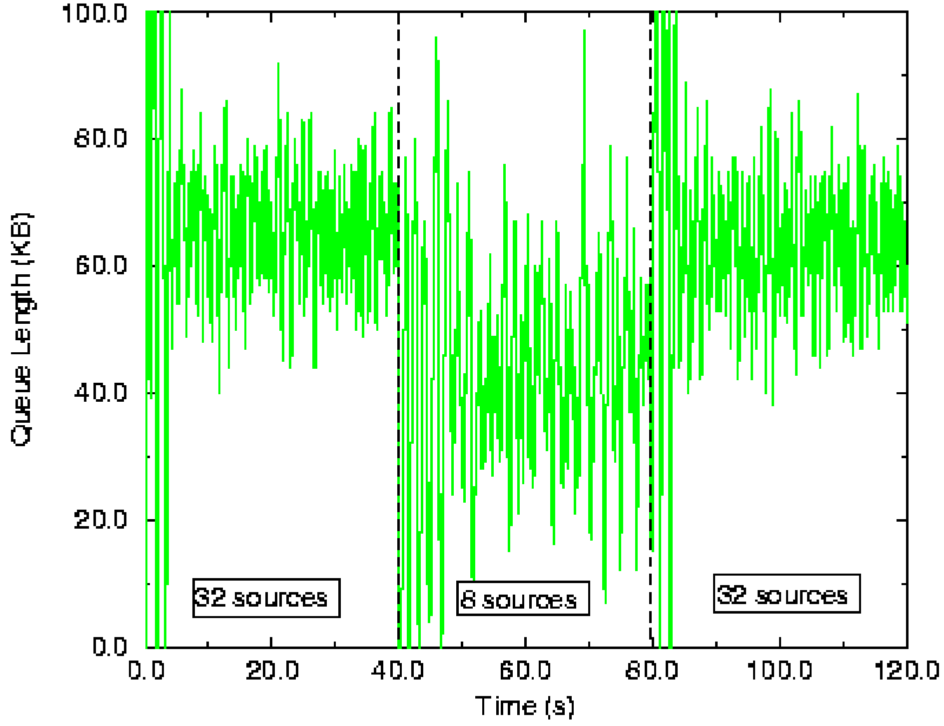


Conservative

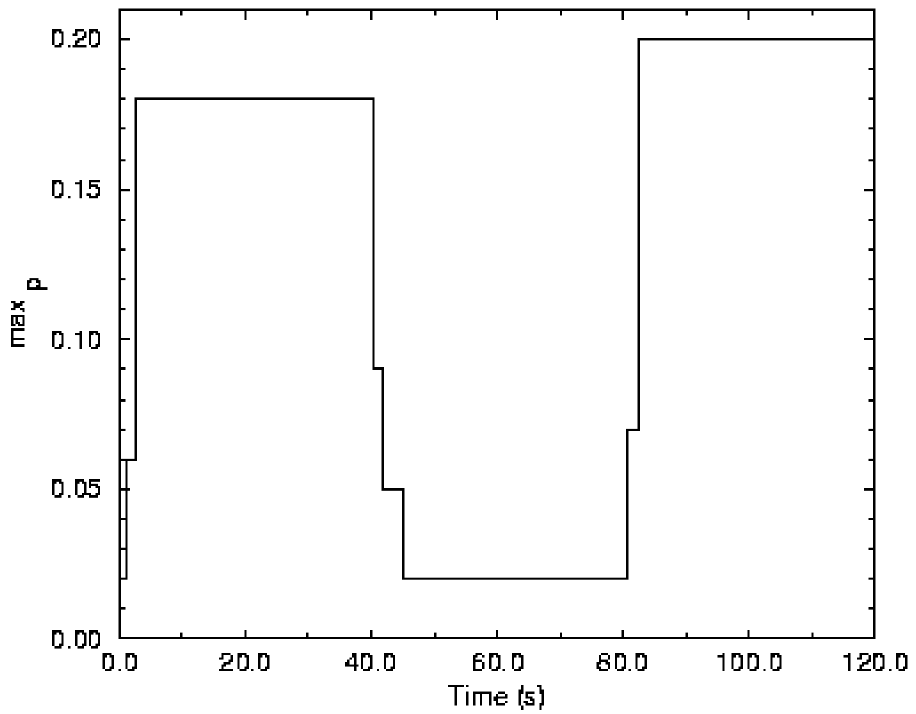


Adaptive RED

Queue length

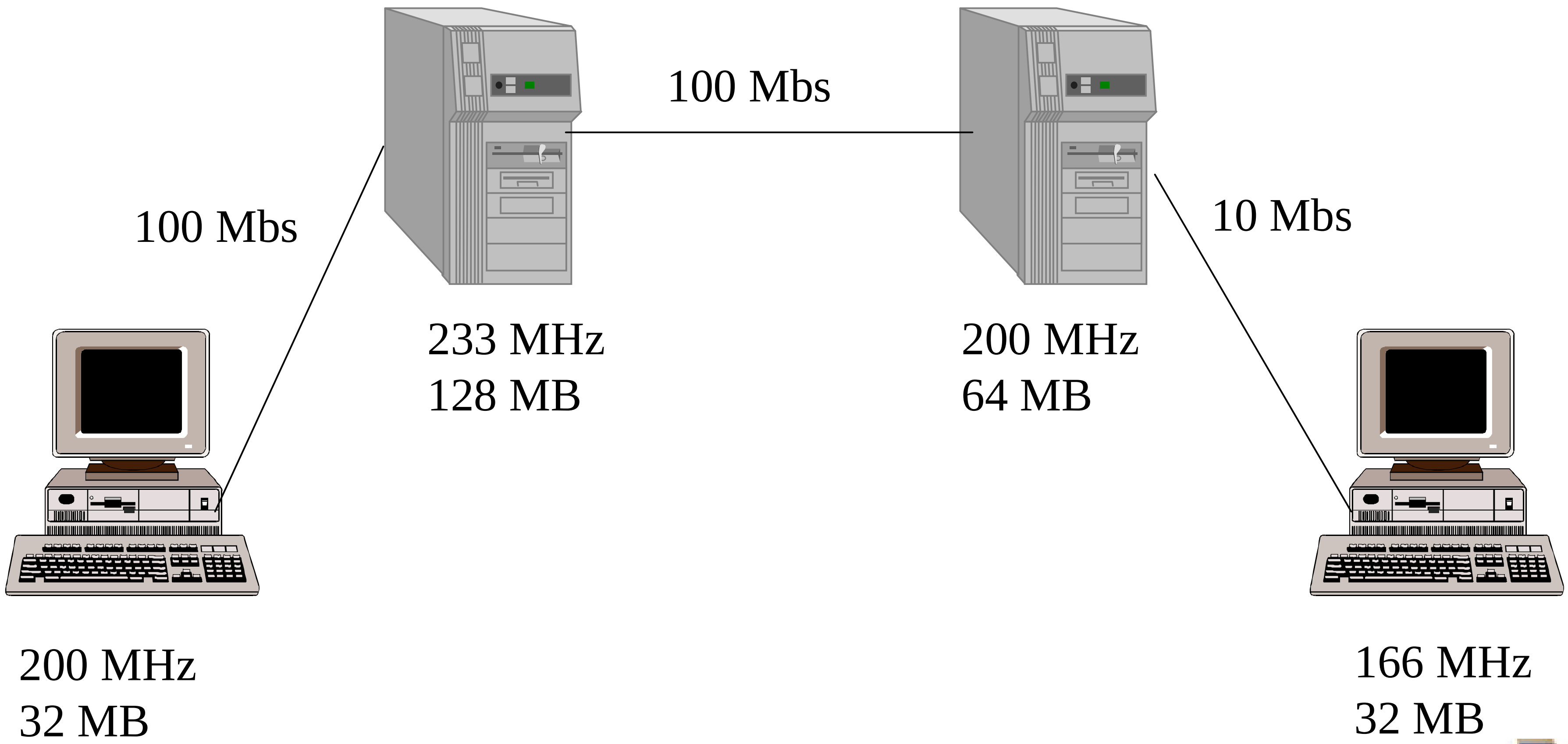


max_p



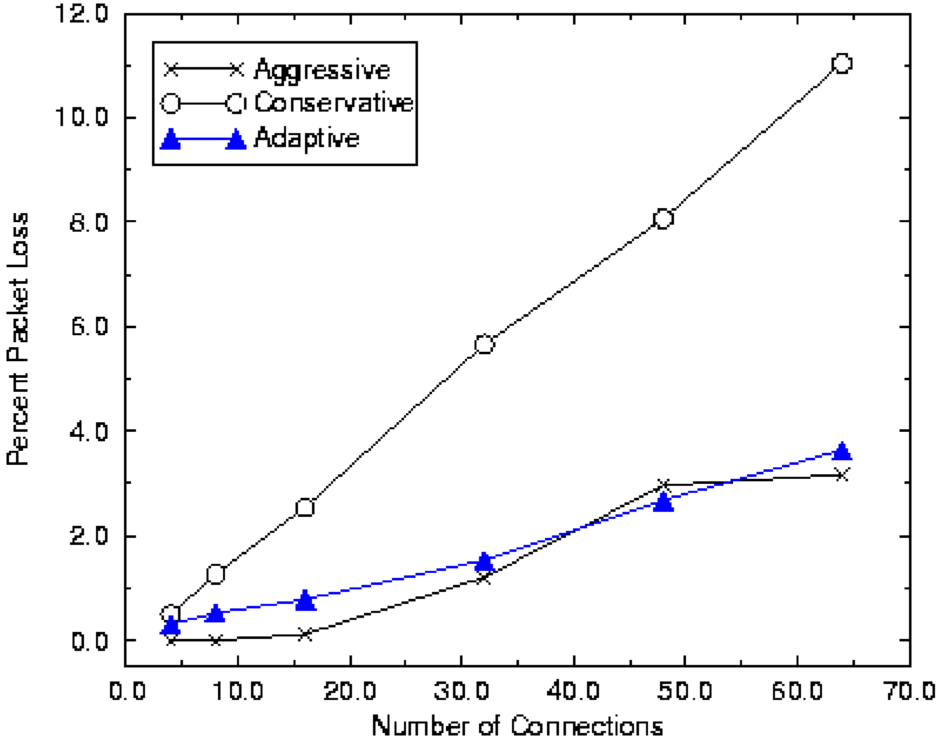
Implementation

- FreeBSD 2.2.6 + ALTQ
- Ascend, Cisco

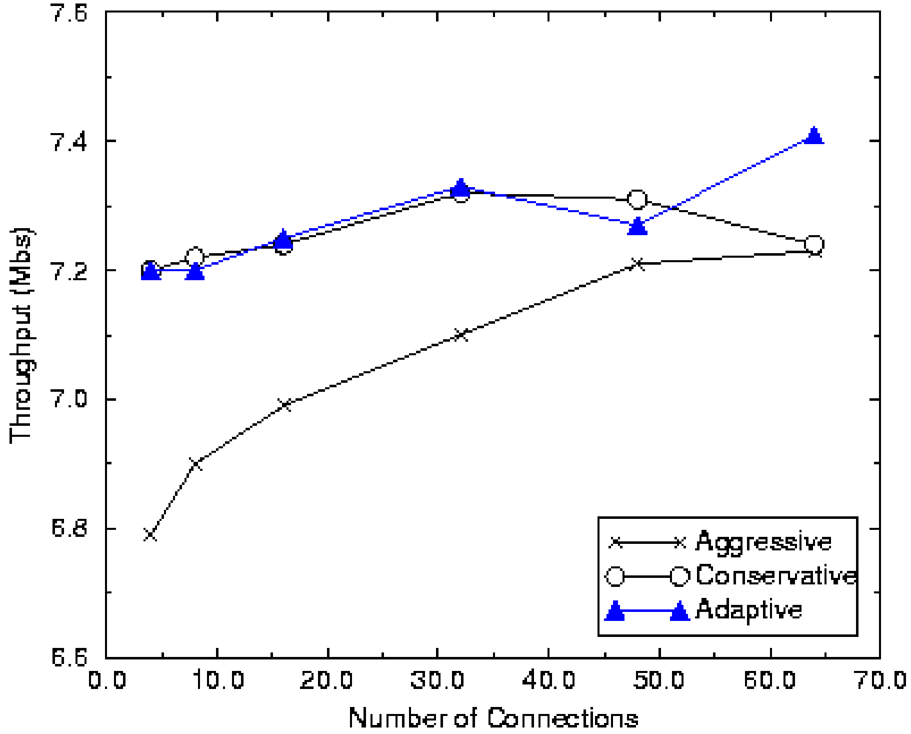


Adaptive RED Performance

Loss rates



Link utilization



Outline

- Motivation
- Congestion control and queue management today (TCP, Drop-tail, RED)
- Solutions for reducing packet loss in the Internet
  - ECN
  - Adaptive RED
  - SubTCP
  - Blue
  - Stochastic Fair Blue
- Providing scalable QoS over the Internet
- Conclusion

Fixing TCP

- Packet loss and low utilization even with Adaptive RED
- Aggregate TCP traffic too aggressive
  - Large queue fluctuations over short periods of time
  - Queue overflow before RED can react
- Example

BW*Delay = 100KB

10 sources

t=0: 10*10KB = 100KB
t=RTT: 10*11KB = 110KB
10% increase in offered load

100 sources

t=0: 100*1KB = 100KB
t=RTT: 100*2KB = 200KB
100% increase in offered load

Fixing TCP

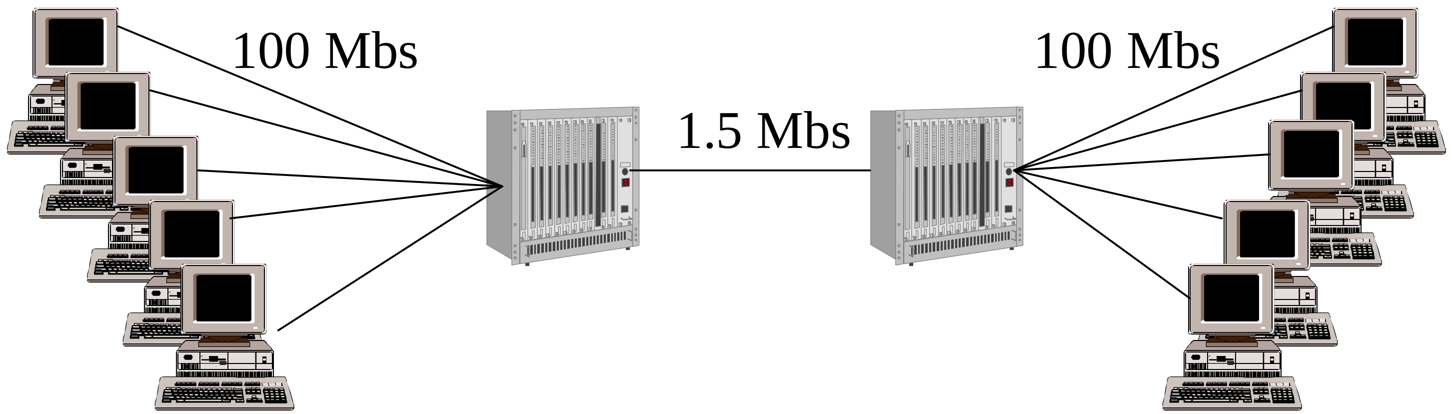
- Limit increase in aggregate TCP per RTT
- TCP
  - Limit window increases by X% per RTT
- Bottleneck link
  - Leave space to buffer X% higher than capacity per RTT

SubTCP

- Make TCP more conservative
- Slow-start unmodified
- Congestion avoidance algorithm
  - min(1, cwnd * X%)
- Modified exponential back-off algorithm

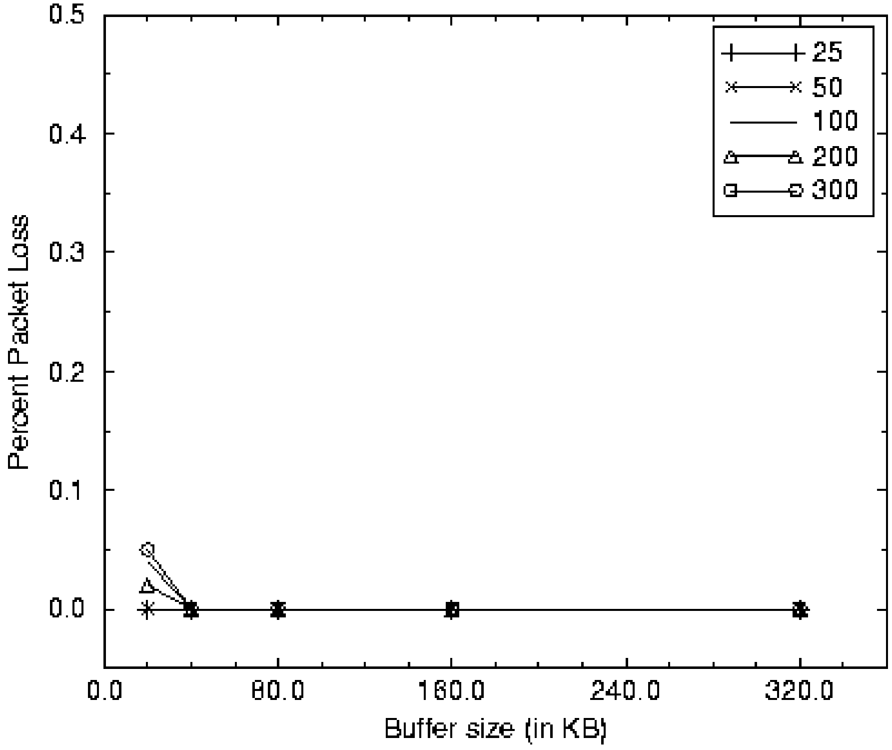
SubTCP Evaluation

- 25-300 connections over T1 link
- X=10%
- Simulated in ns

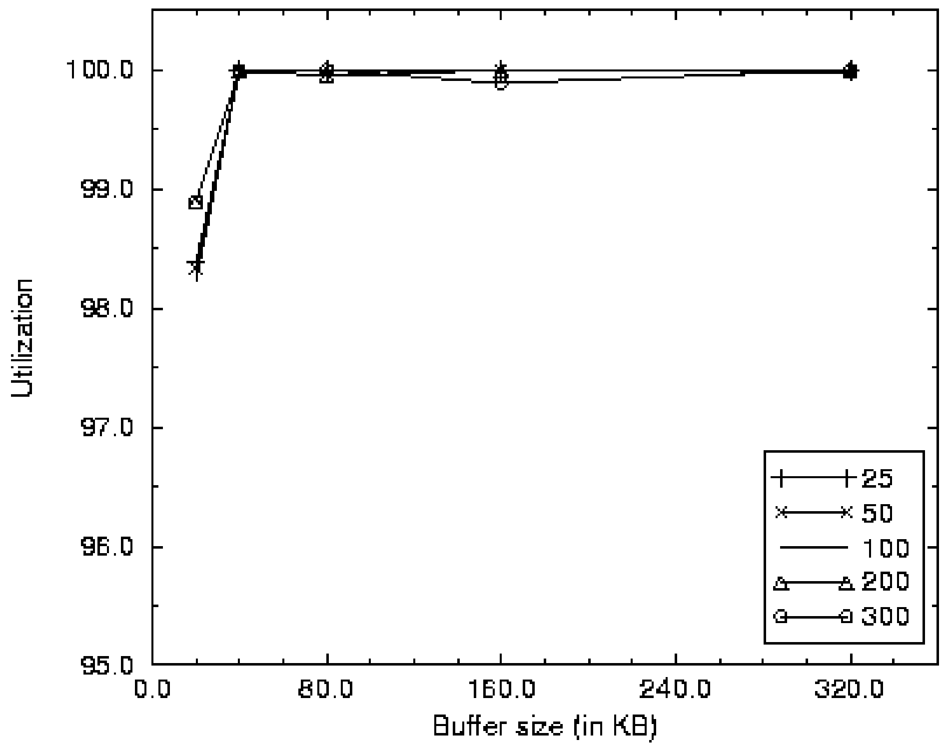


SubTCP Evaluation

Loss rates

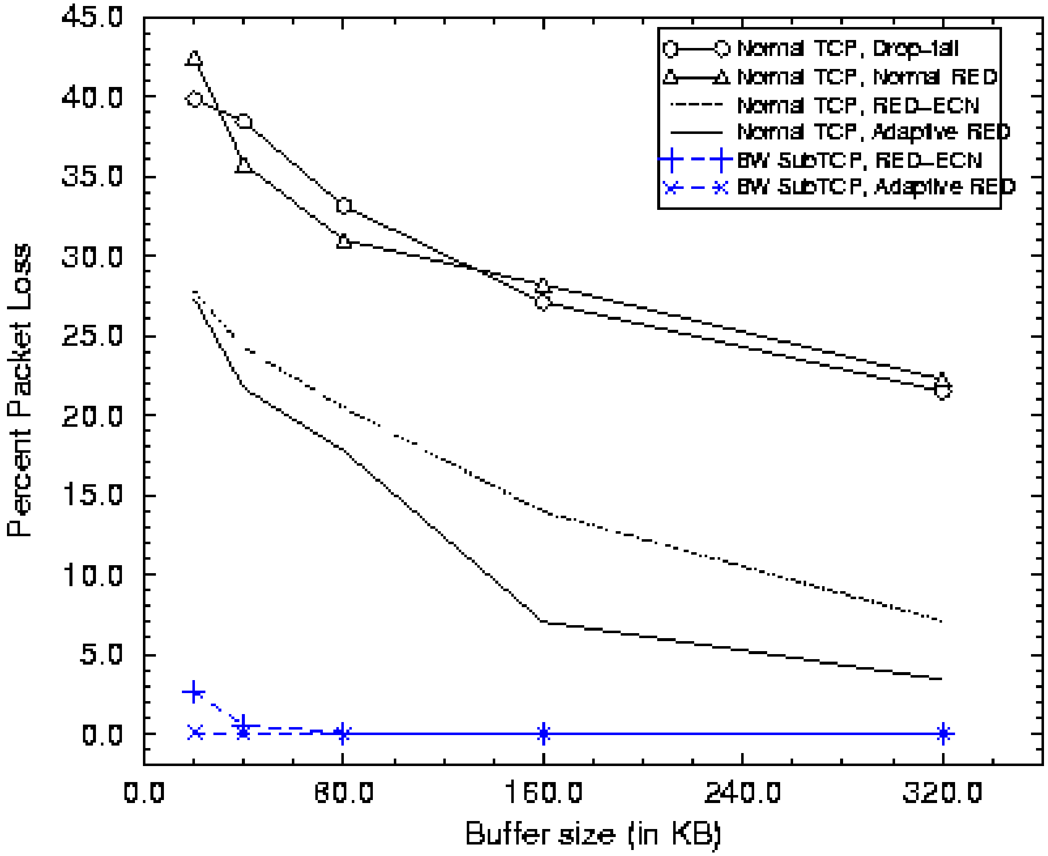


Link utilization



Comparison of Approaches

300 connections



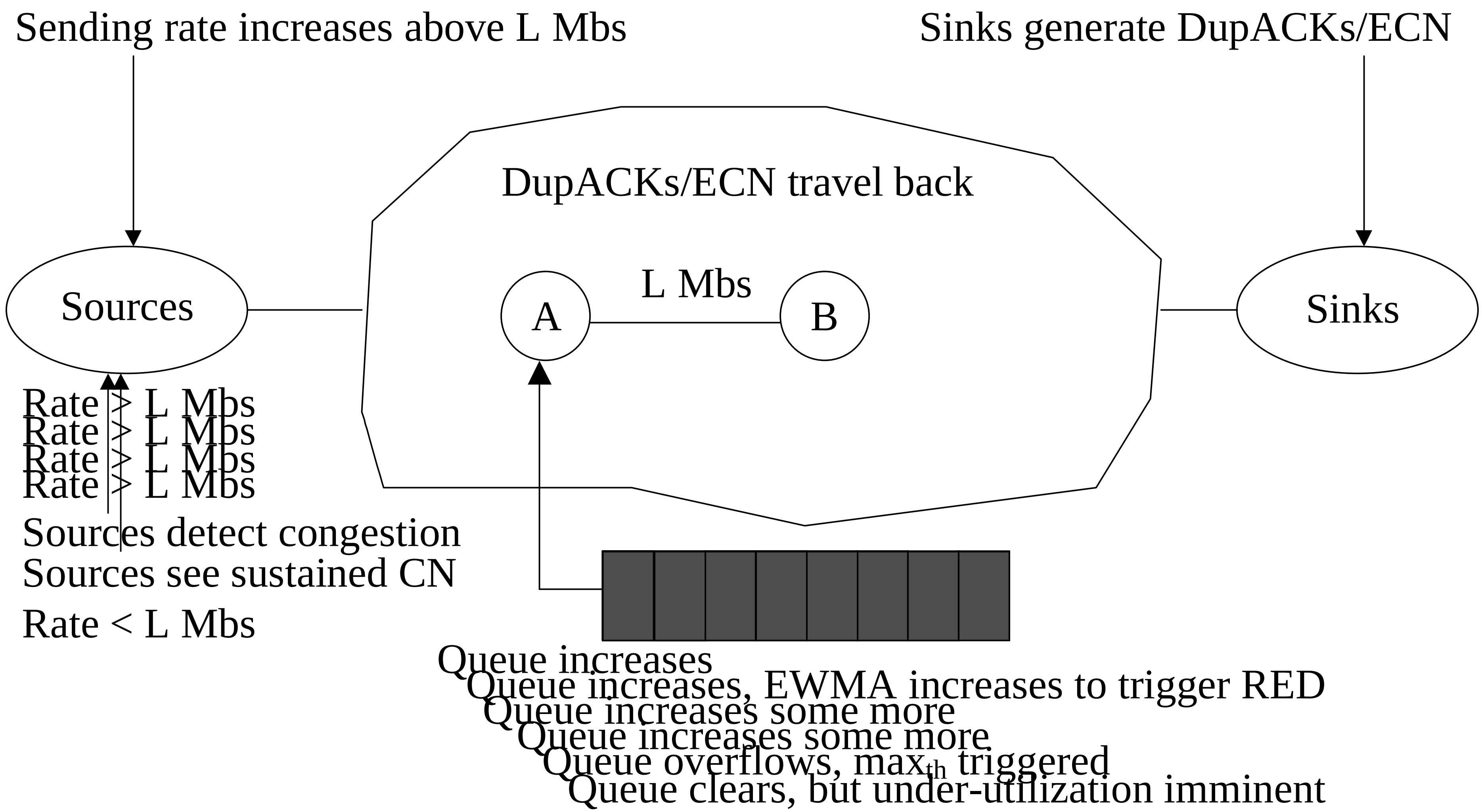
Outline

- Motivation
- Congestion control and queue management today (TCP, Drop-tail, RED)
- Solutions for reducing packet loss in the Internet
  - ECN
  - Adaptive RED
  - SubTCP
  - Blue
  - Stochastic Fair Blue
- Providing scalable QoS over the Internet
- Conclusion

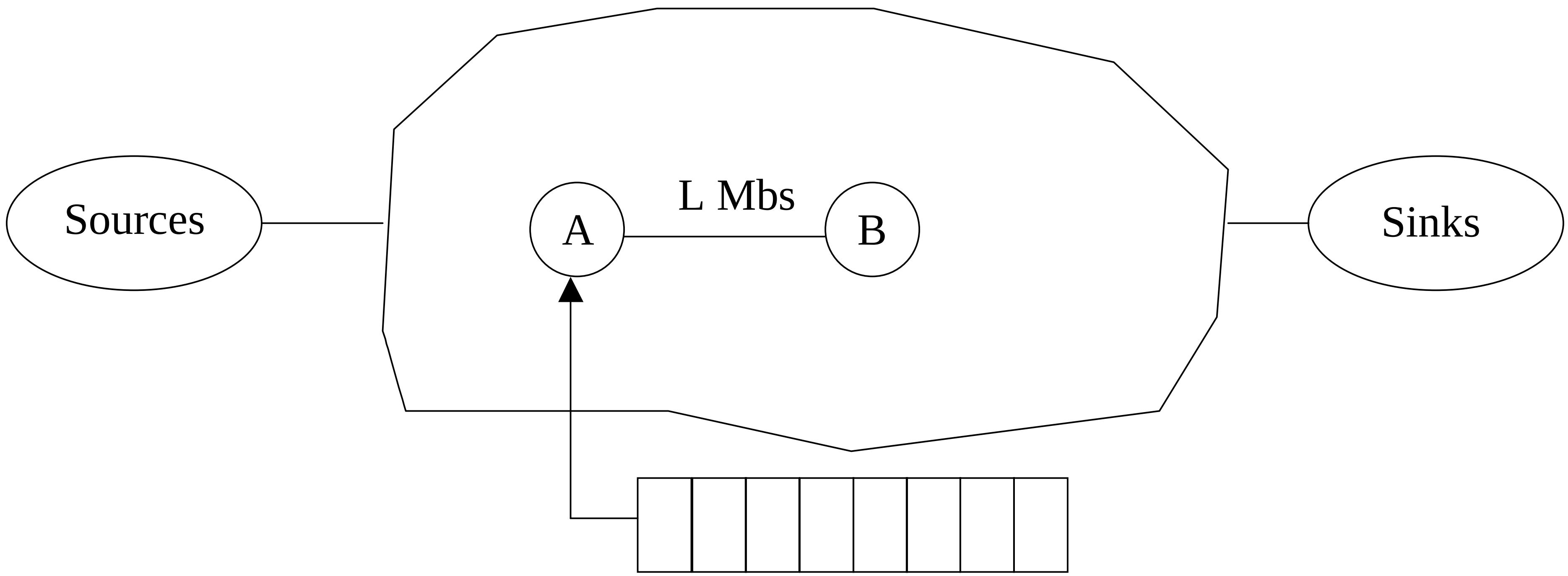
Blue

- RED
  - Queue length fluctuations
  - TCP modifications required (SubTCP)
  - Use of queue length inherently flawed
- Blue
  - Class of fundamentally different queue management algorithms
  - Decouple congestion management from queue length
  - Rely only on queue and link history
  - Example
    - Increase aggressiveness when loss rates high
    - Decrease aggressiveness when link underutilized

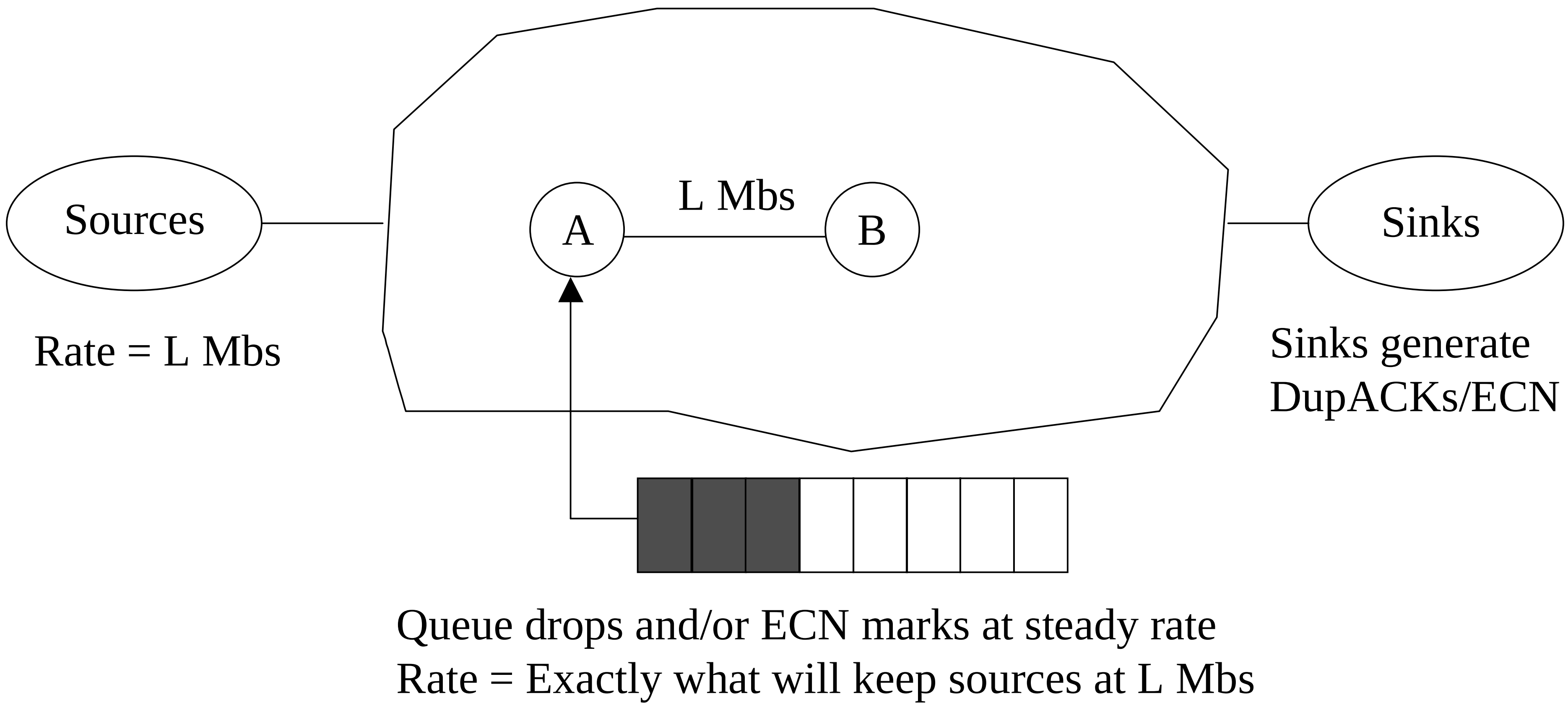
RED Example


Sending rate increases above L Mbs
Sinks generate DupACKs/ECN
DupACKs/ECN travel back
Sources
L Mbs
A
B
Sinks
Rate > L Mbs
Rate > L Mbs
Rate > L Mbs
Rate > L Mbs
Sources detect congestion
Sources see sustained CN
Rate < L Mbs
Queue increases
Queue increases, EWMA increases to trigger RED
Queue increases some more
Queue increases some more
Queue overflows, max_{th} triggered
Queue clears, but under-utilization imminent

RED Example



Ideal Example (Blue)



Example Blue Algorithm

- Single dropping/marking probability
  - Increase upon packet loss
  - Decrease when link underutilized
  - Freeze value upon changing

Upon packet loss:

if ((now - last_update) > freeze_time) then

$P_{mark} = P_{mark} + delta$

last_update = now

Upon link idle:

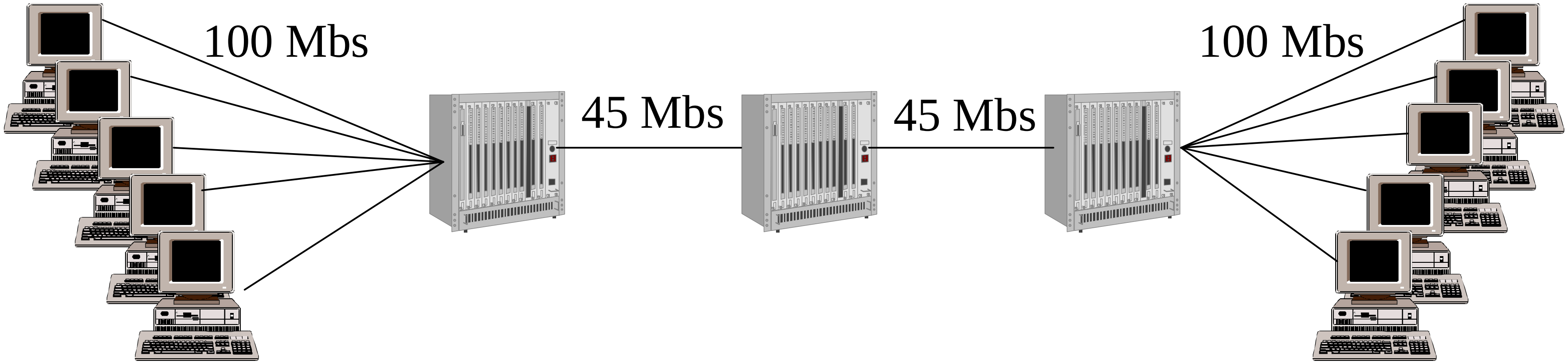
if ((now - last_update) > freeze_time) then

$P_{mark} = P_{mark} - delta$

last_update = now

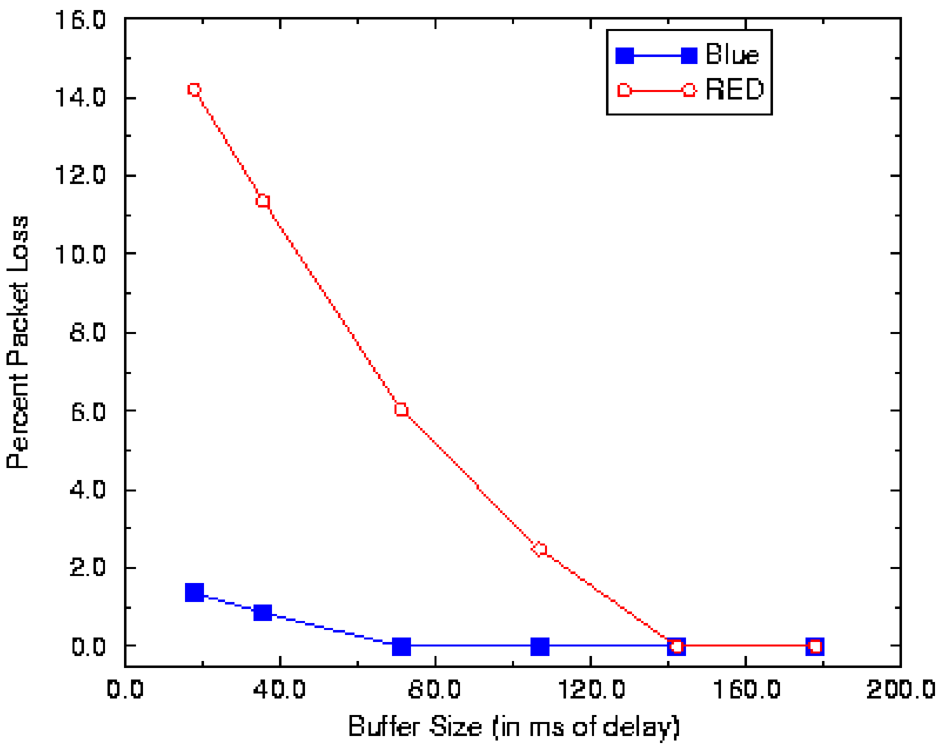
Blue Evaluation

- 400 and 1600 sources
- Buffer sizes at bottleneck link
  - From 100KB (17.8 ms)
  - Up to 1000KB (178 ms)

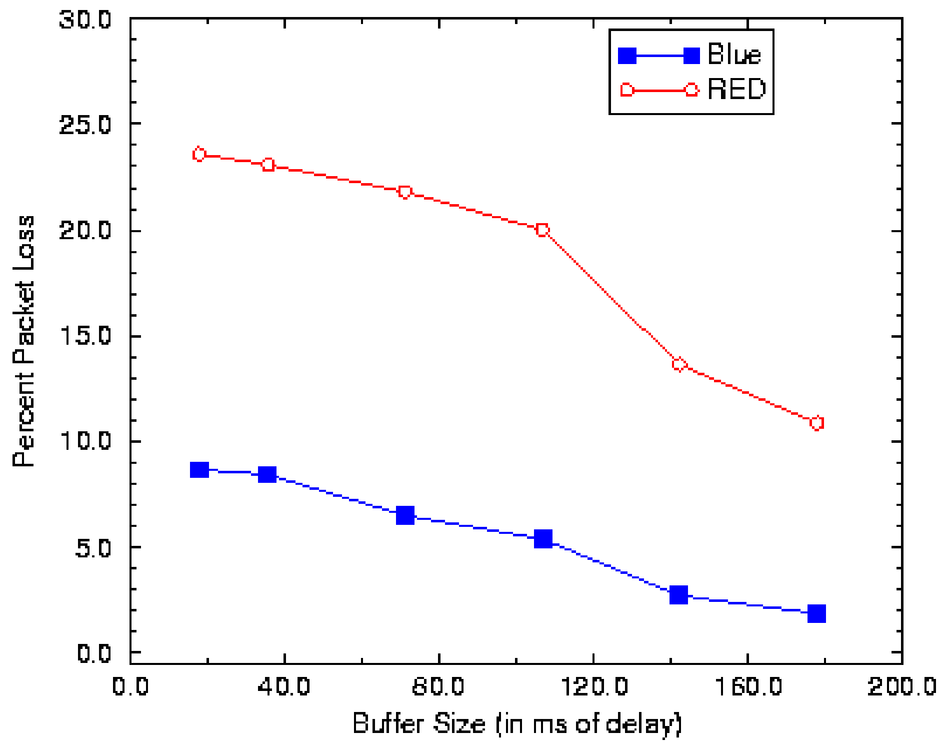


Blue Evaluation

400 sources



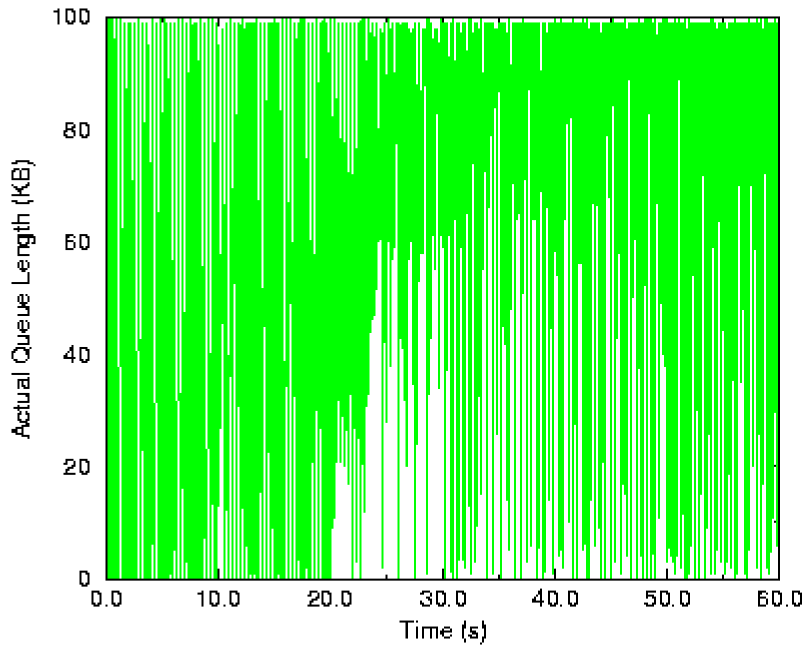
1600 sources



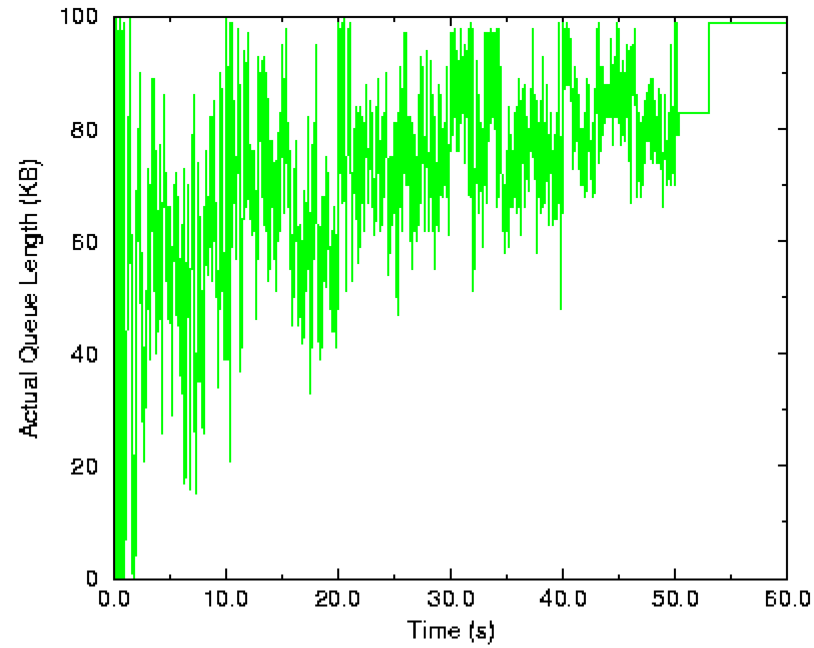
Understanding Blue

- Experiment
  - 50 sources added every 10 seconds
- Queue length plots

RED



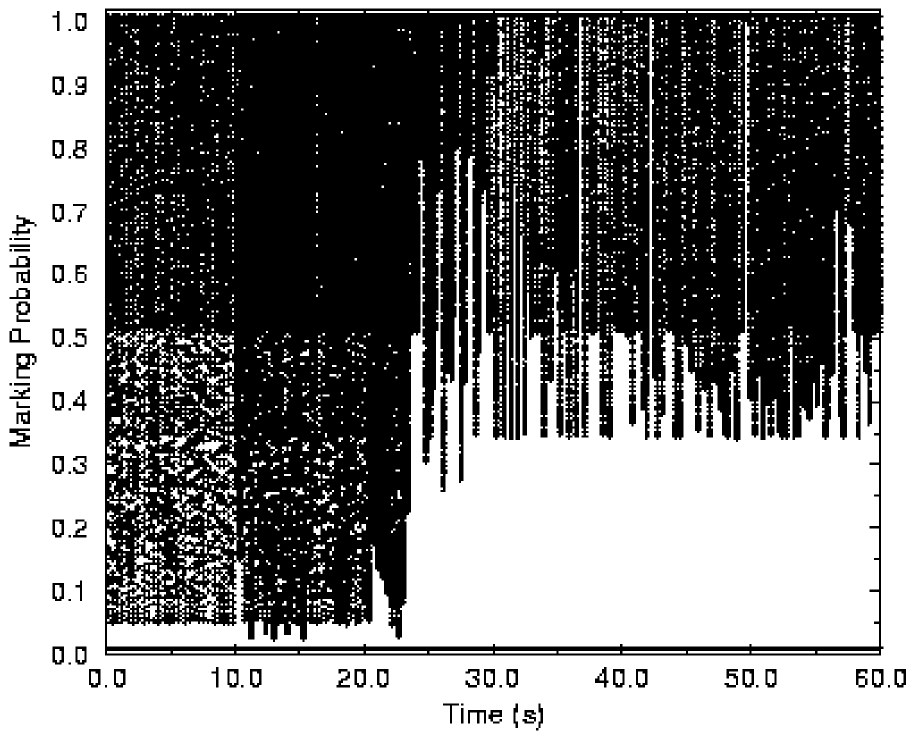
Blue



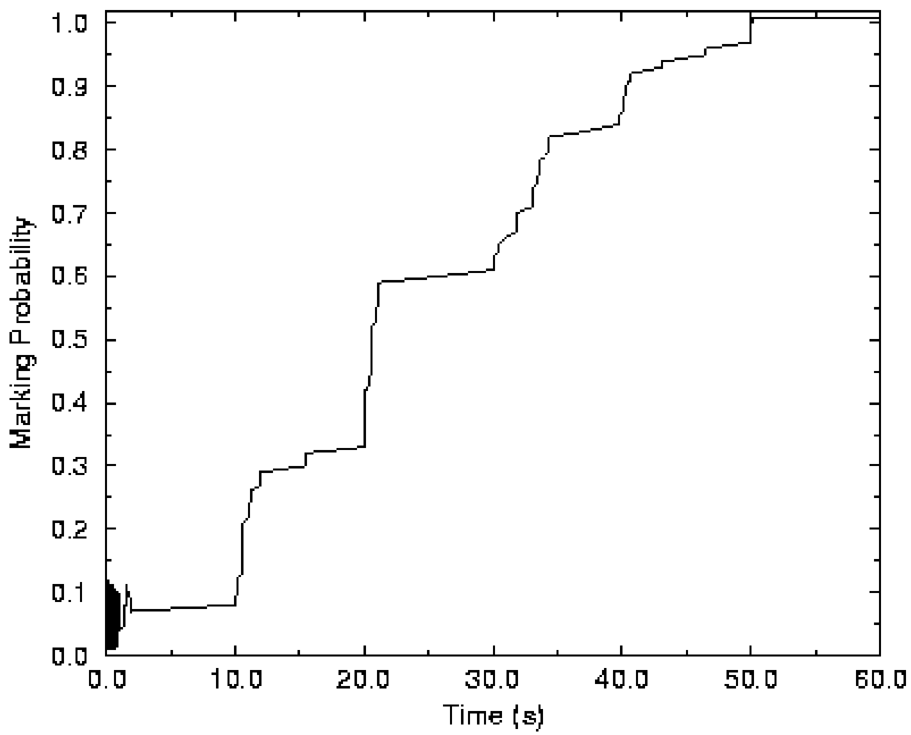
Understanding Blue

- Marking behavior

RED

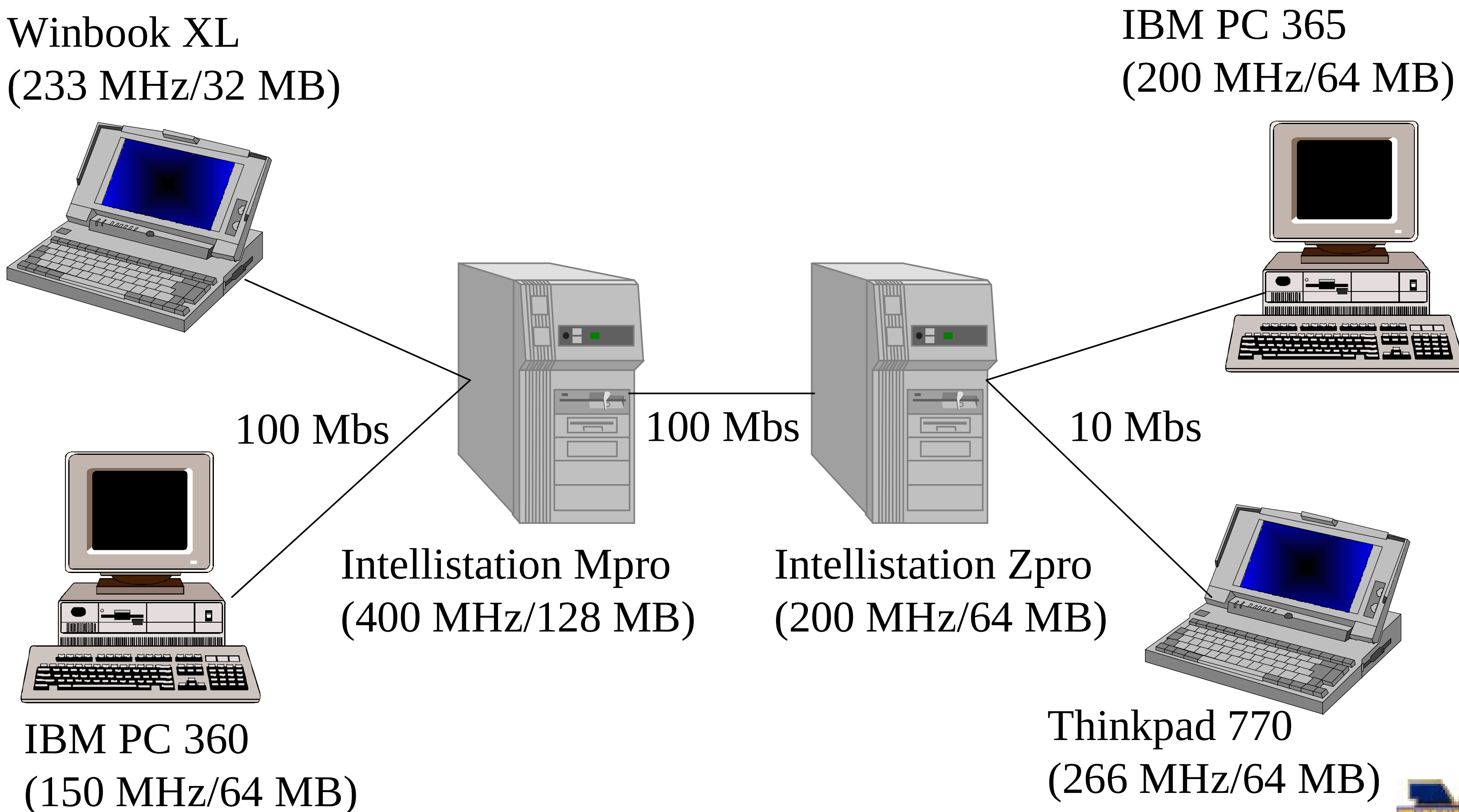


Blue



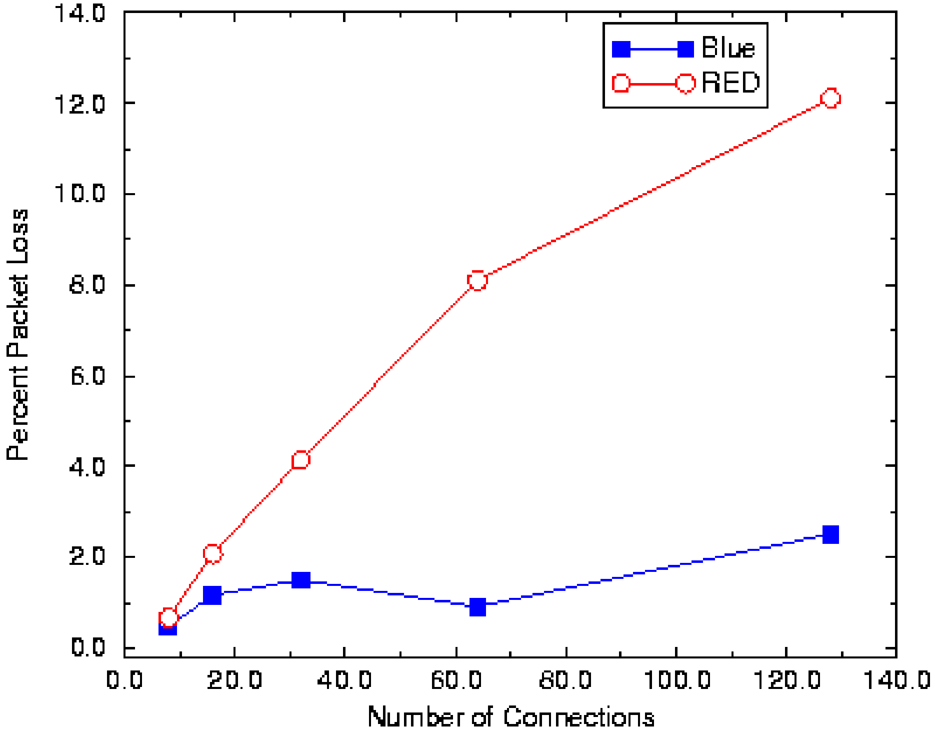
Implementation

- FreeBSD 2.2.7 + ALTQ

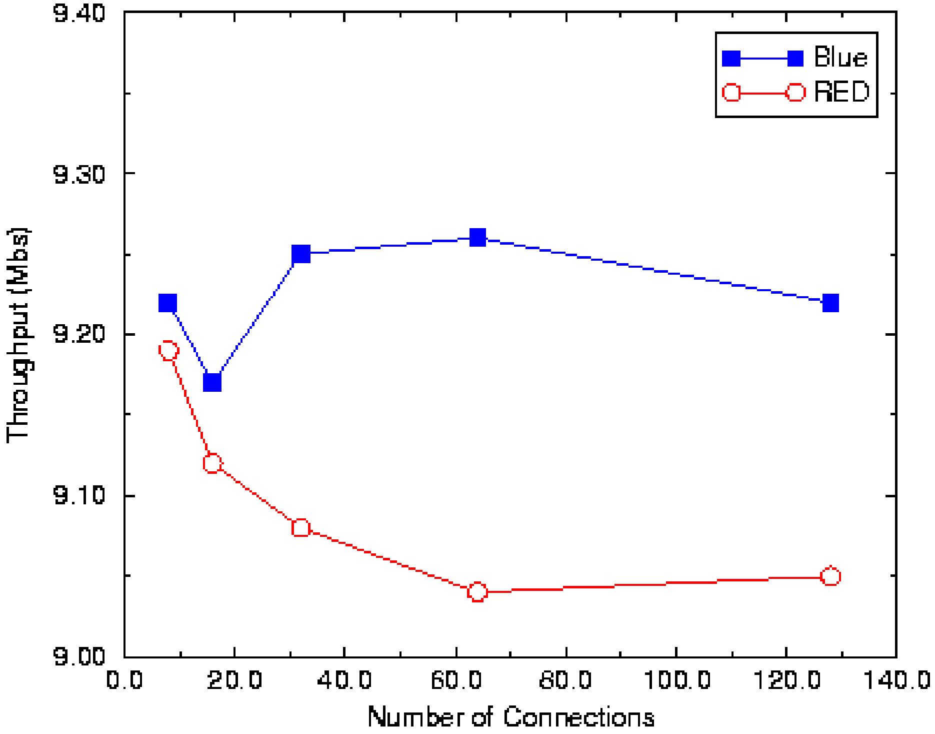


Blue Evaluation

Loss rates



Link utilization



Outline

- Motivation
- TCP, RED, and congestion control
- Solutions for reducing packet loss in the Internet
  - ECN
  - Adaptive RED
  - SubTCP
  - Blue
  - Stochastic Fair Blue
- Providing scalable QoS over the Internet
- Conclusion

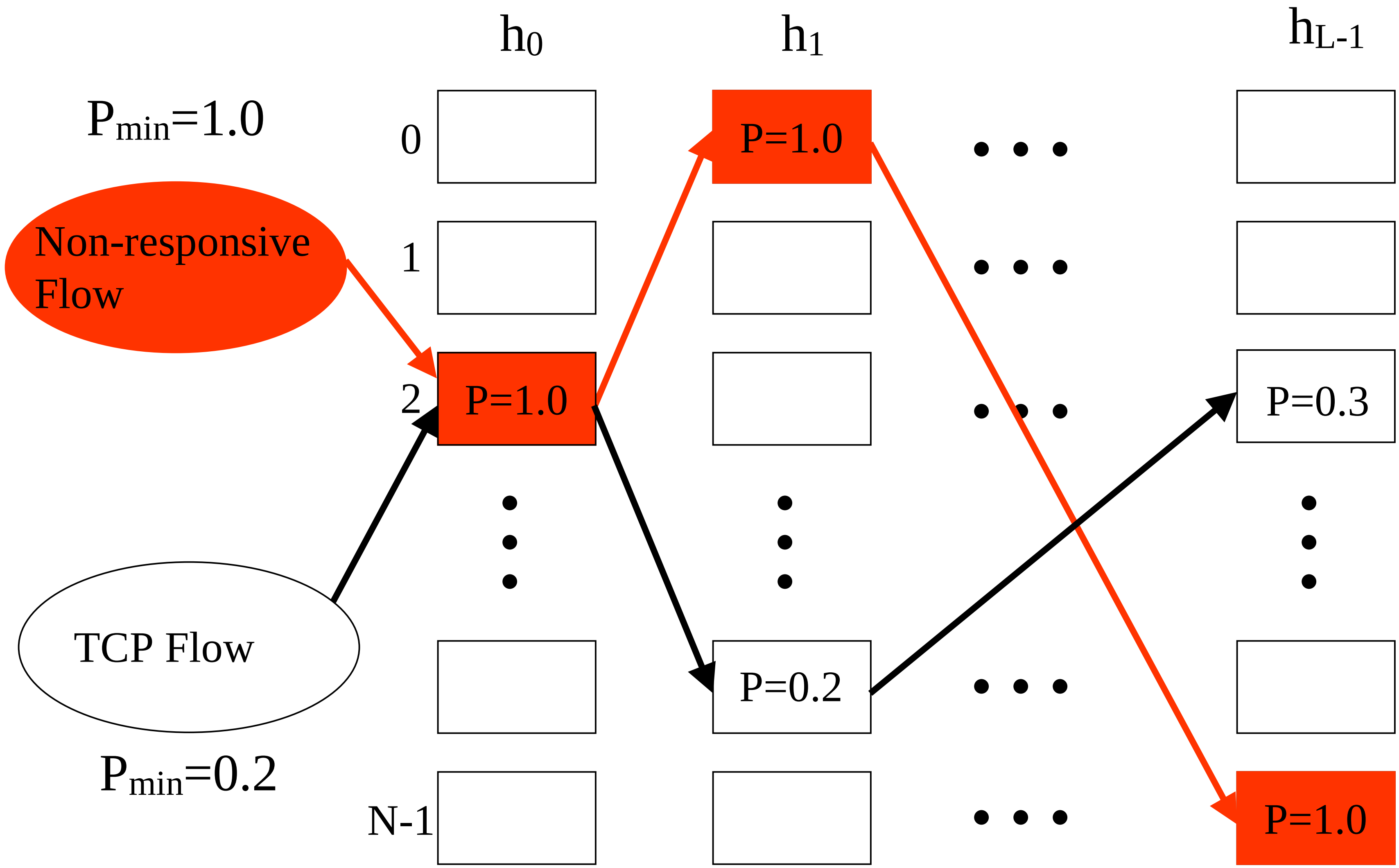
Dealing with Non-responsive Flows

- Fair queuing
  - WFQ, W2FQ [Bennett96], Virtual Clock[Zhang90], SCFQ [Golestani94], STFQ [Goyal96]
  - Stochastic Fair Queuing [McKenney90]
  - Problems
    - Overhead
    - Partitioned buffers
- Buffer management
  - RED with penalty box [Floyd97], Flow RED [Lin97]
  - Problems:
    - Buffer space requirements
    - Inaccuracy

Stochastic Fair Blue (SFB)

- Single FIFO queue
- Multiple independent hash functions applied to each packet
- Packets update multiple accounting bins
- Blue performed on accounting bins
- Observation
  - Non-responsive flows drive P to 1.0 in all bins
  - TCP flows have some bins with normal P
  - $P_{min} = 1.0$, rate-limit
  - $P_{min} < 1.0$, mark with probability P_{min}

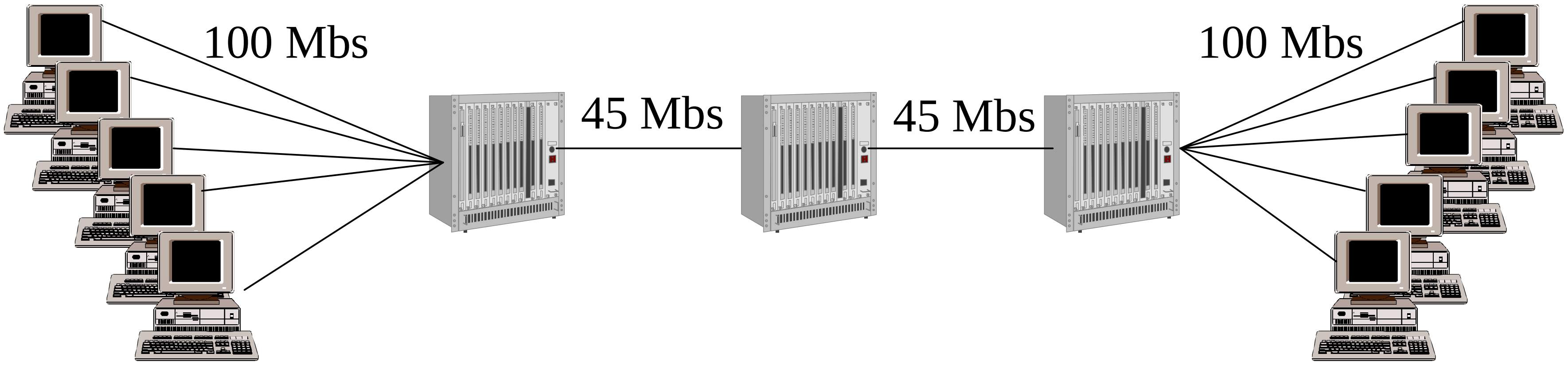
SFB



N^L virtual bins out of L*N actual bins

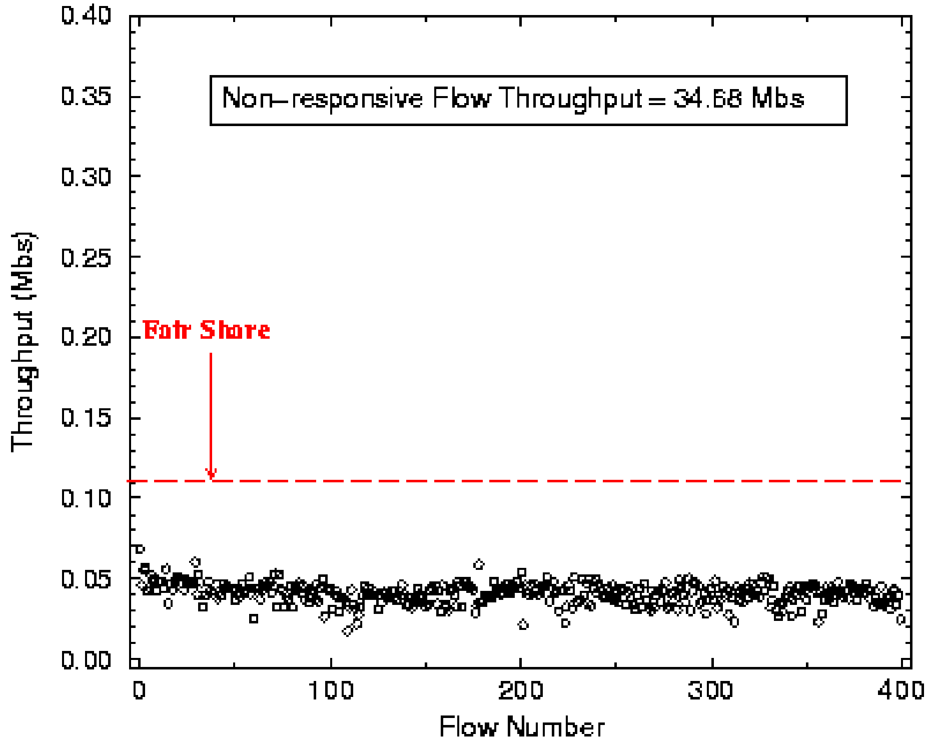
SFB Evaluation

- 400 TCP flows
- 1 non-responsive flow sending at 45 Mbs
- Evaluation
  - 200KB, 2-level SFB with 23 bins per level (529 virtual bins)
  - 200KB RED queue
  - 400KB SFQ with 46 RED queues



SFB Evaluation

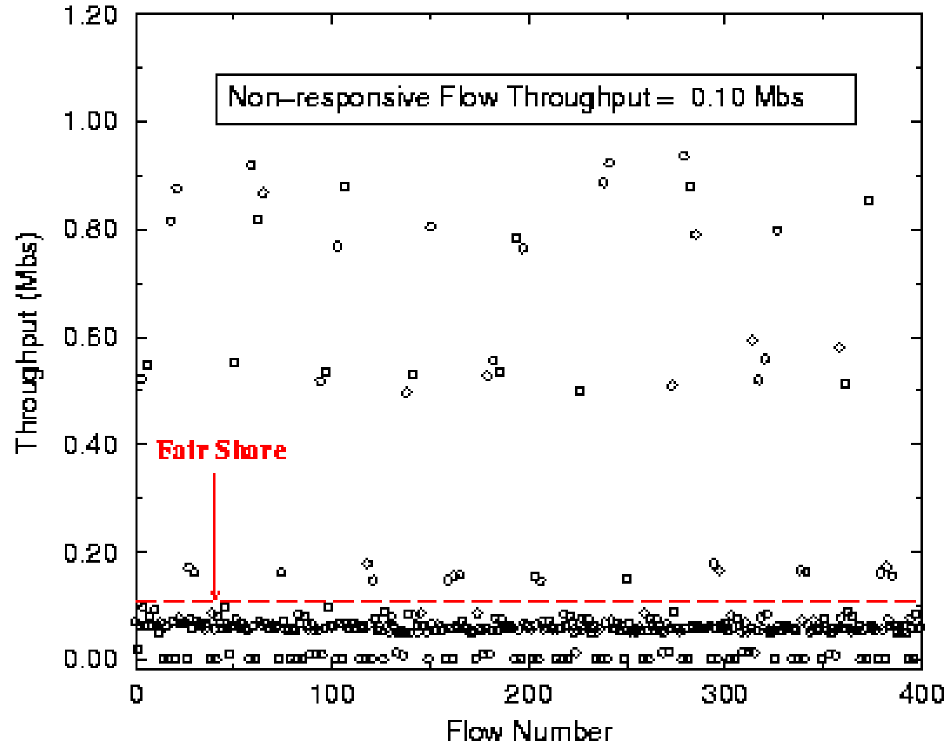
RED



Loss rates

TCP Flows = 3.07 Mbs
Non-responsive = 10.32 Mbs

SFQ+RED

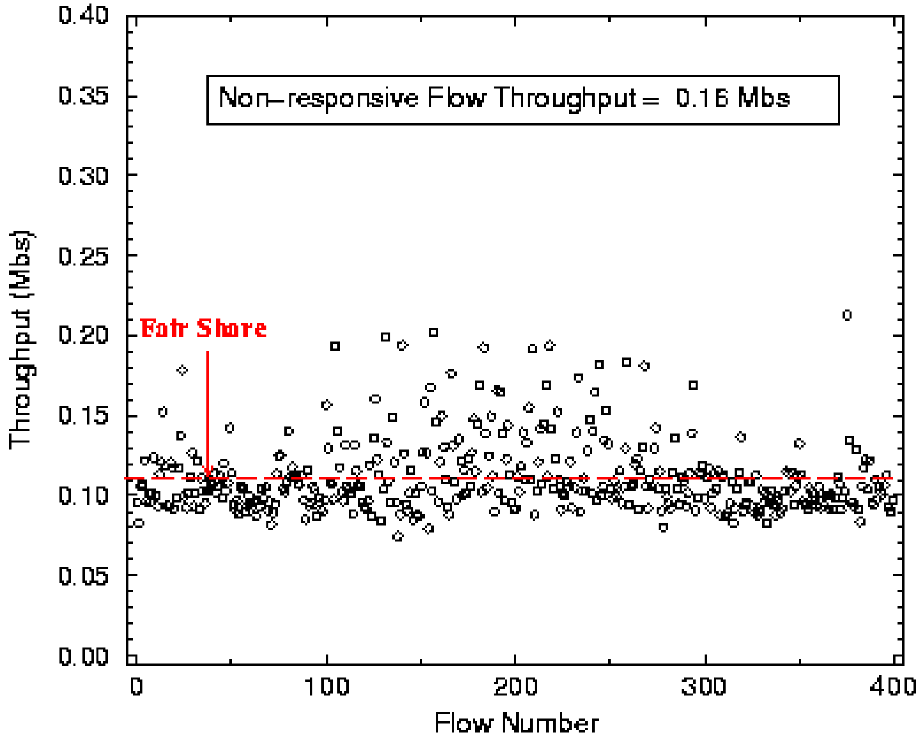


Loss rates

TCP Flows = 2.53 Mbs
Non-responsive = 43.94 Mbs

SFB Evaluation

SFB



<u>Loss rates</u>

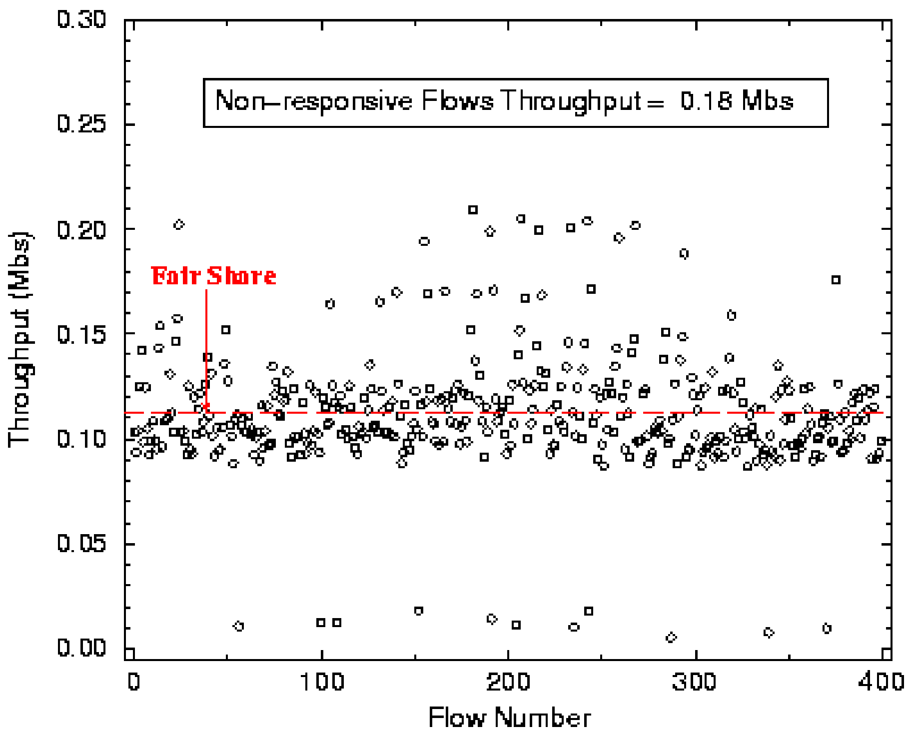
TCP Flows = 0.01 Mbs
Non-responsive = 44.84 Mbs

SFB and Misclassification

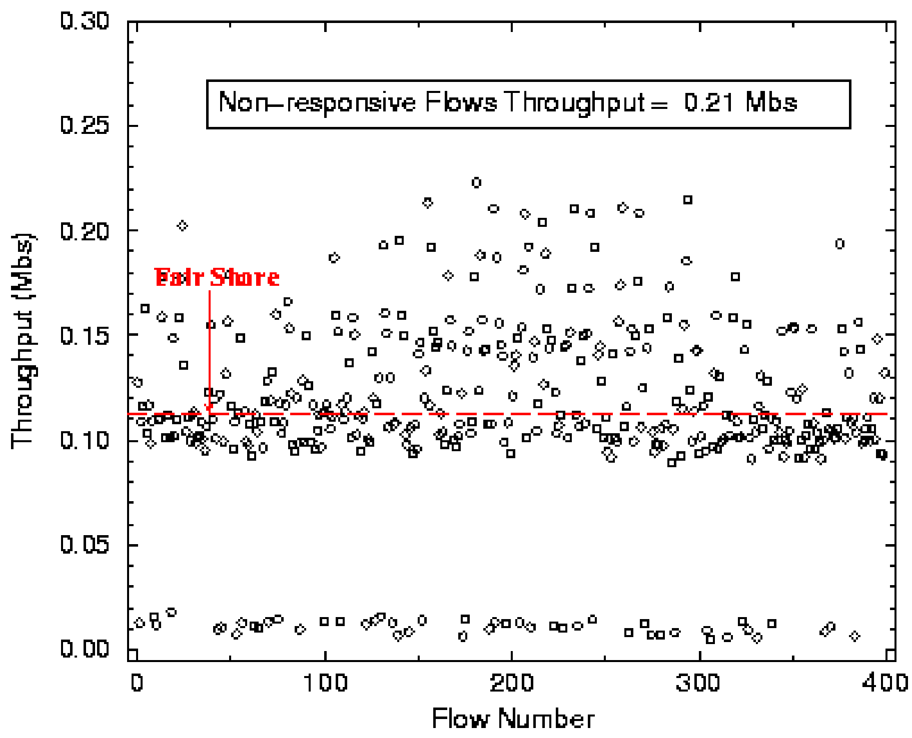
- SFB deteriorates with increasing non-responsive flows
- Non-responsive flows pollute bins in each level
- Probability of misclassification
  - $p = [1 - (1 - 1/N)^M]^L$
  - Given M, optimize L and N subject to L*N=C

SFB and Misclassification

4 non-responsive flows



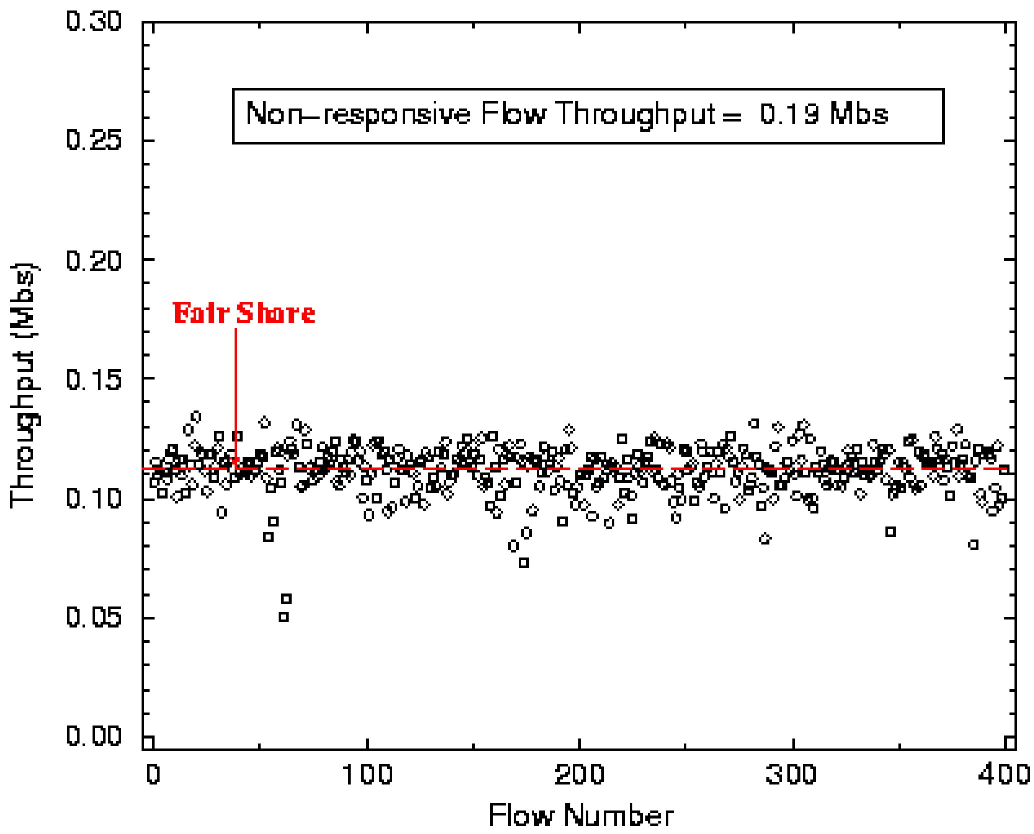
8 non-responsive flows



SFB with Moving Hash Functions

- SFB
  - Virtual buckets from spatial replication of bins
- Moving hash functions
  - Virtual buckets temporally
- Advantages
  - Handles misclassification
  - Handles reformed flows

SFB with Moving Hash Functions



Outline

- Motivation
- TCP, RED, and congestion control
- Solutions for reducing packet loss in the Internet
  - ECN
  - Adaptive RED
  - SubTCP
  - Blue
  - Stochastic Fair Blue
- Providing scalable QoS over the Internet
- Conclusion

Scalable QoS over the Internet

- One of the first papers on Differentiated Services
- Led to formation of current working group
- Contributions
  - Fundamental problems with TCP over DiffServ
  - Modifications for improving performance
  - Architecture for providing soft bandwidth guarantees
  - Novel, end-host marking mechanisms
  - Influence in IETF (AF I-D and DiffServ WG)
  - Influence in industry (Cisco)

Conclusion

- Maximizing network efficiency
  - De-coupling packet loss and congestion notification (ECN)
  - Adaptive queue management (Adaptive RED and Blue)
  - Intelligent end-host mechanisms (SubTCP)
  - Scalable protection against non-responsive flows (SFB)
- QoS through Differentiated Services

Publications

- "Understanding TCP Dynamics in an Integrated Services Internet"
  - NOSSDAV 1997
  - IEEE/ACM Transactions on Networking 1999.
- "Adaptive Packet Marking for Providing Differentiated Services in the Internet"
  - ICNP 1998
  - Accepted IEEE/ACM Transactions on Networking 1999 (minor revisions).
- "A Self-Configuring RED Gateway"
  - INFOCOM 1999
- "Blue: A New Class of Active Queue Management Algorithms"
  - ?