PIE: A Lightweight Scheme to Address the Bufferbloat Problem

- Further Studies (March 12, 2013)

Rong Pan, Preethi Natarajan, Chiara Piglione, Mythili S. Prabhu,
Vijay Subramanian, Fred Baker and Bill Ver Steeg

Outline

- Recap: the PIE Design
- Applying PIE in the Data Center Environment
- Adjusting Delay Reference to Accommodate Changing Network or Traffic Scenarios
- PIE on top of the Fair Queueing structure: FQ_PIE
- Future Work

The design of PIE

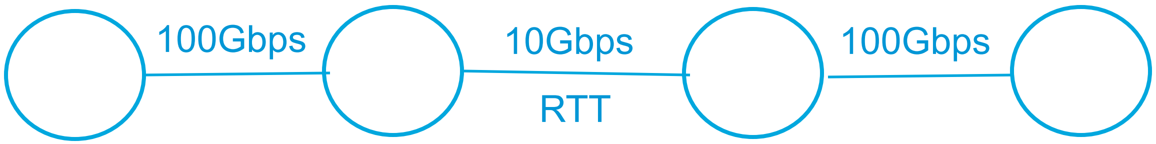
- Upon every packet departure
  - depart_count += deque_packet_size;
  - if dep_threshold is acrossed
    - dep_rate = dep_count/(now-start)
    - start = now
    - depart_count = 0;
- Every T_{update} interval
  - estimated_delay, est_del = queue_length/dep_rate
  - drop_prob += alpha*(est_del – target_delay) + beta* (est_del – est_del_old)
  - est_del_old = est_del;
- Upon every packet arrival
  - randomly drop a packet based on drop_prob

alpha and beta are chosen via control analysis

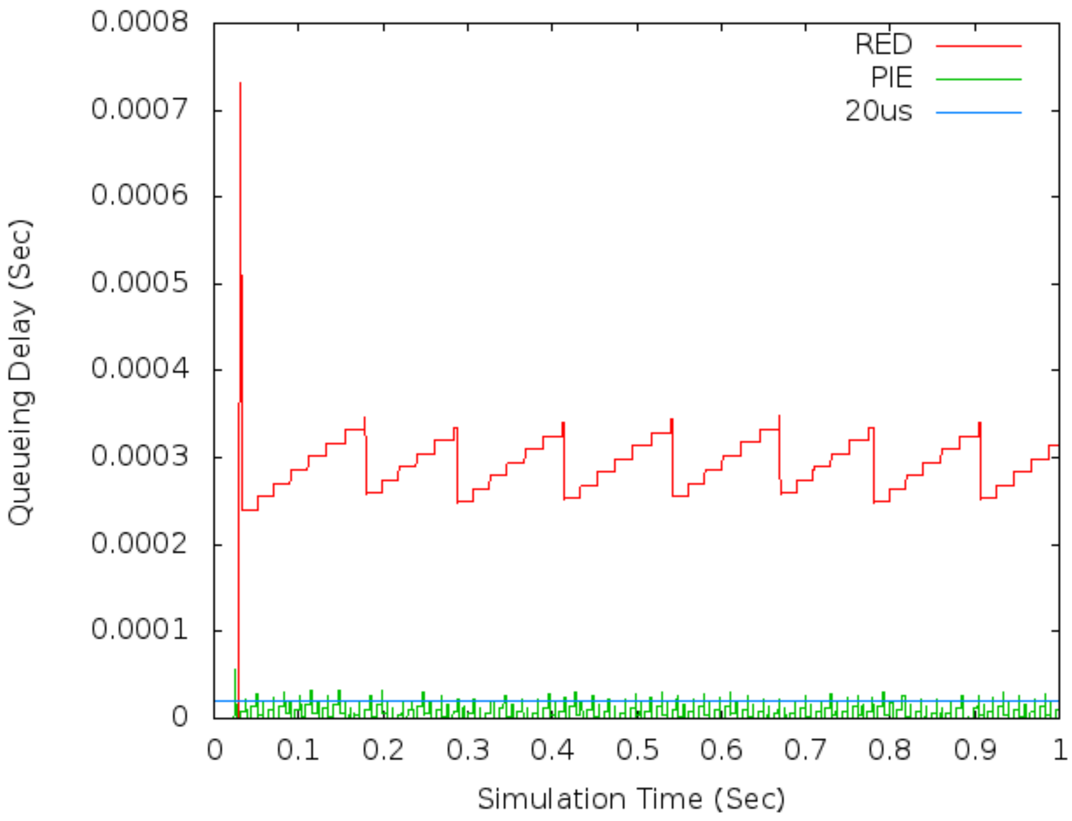
PIE for Data Centers

Simulation Setup

- alpha = 25; beta = 250, Tupdate = 100us, Del_Ref = 20us;
- Congestested Link Bw: 10Gbps
- Avg Pkt Size: 1000B
- TCP: Cubic



Simulation RED+ECN vs. PIE+ECN - 1 Cubic TCP Flow

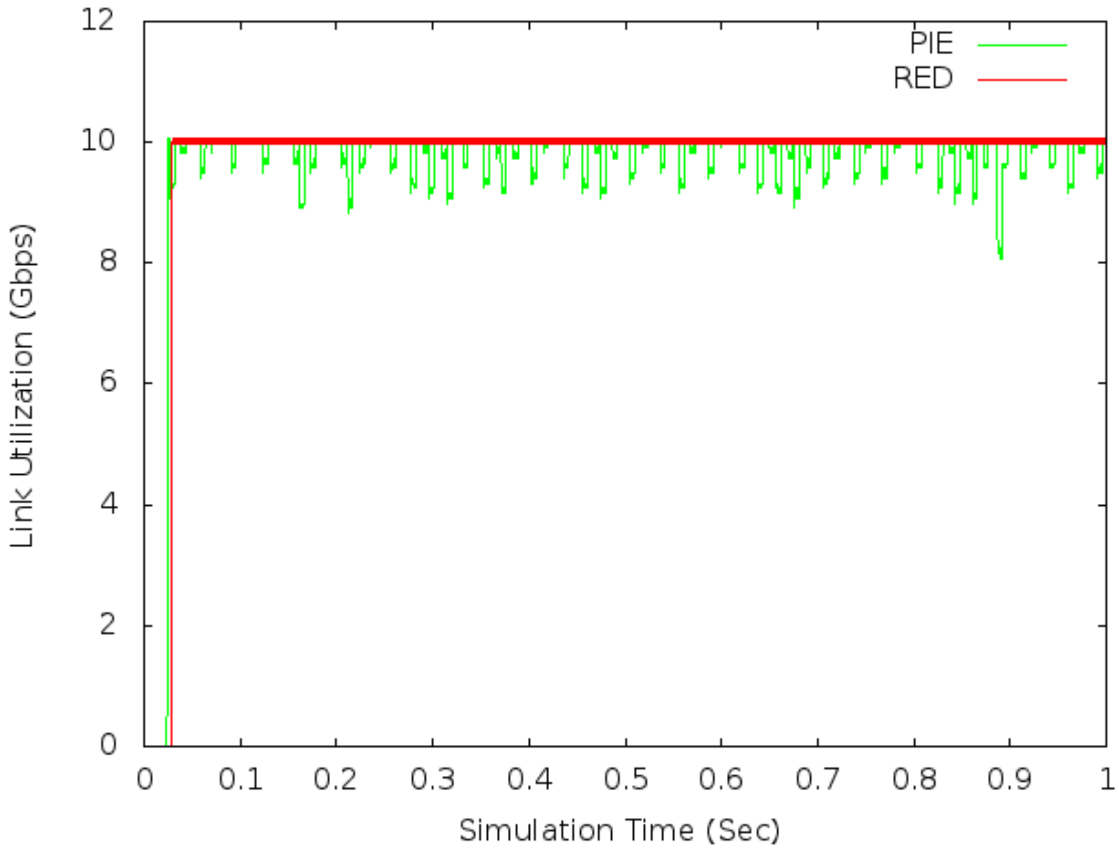


RTT = 100us
BufferLimit = 2000 packets
$\text{Min}_{th}/\text{Max}_{th}$ = 20%/50%

Under RED, ECN marks won't occur until the queue size is across the minimum threshold, which is around 400 packets and its corresponding delay is around 320us.

PIE automatically calculates the marking probability based on how latency and latency trend moves. As a result, the queueing latency is stabilized around the reference value, 20us.

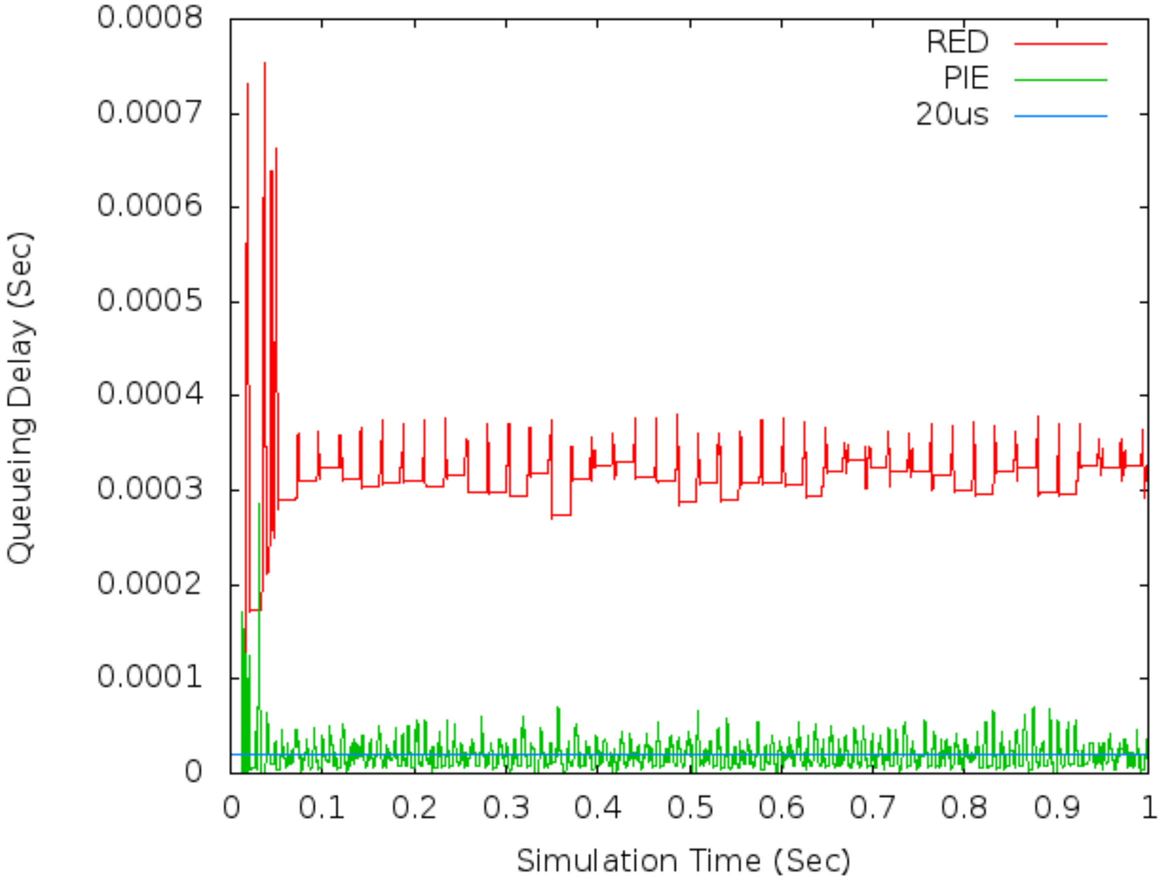
Simulation PIE+ECN
- 1 Cubic TCP Flow



By controlling the latency to be low, PIE loses throughput slightly. But the throughput is still high with 9.84Gbps.

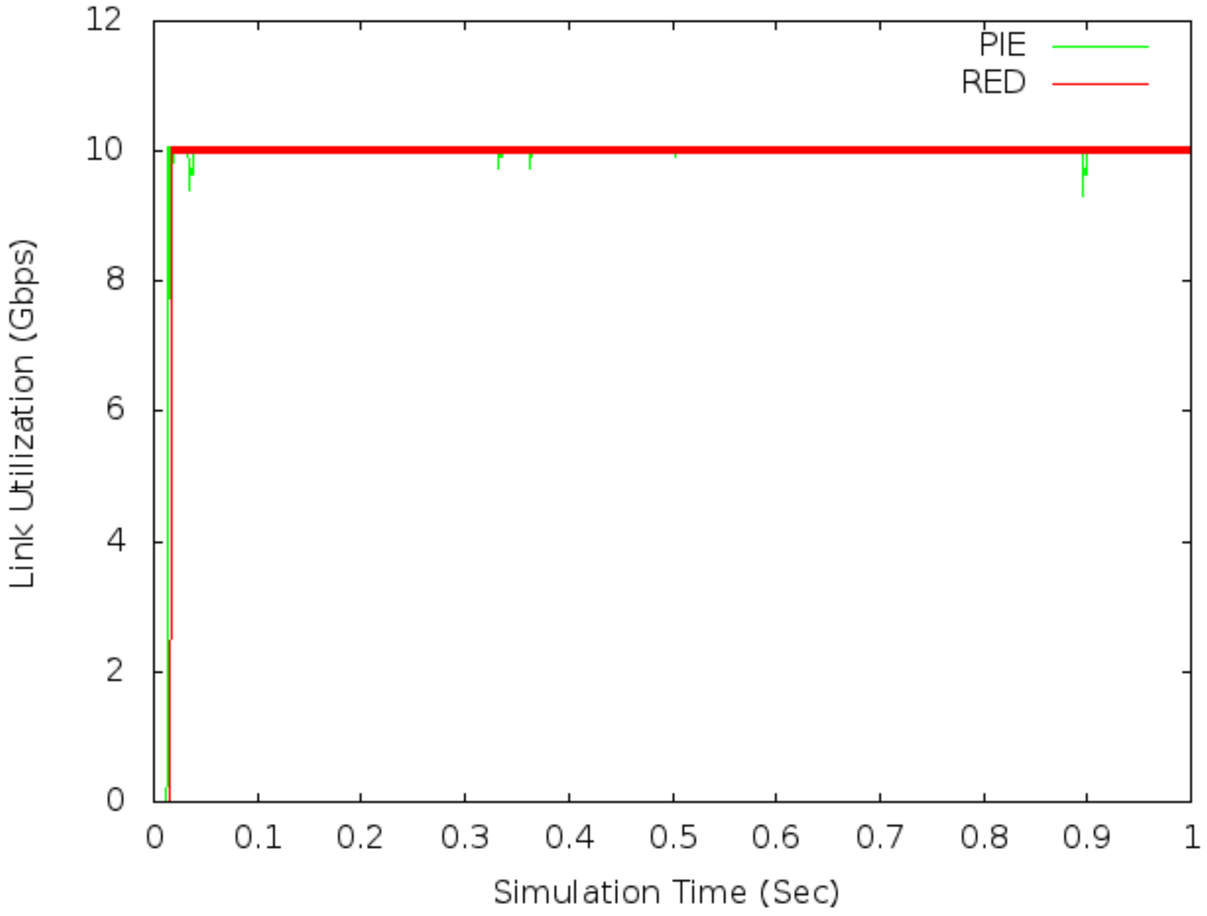
RTT = 100us
del_ref = 20us
BufferLimit = 2000 pakcets

Simulation RED+ECN vs. PIE+ECN - 5 Cubic TCP Flows (low multiplexing)



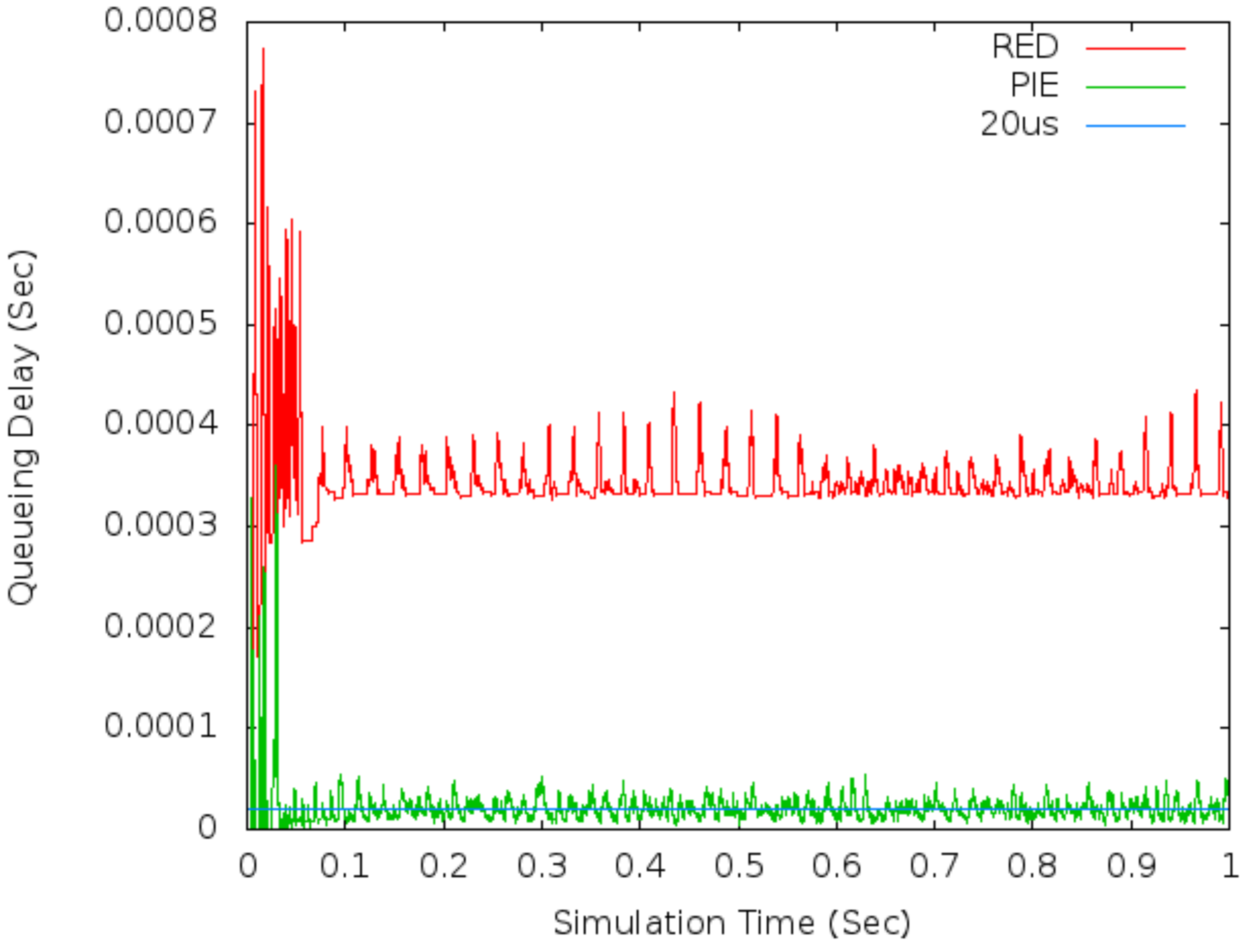
RED, whose minimum and maximum thresholds are based on proportions of the buffer size, can not control latency directly. PIE is able to effective control the queueing delay based on the delay reference value, 20us.

Simulation RED+ECN vs. PIE+ECN - 5 Cubic TCP Flows (low multiplexing)



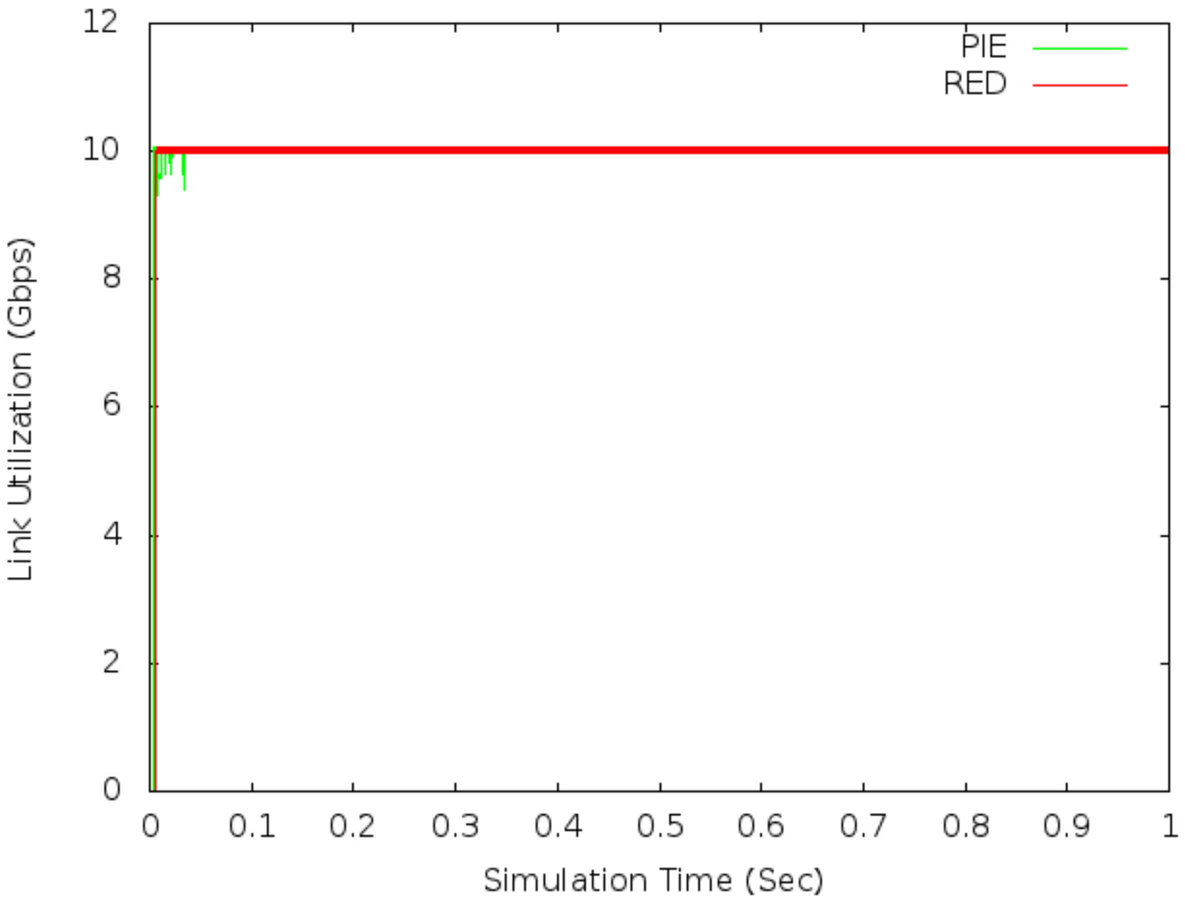
With a slight more multiplexing, 100% is achieved with PIE+ECN.

RED+ECN vs. PIE+ECN:
- 20 Cubic TCP Flows (Queueing Delay)



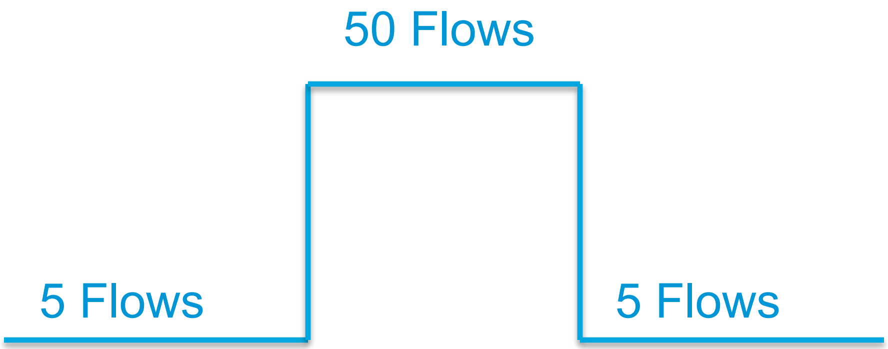
Under moderate congestion, PIE+ECN continues to outperform RED+ECN.

RED+ECN vs. PIE+ECN: - 20 Cubic TCP Flows (Throughput)

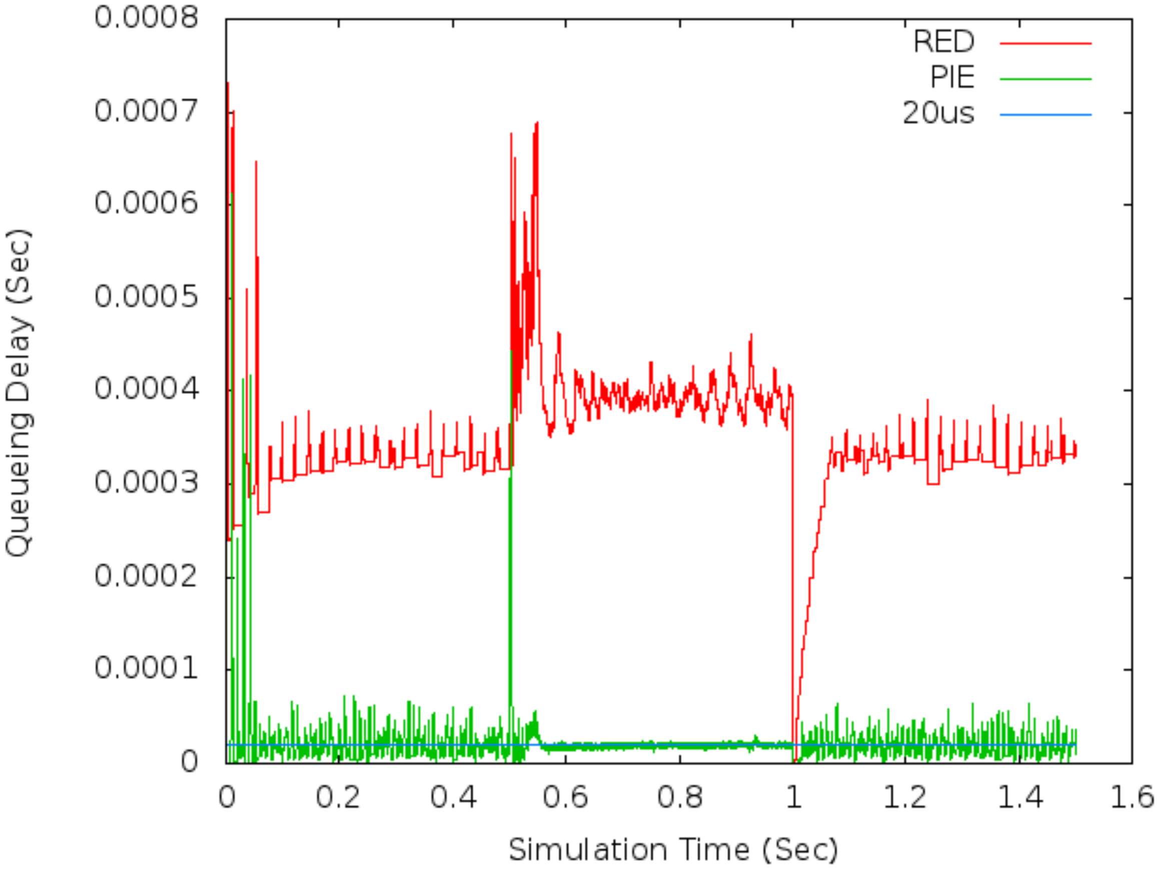


Throughput is again 100%.

Varying TCP Traffic Intensity on 10Gbps Link

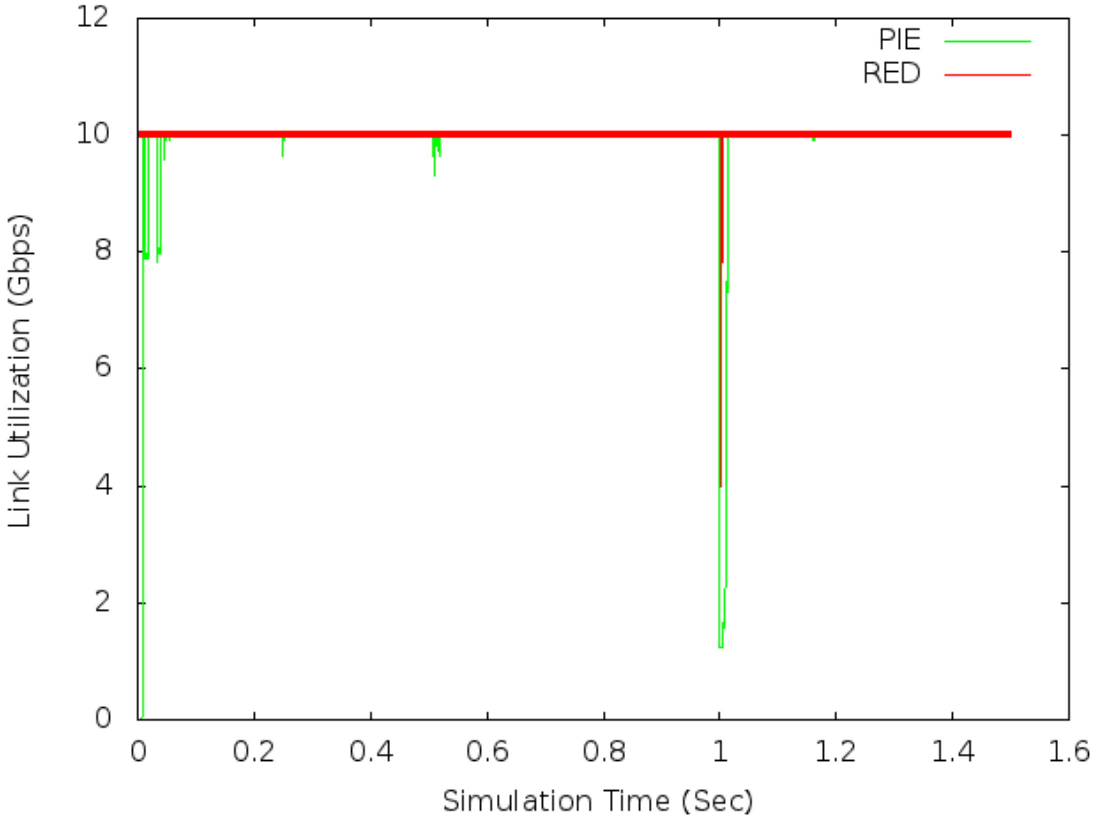


RED vs. PIE – Changing Traffic Intensity, Queue Delay



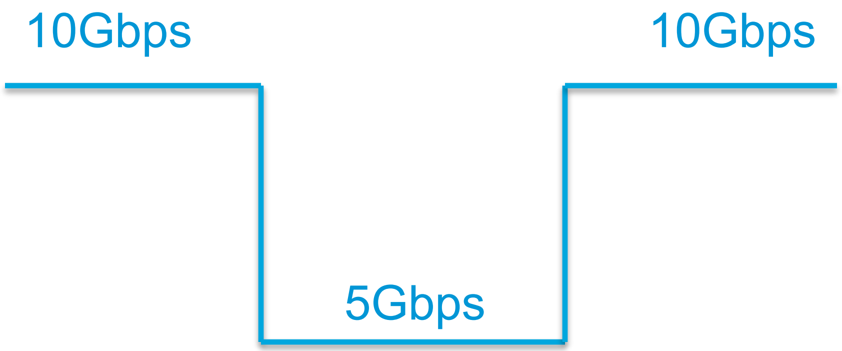
Under dynamic scenarios, PIE consistently outperforms RED

RED vs. PIE
– Changing Traffic Intensity, Throughput



Overall high throughput is achieved. Slightly more dip of throughput because of low latency maintained.

Varying Link Capacity, 20 Cubic TCP Flows

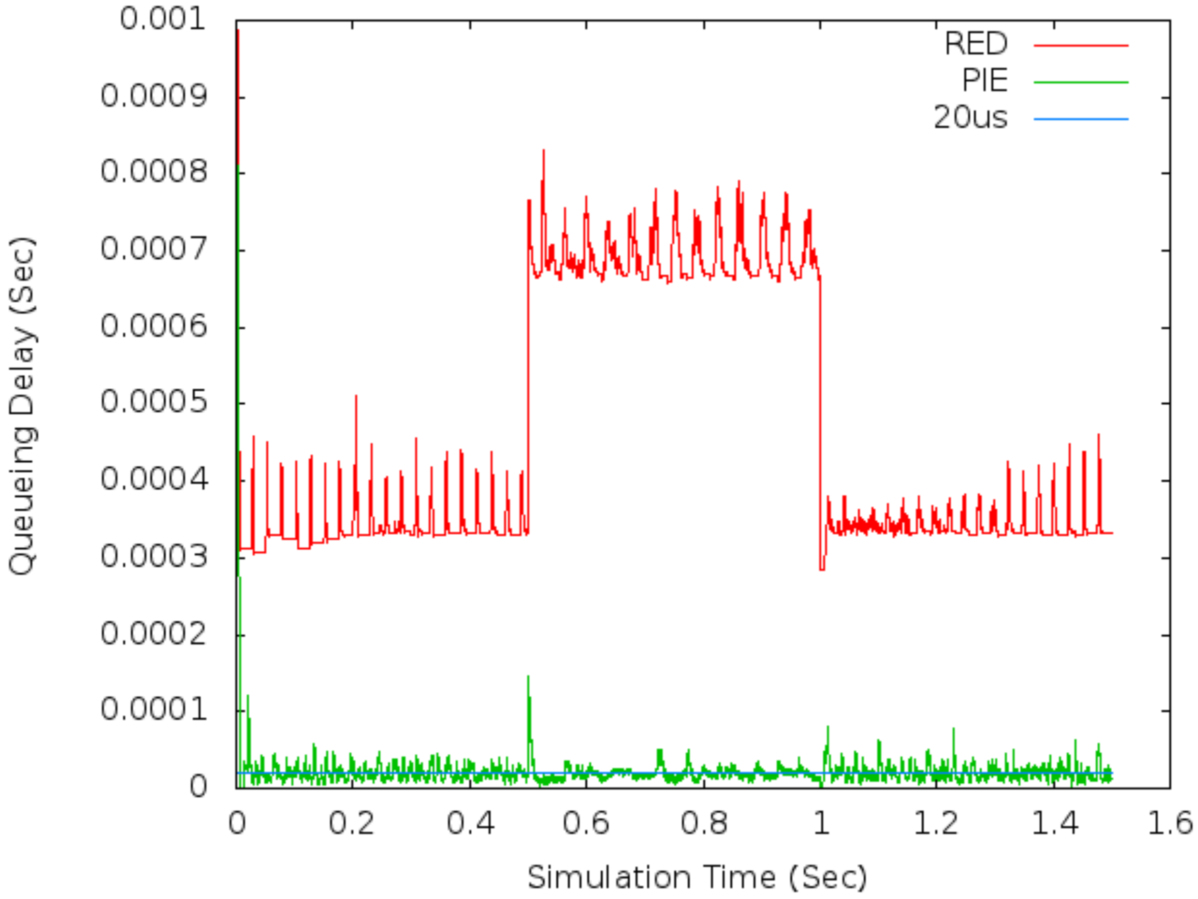


10Gbps

5Gbps

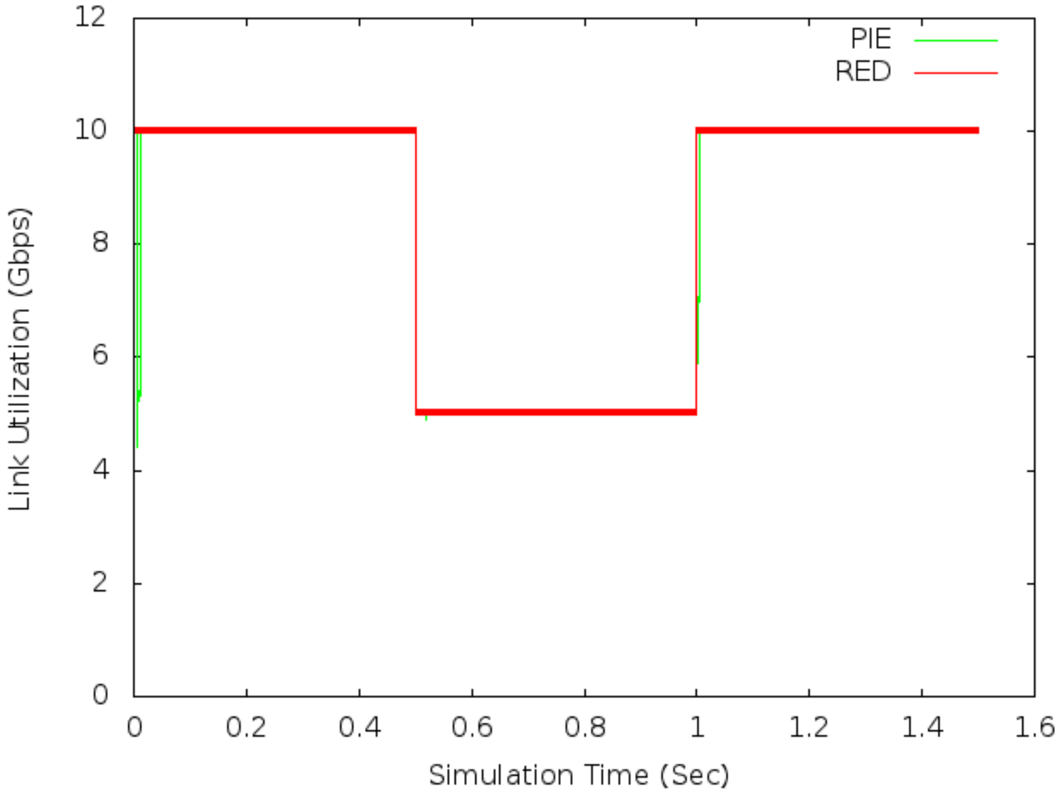
10Gbps

RED vs. PIE – Queue Delay



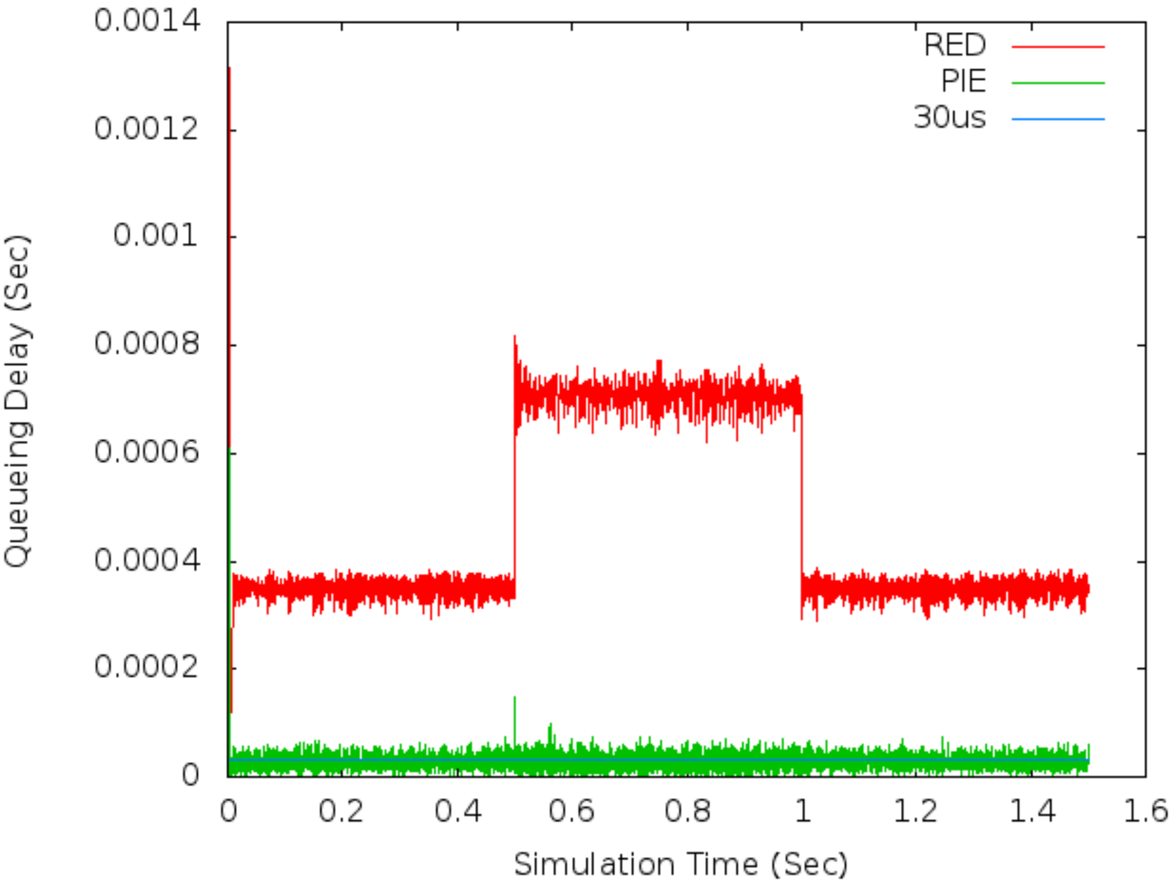
When the link capacity changes, RED fails to maintain consistent queueing latency. On the other hand, PIE is able to achieve low latency regardless of varying capacity.

RED vs. PIE – Changing Bandwidth, Throughput



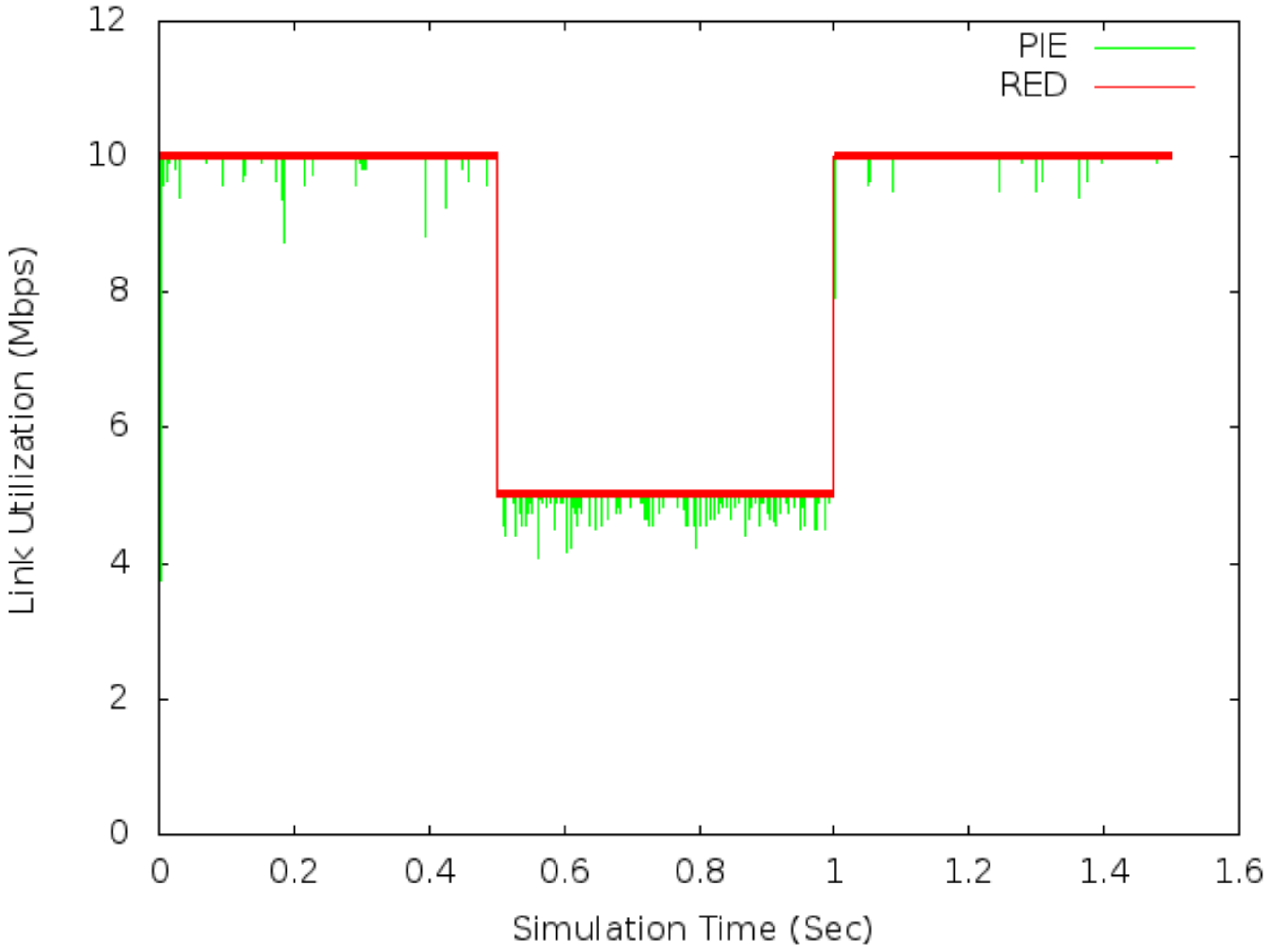
Throughput behavior is similar in this case

Using TCP Sack - Delay



Similar behavior except smoother queueing delay as TCP Cubic is more aggressive.

RED vs. PIE
– Changing Bandwidth, Throughput

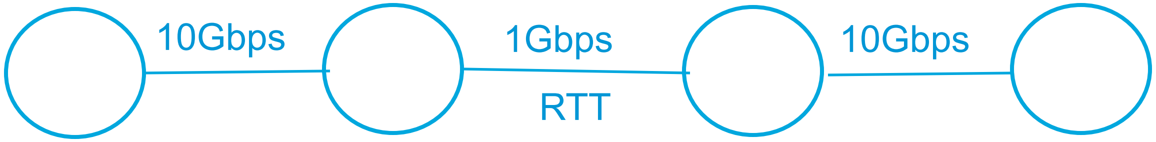


Throughput behavior is similar.

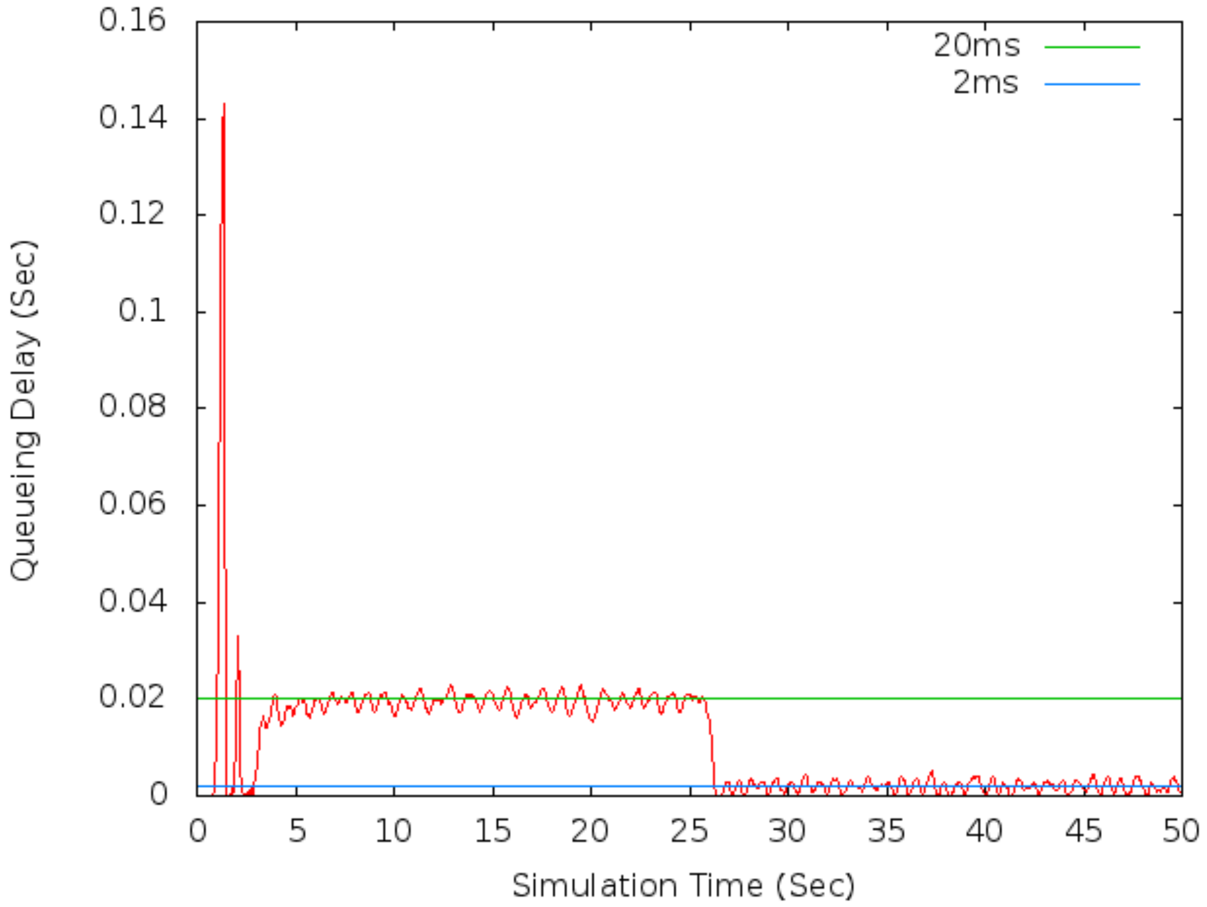
Service Provider Space
- Adjusting Del_Ref

Simulation Setup

- alpha = 0.125; beta = 1.25, Tupdate = 30ms
- Congestested Link Bw: Vary
- Avg Pkt Size: 1000B
- TCP: Sack
- 200 TCP Flows

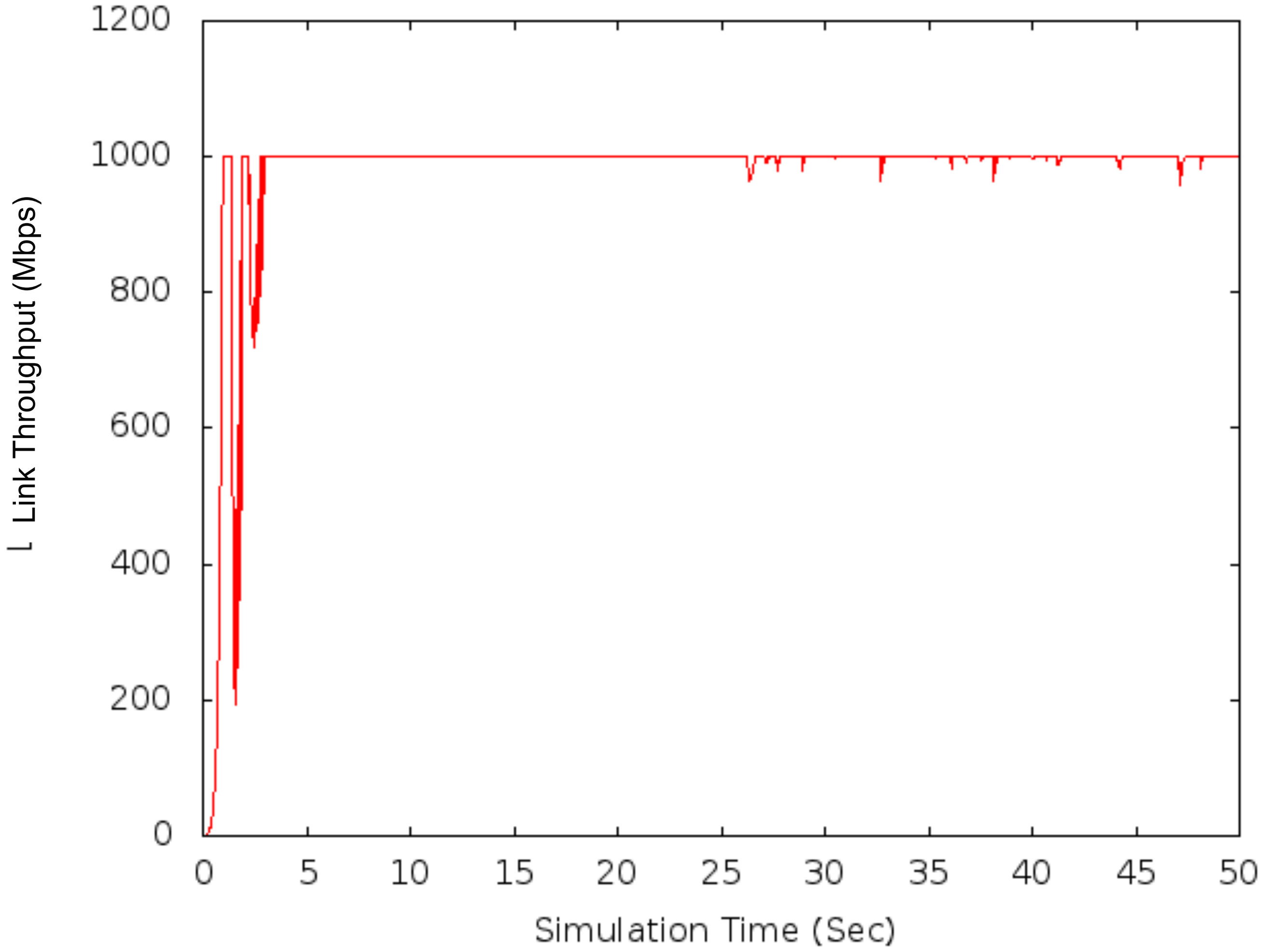


Based on link capacity or application requirement



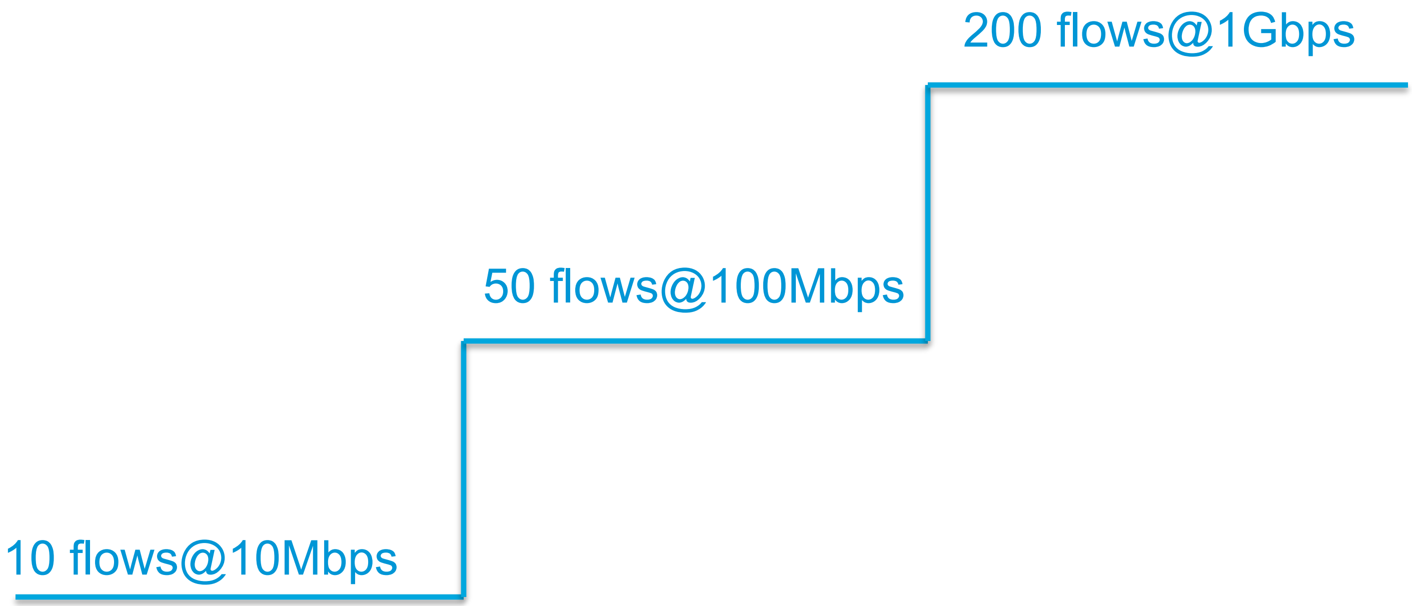
Delay reference can be tuned dynamically without affecting stability.

Link Throughput During the Adjustment



Throughput is maintained high, Based on the applications, delay reference could be set evener lower with some compromise on the throughput.

Traffic Scenarios

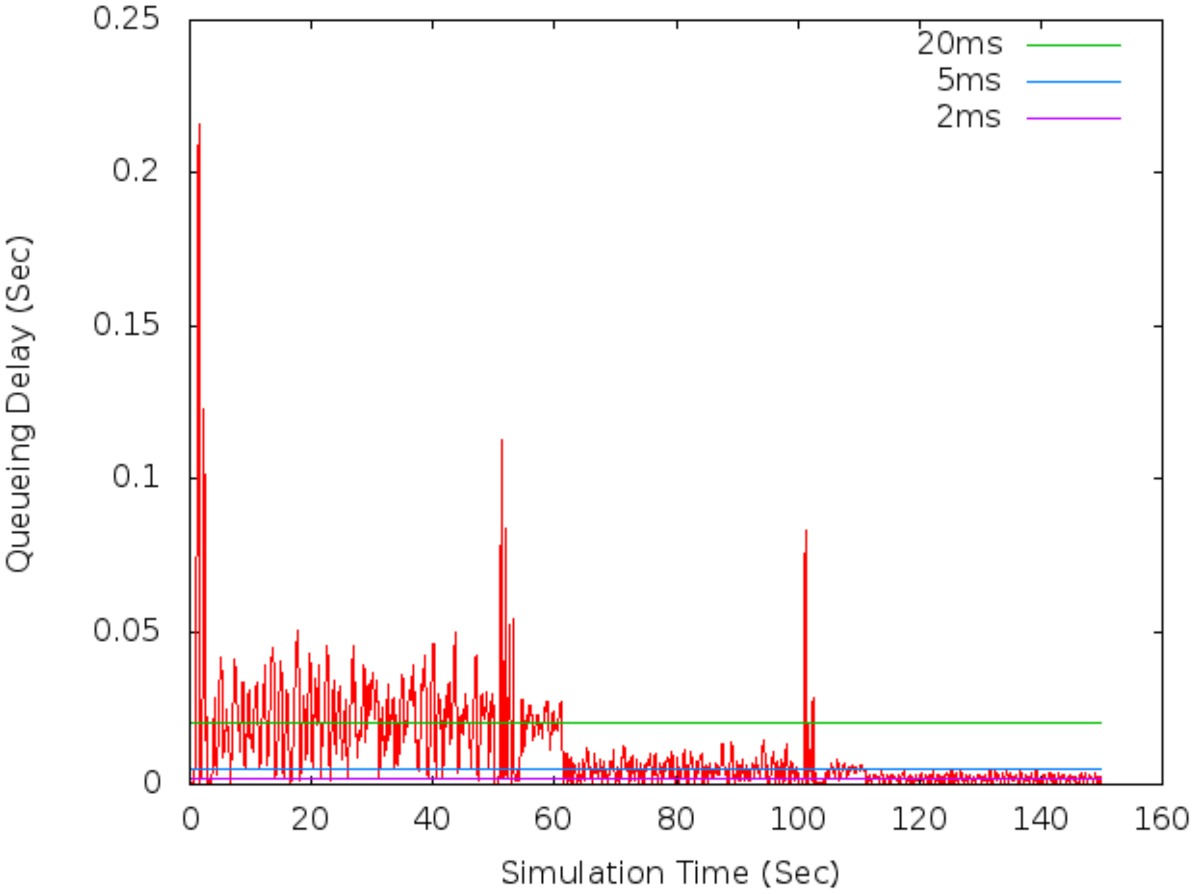


200 flows@1Gbps

50 flows@100Mbps

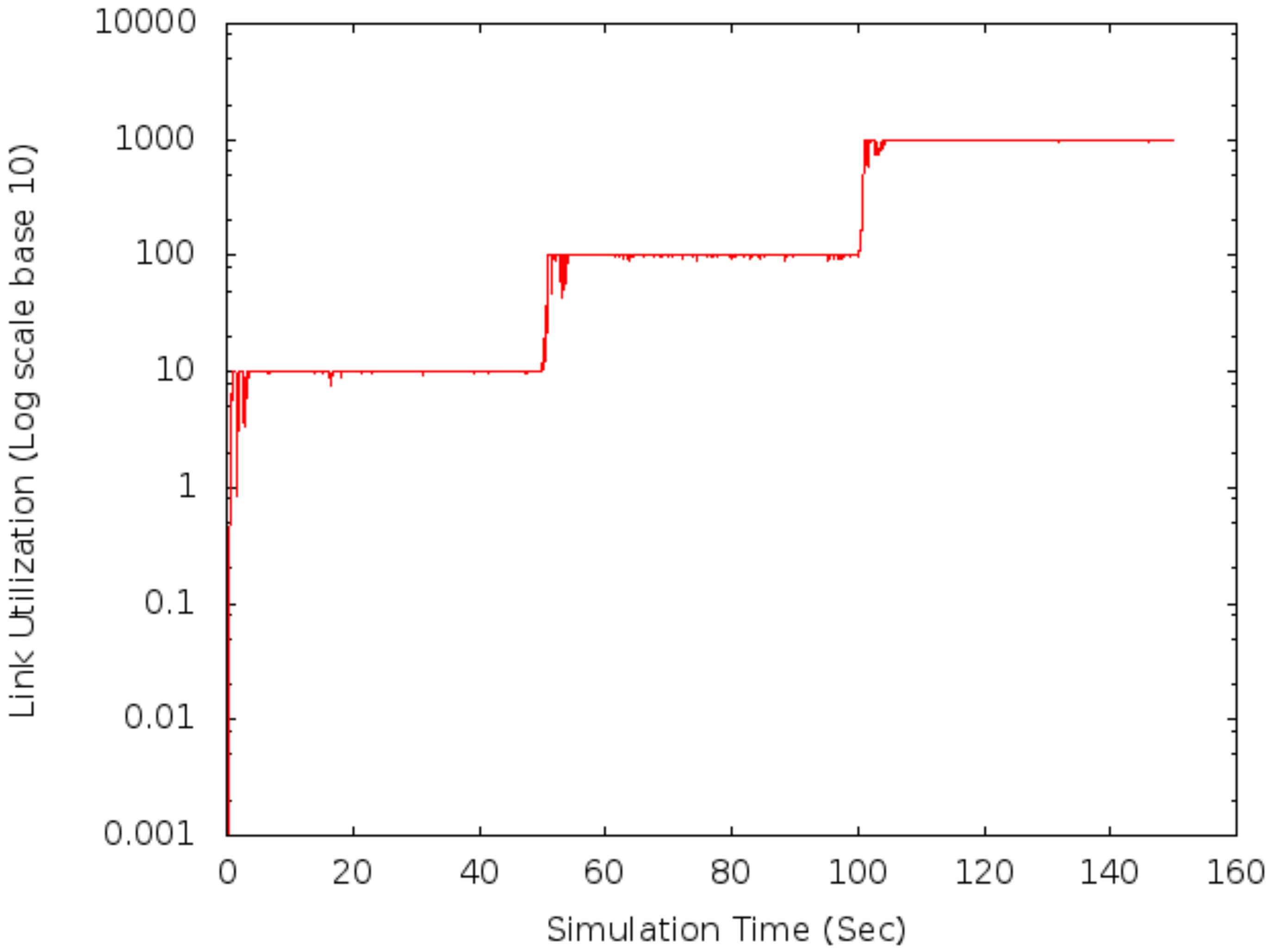
10 flows@10Mbps

Auto-setting Based on the Queue Draining Speed



If the algorithm detects that a draining speed have increased and has been stable for 10sec, the scheme automatically lowers the delay reference value.

Auto-setting Based on The Link Speed

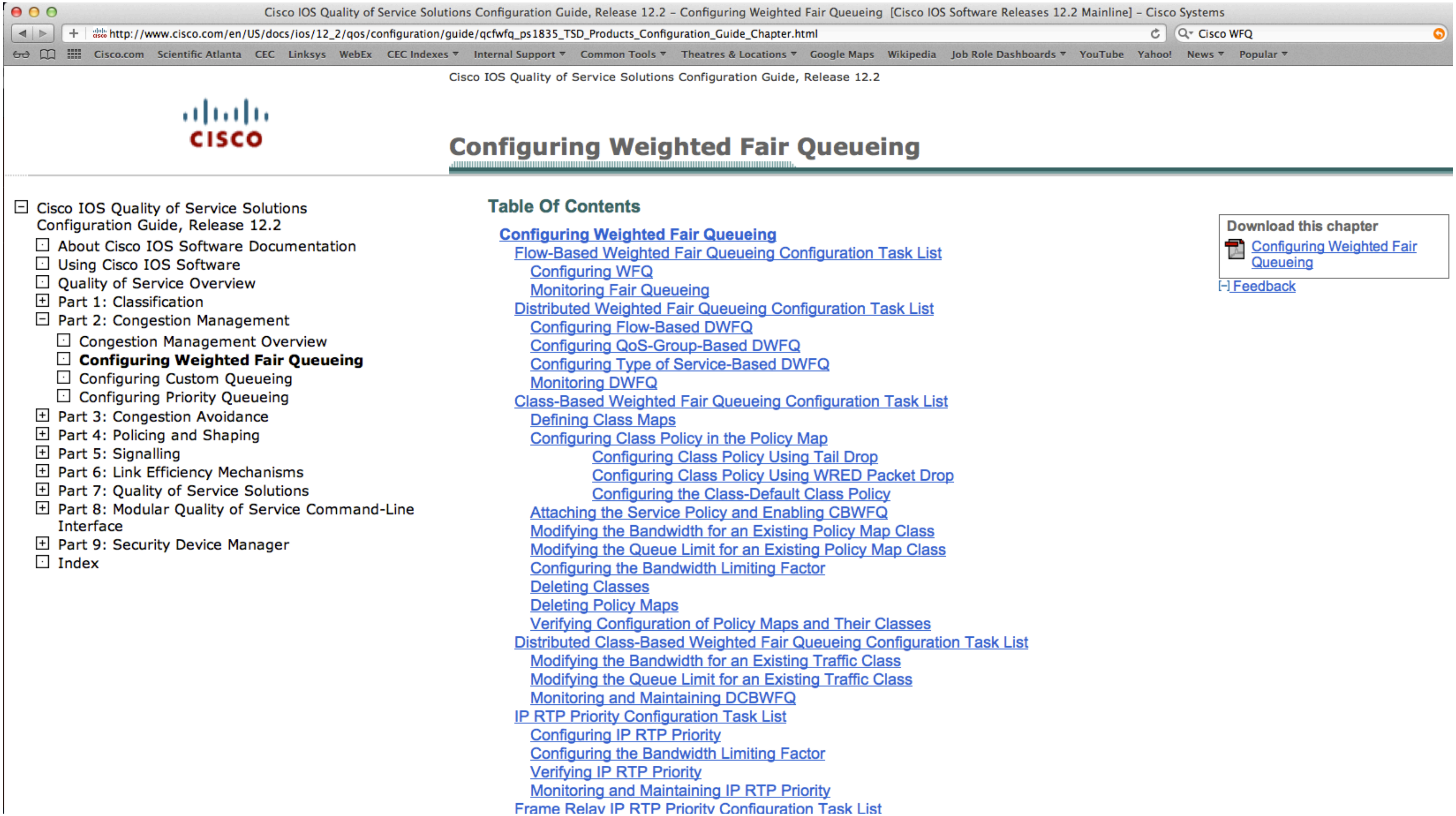


FQ_PIE

Fair Queueing and Its Derivative at Cisco

- History of Fair Queueing Research
  - Originally proposed in RFC 970
  - Significant research reported in SIGCOMM and INFOCOM 1988-1992
- History of Fair Queueing Implementations
  - First known commercial implementation: ACC 1991
  - Cisco implemented in IOS 11.0 and 11.1ac 1995
    - Weighted stochastic (hashed) per-flow fair queueing
    - Target: achieve common bit rate for the top sessions

FQ Offerings at Cisco



Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2 – Configuring Weighted Fair Queueing [Cisco IOS Software Releases 12.2 Mainline] – Cisco Systems
http://www.cisco.com/en/US/docs/ios/12_2/qos/configuration/guide/qcfwfq_ps1835_TSD_Products_Configuration_Guide_Chapter.html
Cisco WFQ
Cisco.com
Scientific Atlanta
CEC
Linksys
WebEx
CEC Indexes
Internal Support
Common Tools
Theatres & Locations
Google Maps
Wikipedia
Job Role Dashboards
YouTube
Yahoo!
News
Popular

Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2

Configuring Weighted Fair Queueing

Table Of Contents

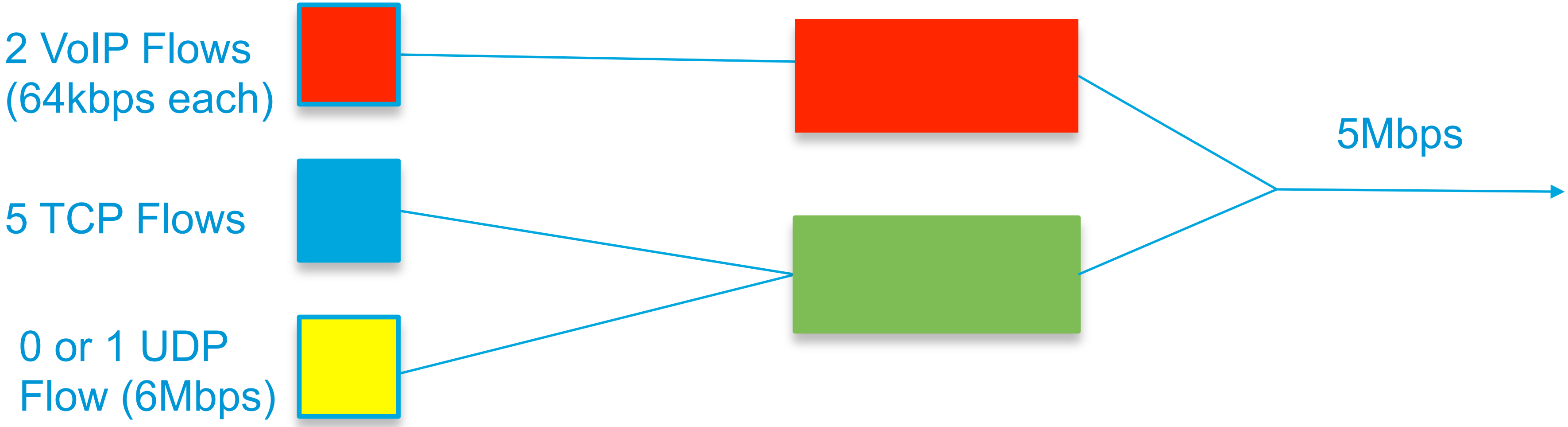
Download this chapter

Configuring Weighted Fair Queueing

[-] Feedback

PIE is designed to be on top of an FQ structure

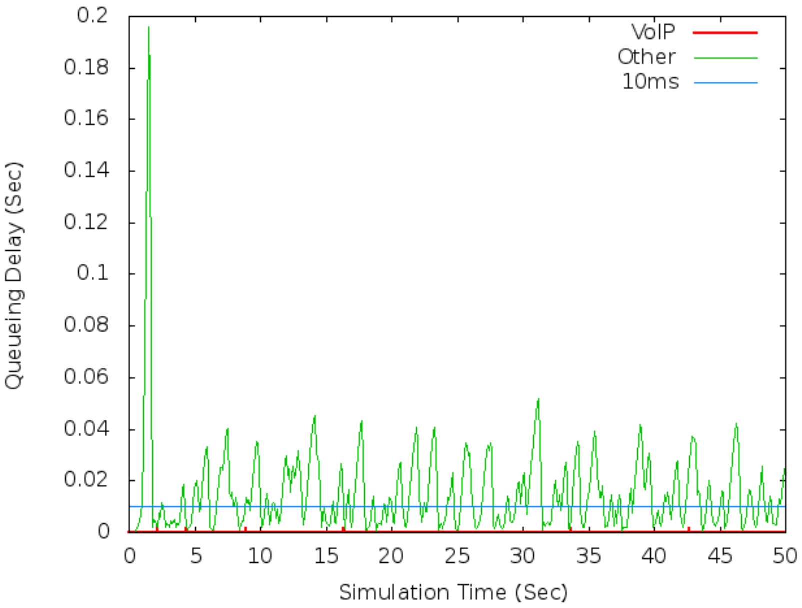
FQ_PIE for Mixed Traffic - VoIP in strict priority

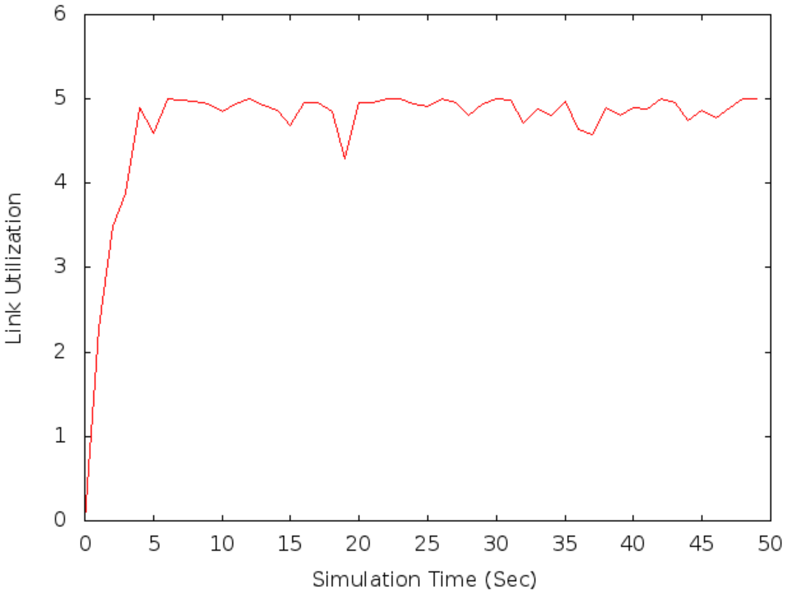


PIE parameters are the same as before, Bufferlimit = 200 packets

Cisco's Products (including Cable Business Unit) Put VoIP traffic in the Strict Priority Queue

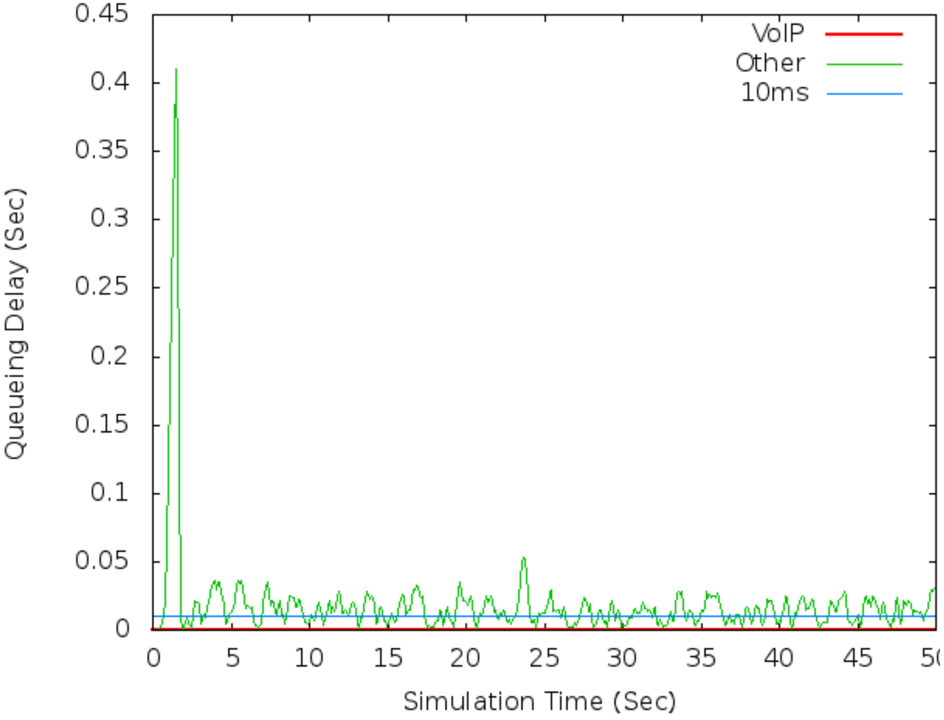
VoIP Traffic Incurs Almost 0 Queueing Latency

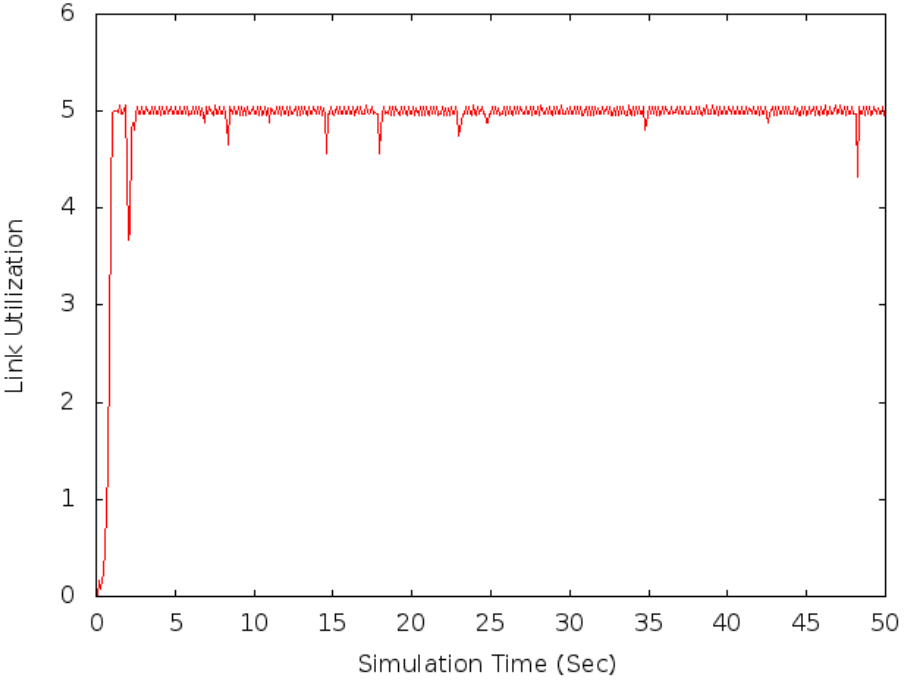




The strict priority queue is overprovisioned. The latency for the strict priority queue is zero. Hence, VoIP traffic incurs almost zero queueing latency. PIE maintains the low latency for the regular queue when 5 TCP flows are sending traffic.

FQ_PIE Controls Latency to be around 10ms given Mixture of TCP and One UDP Traffic





When a reliable UDP traffic (6Mbps) is sent over the 5Mbps link, the link is congested. However, VoIP traffic still incurs almost zero queueing latency. In addition, PIE still maintains the low latency for the regular queue when 5 TCP flows is mixed with one UDP traffic.

Next Steps

Plan for Next Step

- Apply PIE to other focus areas: such as Cable Modem/CMTS, and edge routers
  - Token bucket based Queues, how does it interact with PIE’s burst tolerance?

- Techniques to further improve the algorithm
  - Pure random drops can be back-to-back
  - Burst tolerance might cause sudden increase of drop probability

- More extensive traffic mix studies for key areas
  - Reflective of Cable/CTMS scenarios
  - Service provider edge routers
  - Data centers: including map-reduce traffic

Thank you.

cisco

Backup