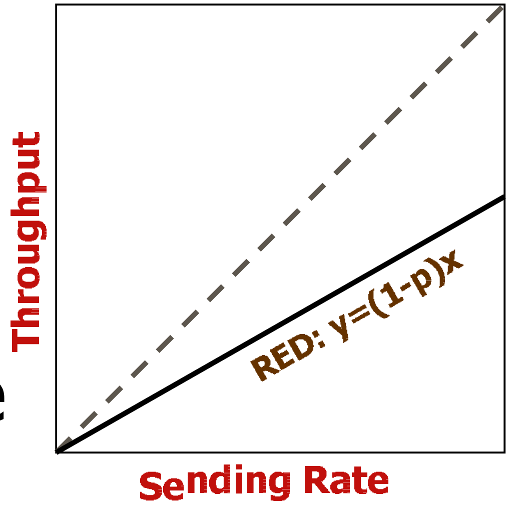
Controlling High-Bandwidth Flows at the Congested Router

Ratul Mahajan[†]
Sally Floyd[‡]
David Wetherall[†]

[†]University of Washington
[‡]AT&T Center for Internet Research at ICSI (ACIRI)

Problem

- Without flow based differentiation all flows see the same drop rate
  - flows get more by sending more
- High-bandwidth flows increase the drop rate at the router
  - short RTT TCP flows, unresponsive flows



p: drop rate

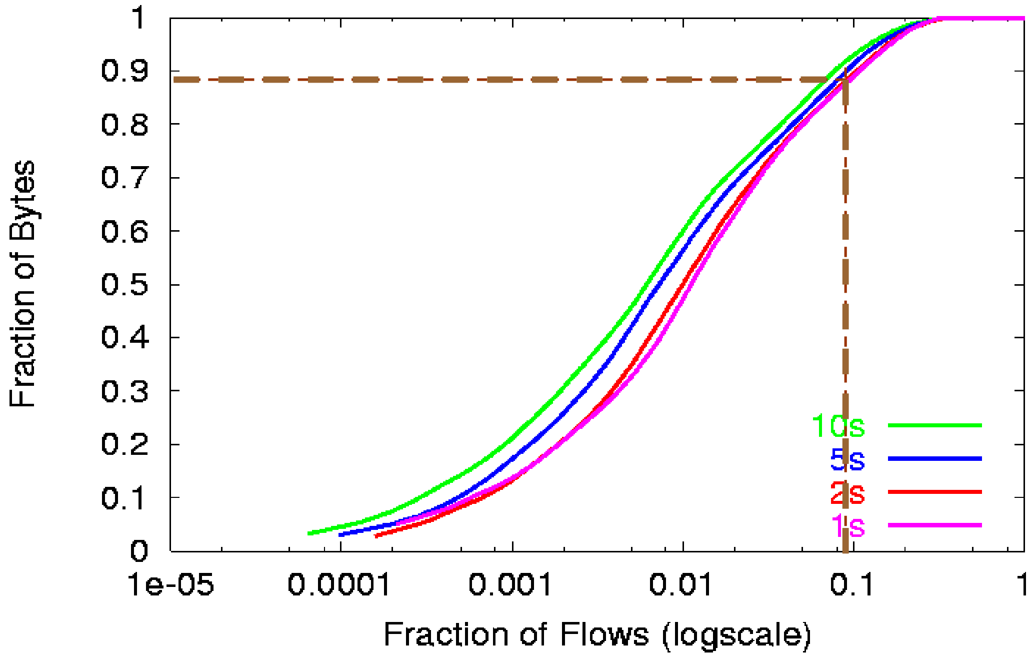
Need router mechanisms to protect rest of the traffic from high-bandwidth flows

Goal: Simple Protection

- Protection
  - from high-bandwidth flows
  - examples: per-flow state approaches
- Simplicity
  - single FIFO queue, simple fast path operations
- Approach: Partial Flow State
  - state for the high-bandwidth flows only

Why Partial Flow State Approach Works

What fraction of flows get what fraction of bytes over different time windows.



- Bandwidth distribution is skewed - a small fraction of flows accounts for most of the bandwidth.
- Bandwidth consumption is predictive - high-bandwidth flows remain high-bandwidth (not in graph)

RED with Preferential Dropping (RED-PD): Overview

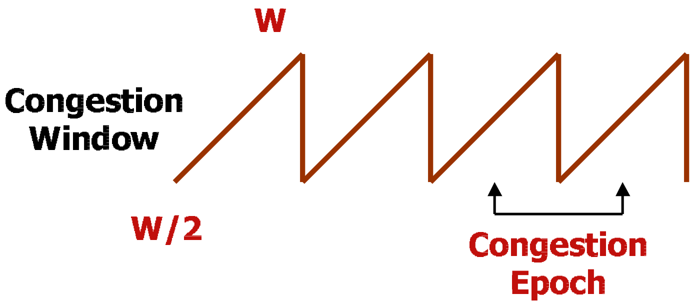
- Identify high-bandwidth flows during times of congestion
  - called *monitored* flows
  - use drop history

- Restrict the throughput of monitored flows
  - use preferential dropping

Defining "High Bandwidth"

- Pick a round trip time (RTT) *R*
- High bandwidth:
  - more than a TCP flow with RTT *R*
    - $\frac{\sqrt{1.5}}{R\sqrt{p}}$ [FF99]
- Function of drop rate *p* at the router

Identification (Theory)



- TCP suffers one drop in a congestion epoch
  - *CELength(R,p)* = $\frac{R}{\sqrt{1.5}\,\sqrt{p}}$

- Identify flows with one or more drops in *CELength(R,p)* seconds
  - Flows that send more suffer more drops

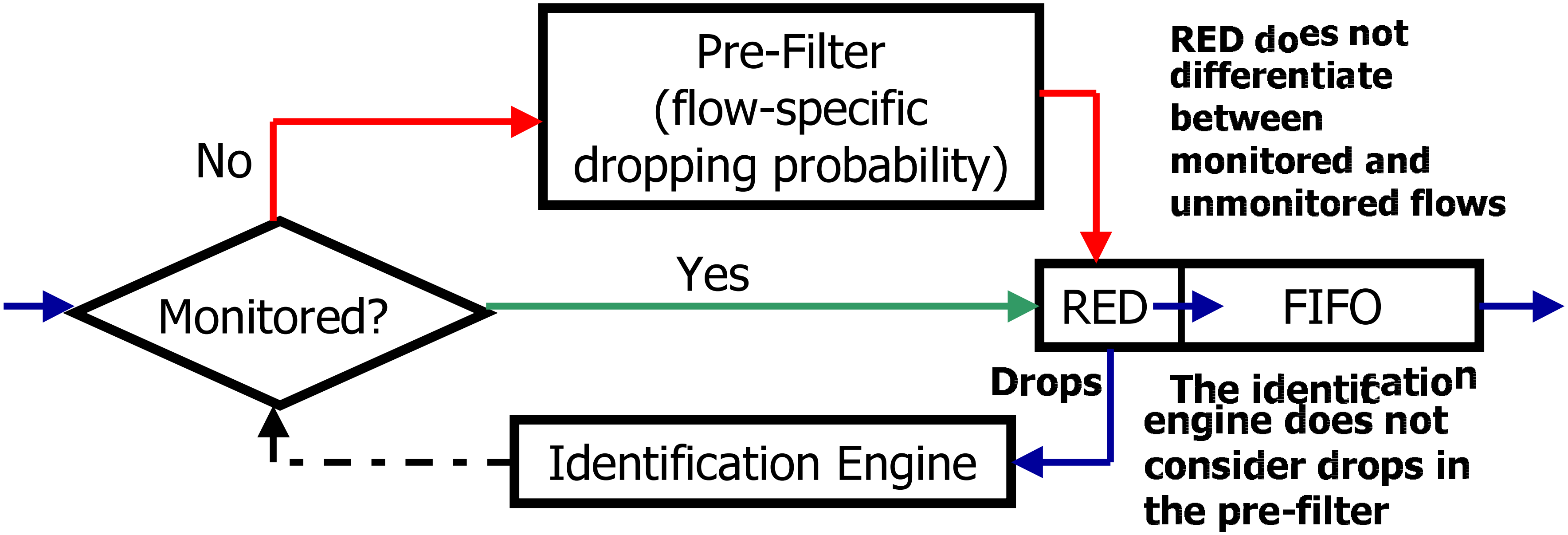
Identification (Practice)

- Flows suffer occasional drops
  - keep the drop history of *K* congestion epochs
- Multiple losses in a window of data
  - consider loss *events* by breaking down the drop history into *M (>K)* lists
- Identify flows with drops spread over *K* or more lists

Controlling High-Bandwidth Flows

- Preferential dropping
  - lightweight mechanism to restrict the throughput of identified flows
  - probabilistically drop packets from the flow before it enters the output queue
- What should the dropping probability be?
  - the flow should not be "high-bandwidth" when it enters the output queue

Architecture

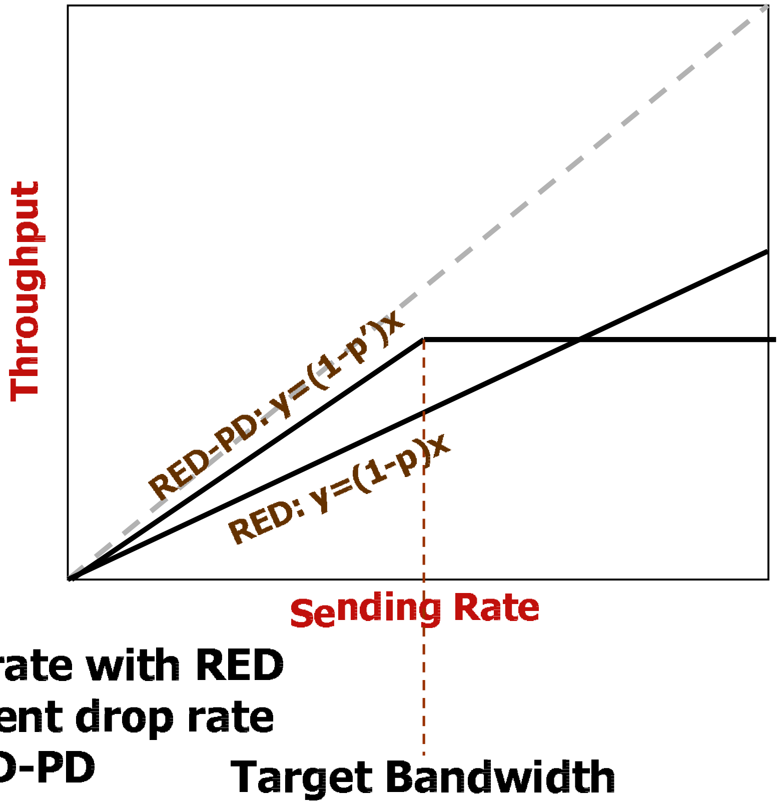


RED does not differentiate between monitored and unmonitored flows

The identification engine does not consider drops in the pre-filter

- If a monitored flow is identified again, increase the dropping probability
  - increase amount is a function of RED drop rate and excess rate of the flow
- If a monitored flow is suffering too few drops, decrease the dropping probability

Effect of RED-PD



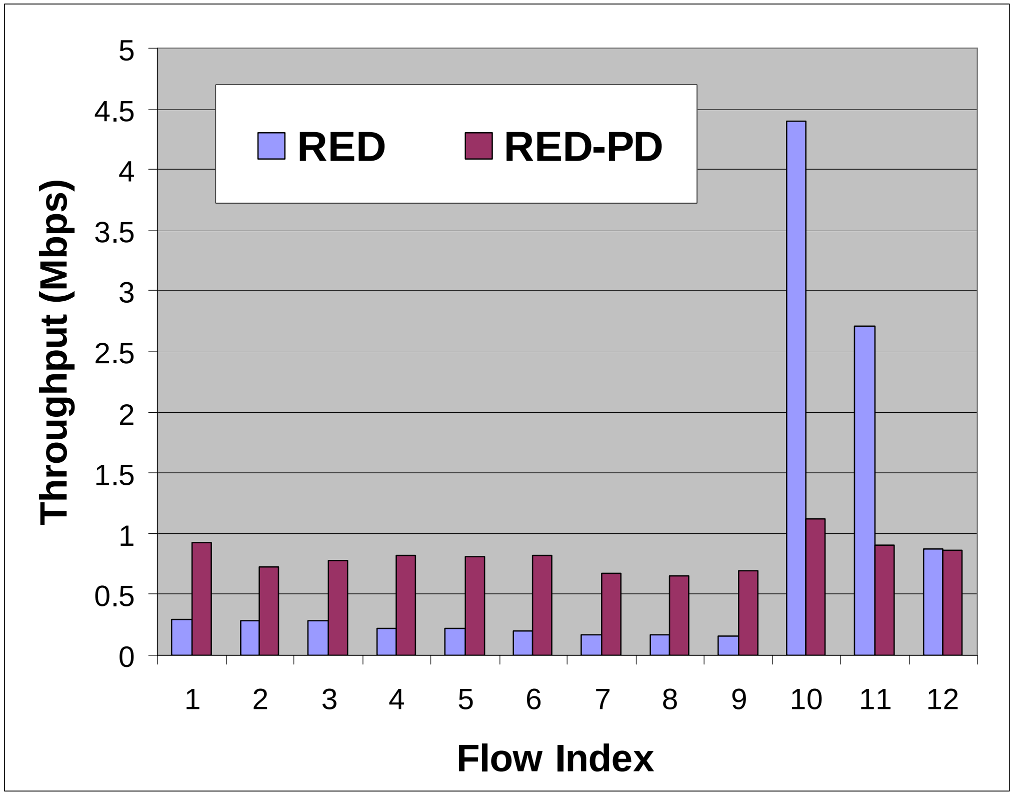
p: drop rate with RED
p': ambient drop rate with RED-PD

- Reduction in ambient drop rate (p′ < p)
- Full max-min fair in the extreme case (p′ = 0)

Evaluation

- Fairness
- Effect of target RTT *R*
- Response time
- Probability of identification
- Persistent congestion throughput
- Web traffic
- Multiple congested links
- TFRC
- Byte mode operation

Fairness

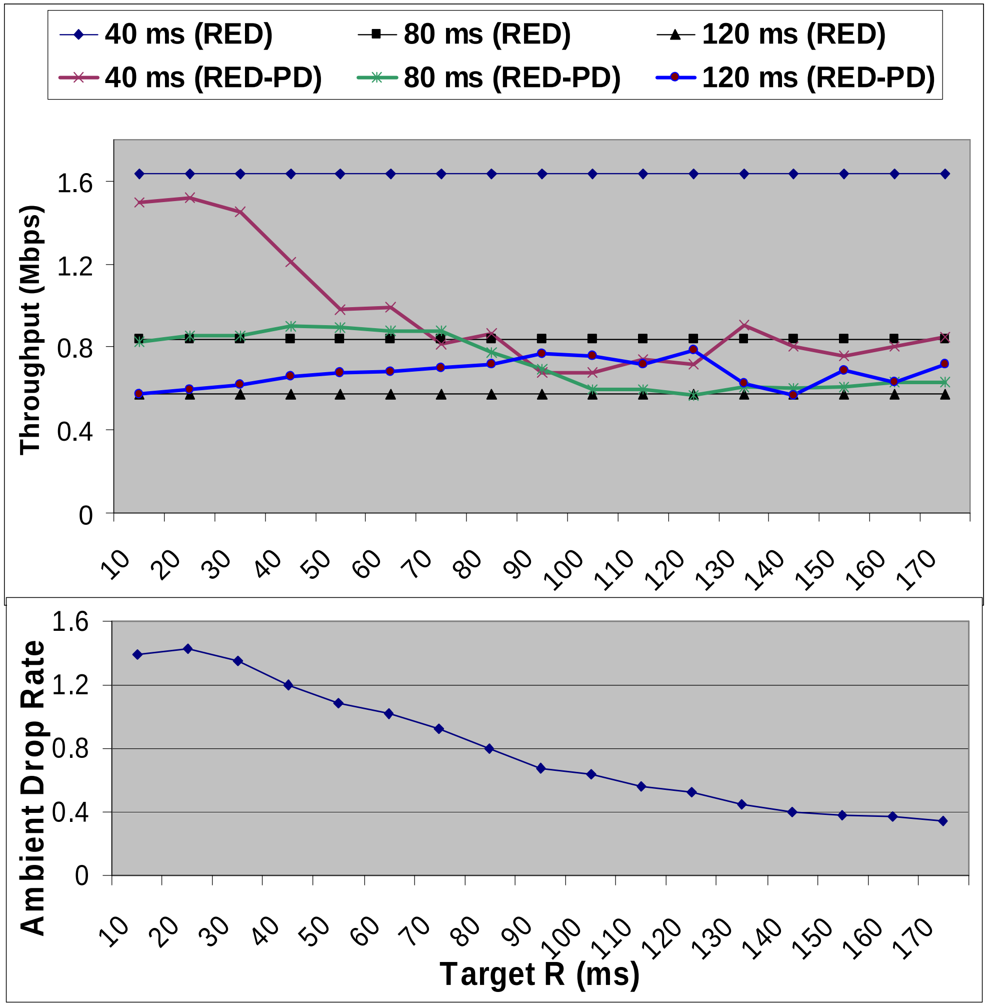


10 Mbps link
R= 40ms
Flow Index

1-3	30ms TCP
4-6	50ms TCP
7-9	70ms TCP
10	5 Mbps CBR
11	3 Mbps CBR
12	1 Mbps CBR

RED-PD’s iterative probability changes successfully approximate fairness

Effect of target RTT *R*

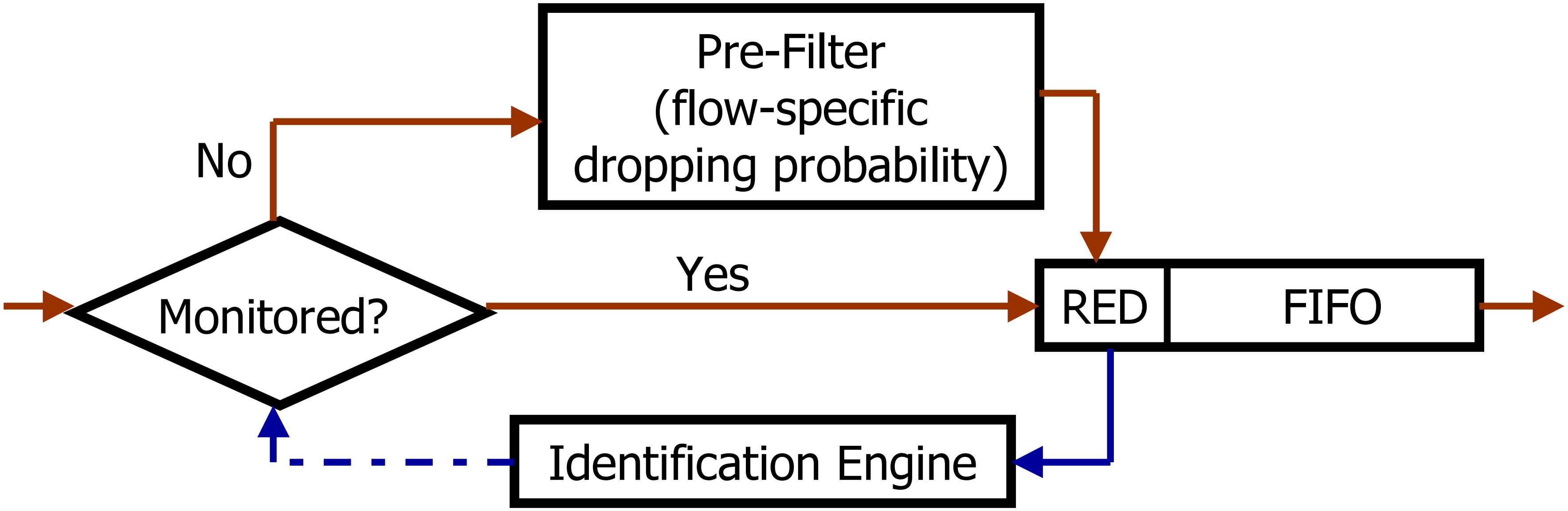


10 Mbps link
14 TCP flows
2 each of RTT 40, 80 &
120 ms
8 of RTT 160 ms

Increasing *R*

- increases fairness
- increases state
- decreases ambient drop rate

Implementation Complexity



- Identification engine
  - state for drop history; not in fast forwarding path
- Fast-path operations
  - lookup and probabilistic drop for a small fraction of flows

Conclusions

- Need router mechanisms to protect against high drop rates caused by high-bandwidth flows
- Skewed bandwidth consumption can be leveraged to provide lightweight protection
- RED-PD combines simplicity and protection
- Provides a knob to tune the degree of fairness