IEEE 802.1Q Congestion Notification

Overview

Patricia Thaler

Broadcom

IETF 87 Berlin, Germany, July 2013

Congestion Notification (CN) Purpose

- Provide a means for a bridge to notify a source of congestion causing the source to reduce the flow rate.
- Motivated in part by I/O consolidation onto data center networks
  - Moving protocols from specialized networks with link flow control to Ethernet, e.g. FCoE and RoCE
  - Provide a way to mitigate congestion spreading from link flow control such as Priority-based Flow Control

Congestion Notification (CN)

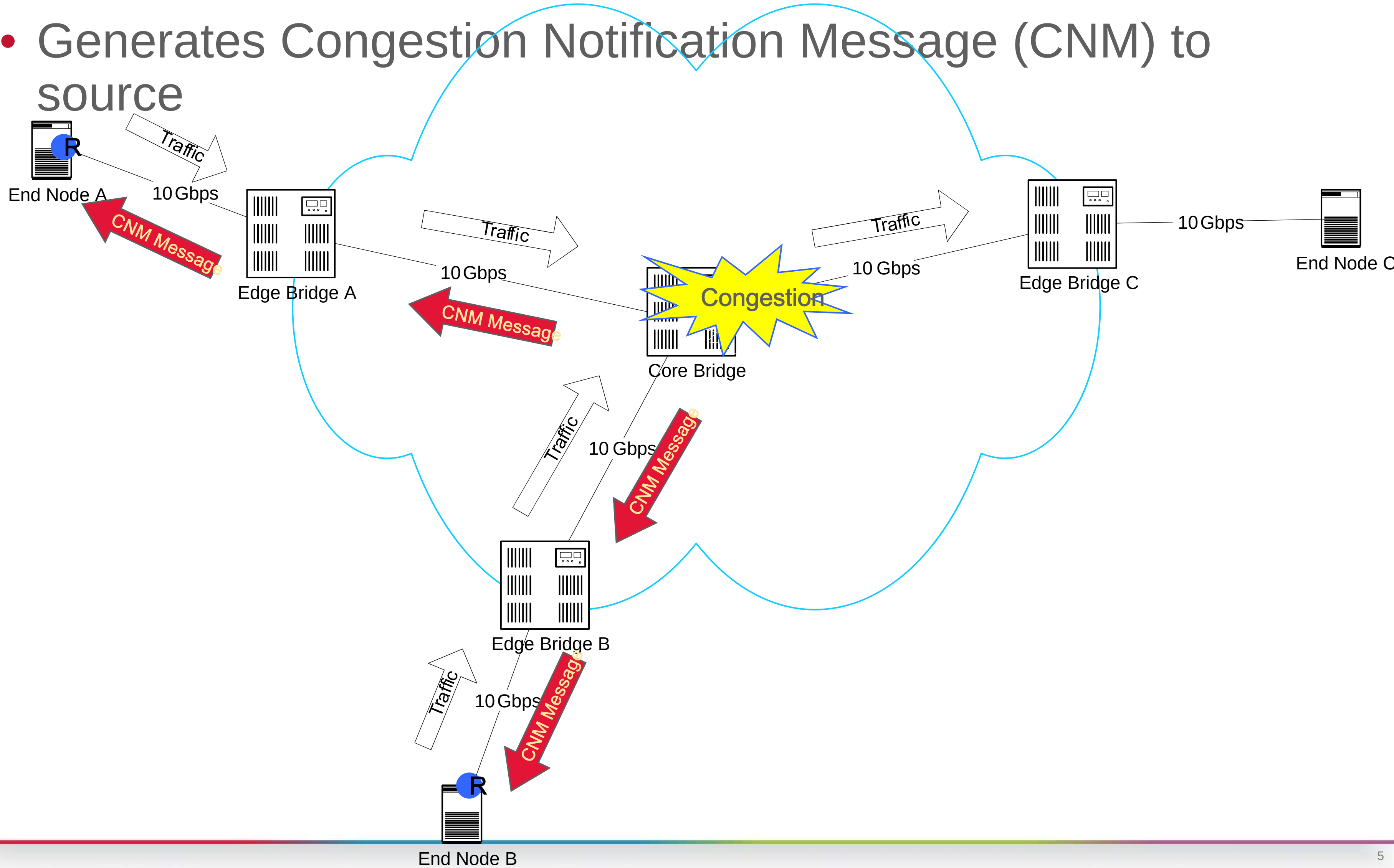
- Provide a means for a bridge to notify a source of congestion causing the source to reduce the flow rate.
- CN is targeted at
  – Networks with low bandwidth delay products: e.g. data center
  – Long lived data flows
- Operates on frames in a VLAN priority
  – Allows for sharing the network between congestion controlled and non-controlled traffic.
- Goals: avoid frame loss; reduce latency; improve performance
- Originally IEEE Std 802.1Qau; now incorporated into IEEE 802.1Q-2011

Selected Objectives

- Independent of upper layer protocol
- Coexist with TCP
  - (i.e. nested control loops of CN and TCP flow control produce reasonable behavior)
- Unicast traffic
- Support bandwidth delay product of 5 Mbit
- Operates over a domain where all bridges and end stations support CN
- No per flow state or per flow queuing in bridges

CN uses BCN messages

- Bridge detects congestion when queue is above equilibrium level
- Generates Congestion Notification Message (CNM) to source



Identifying flows

- Source may tag frames with a CN-Tag
  - Contains a 2 byte Flow ID
  - Flow ID meaning is local to source
  - Flow ID is returned in the CNM
- Allows source to identify the flow to apply the rate limit

Congestion Notification Message (CNM) Content

- Version, 4 bit
- Quantized Feedback, 6 bits, a function of cnmQoffset and cnmQDelta
- Congestion Point Identifier (CPID)* - 8 byte
- cnmQoffset*, units of 64 bytes - 2 byte
- cnmQDelta*, units of 64 bytes - 2 byte
- Encapsulated priority (i.e. priority of the sampled frame), 3 bit
- Encapsulated destination MAC address (DA of sampled frame), 6 byte
- Encapsulated MSDU length, 2 byte – max value 64
- Encapsulated MSDU, up to 64 bytes

* Not used by reaction point algorithm – CPID can be used to identify the congestion location.

CNM transmission

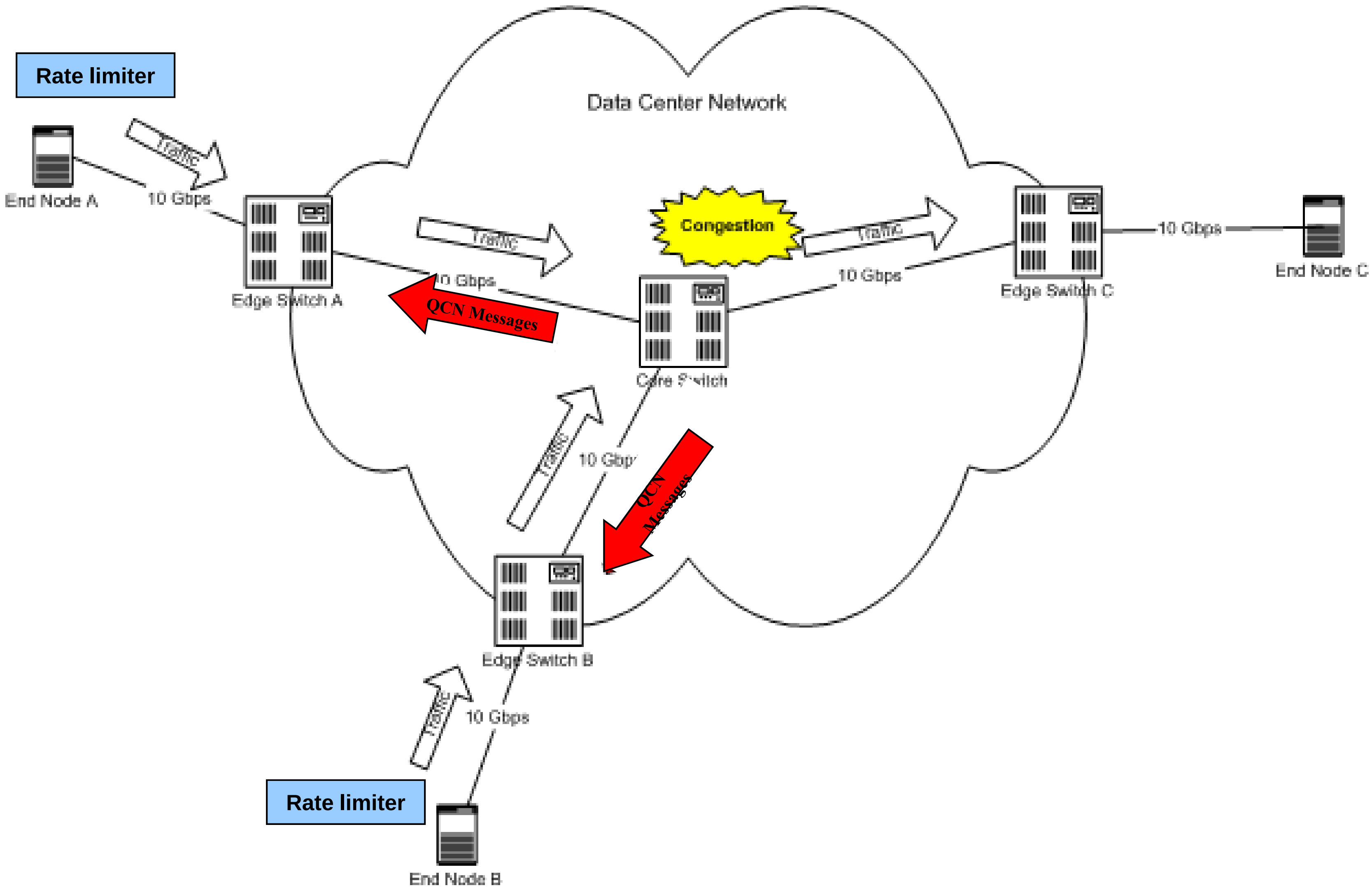
- CNMs normally transmitted in a higher priority to reduce reaction time
- Encapsulated MSDU can be used to forward CNMs produced by a bridge that receives tunneled frames
  - E.g. in an IEEE 802.1 Provider Backbone Bridged Network frames are encapsulated with an outer address that identifies the PBB edge bridges, not frame source and destination. (IEEE 802.1Q 32.16)
  - A bridge in the PBBN will send the CNM to the edge bridge DA
  - The PBB edge bridge
    - Removes the inner source address, destination address, VLAN Tag and CN-Tag, if present, from the encapsulated MSDU field,
    - Places DA in the Flow ID in encapsulated destination MAC address field
    - Adds the CN-Tag and VLAN Tag to the frame
    - Sends the frame to the inner source address

QCN: Algorithm Overview and Its Basic Benchmark Simulation

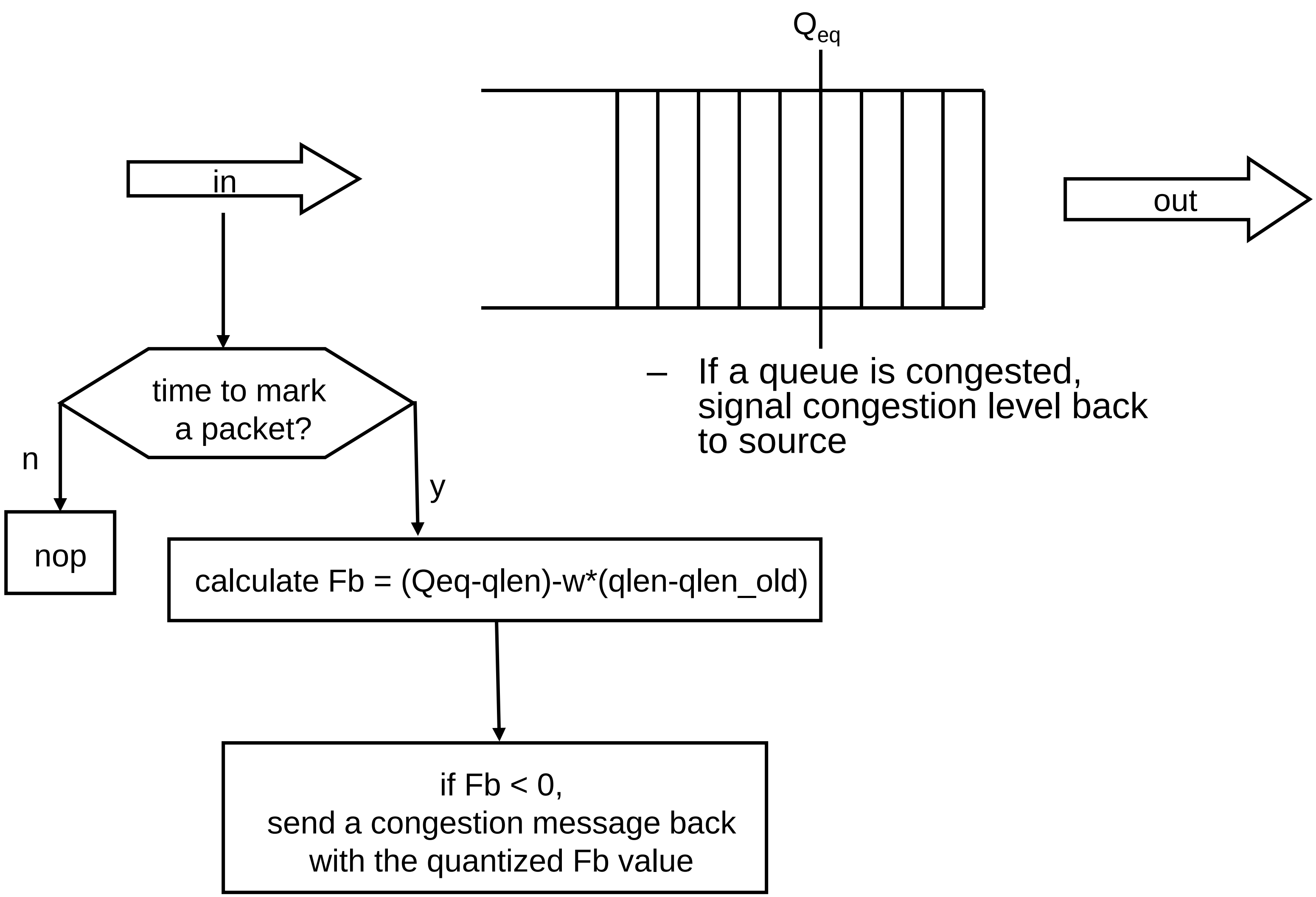
Rong Pan

IETF 87
Berlin, Germany
July, 2013

Ethernet Congestion Control for Data Center Network

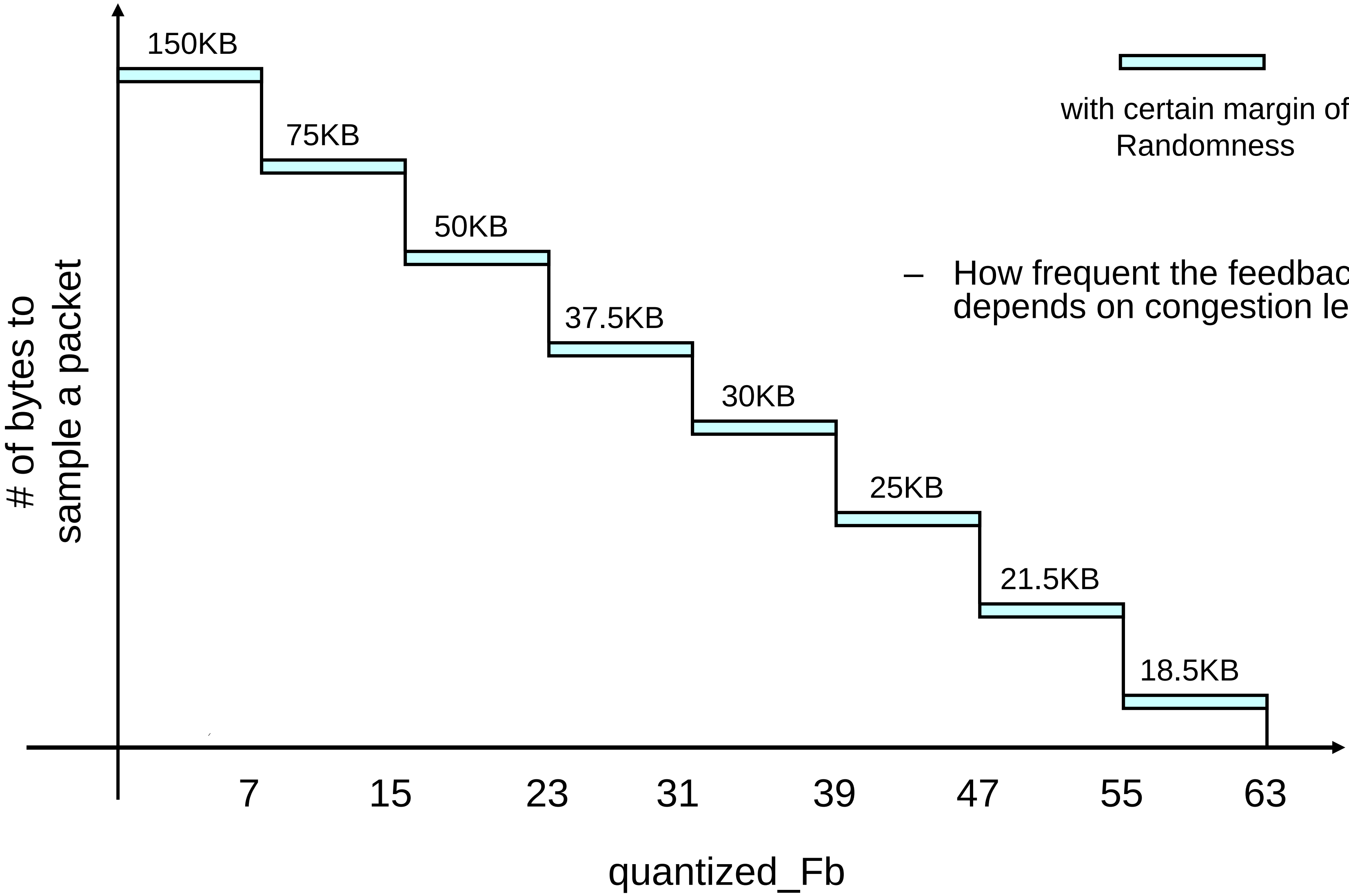


QCN (A Brief Review)
- Congestion Point



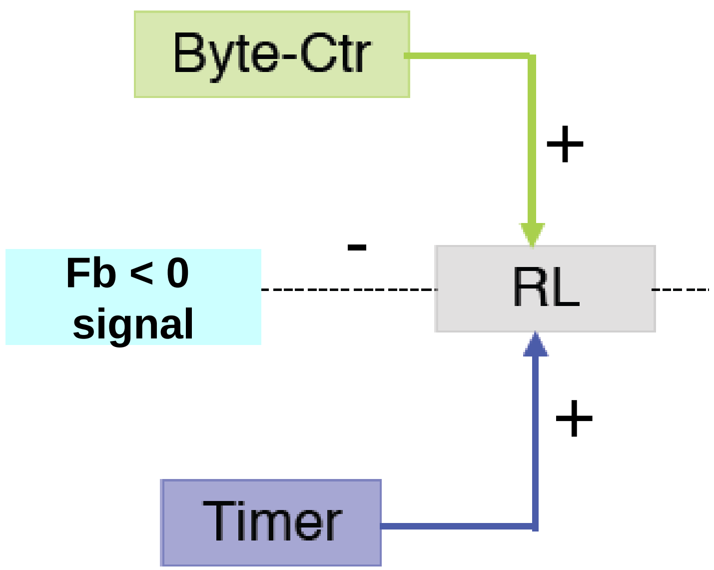
– If a queue is congested, signal congestion level back to source

How To Sample & Mark A Packet?



– How frequent the feedback is depends on congestion level

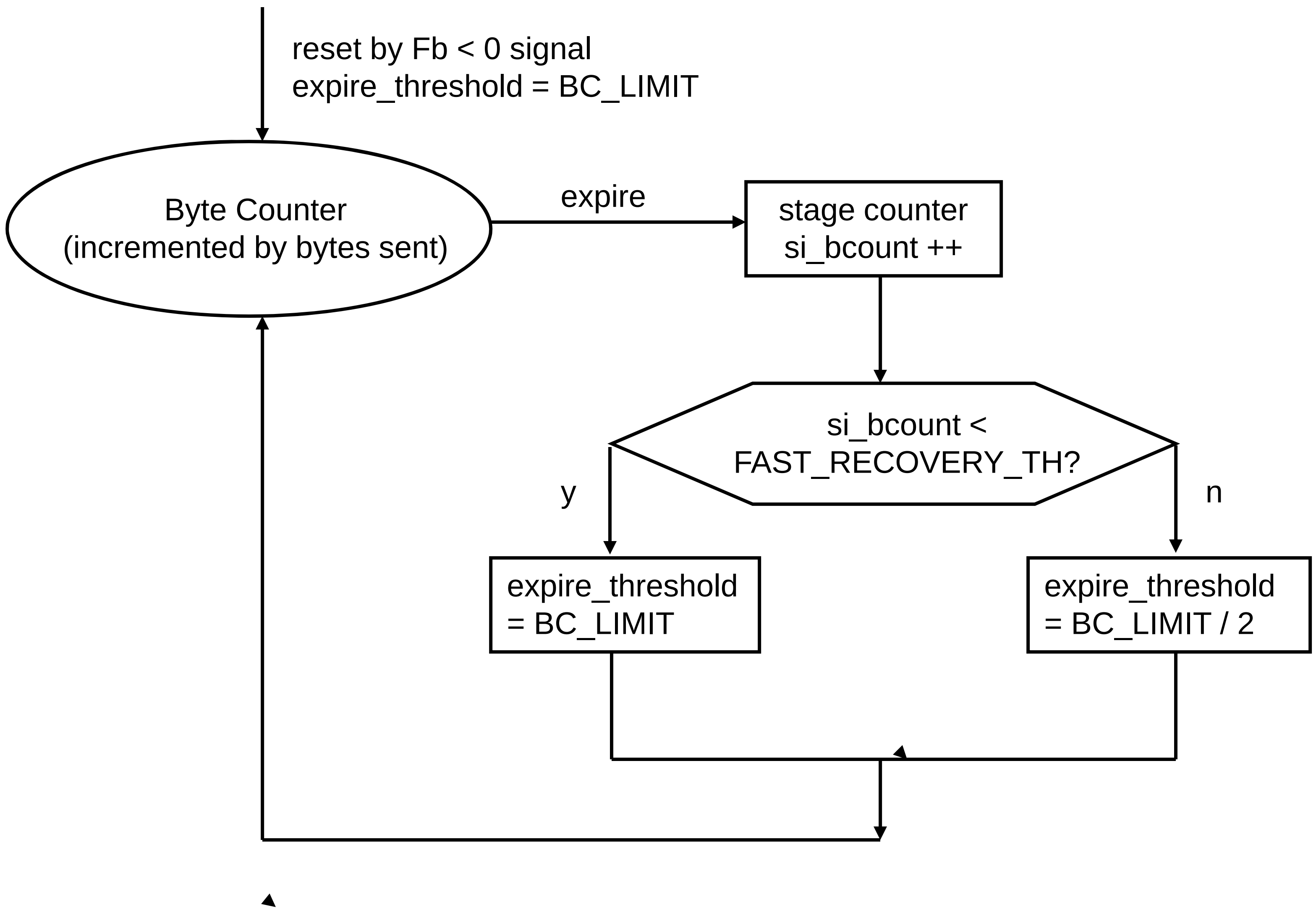
QCN (A Brief Review)
- Reaction Point



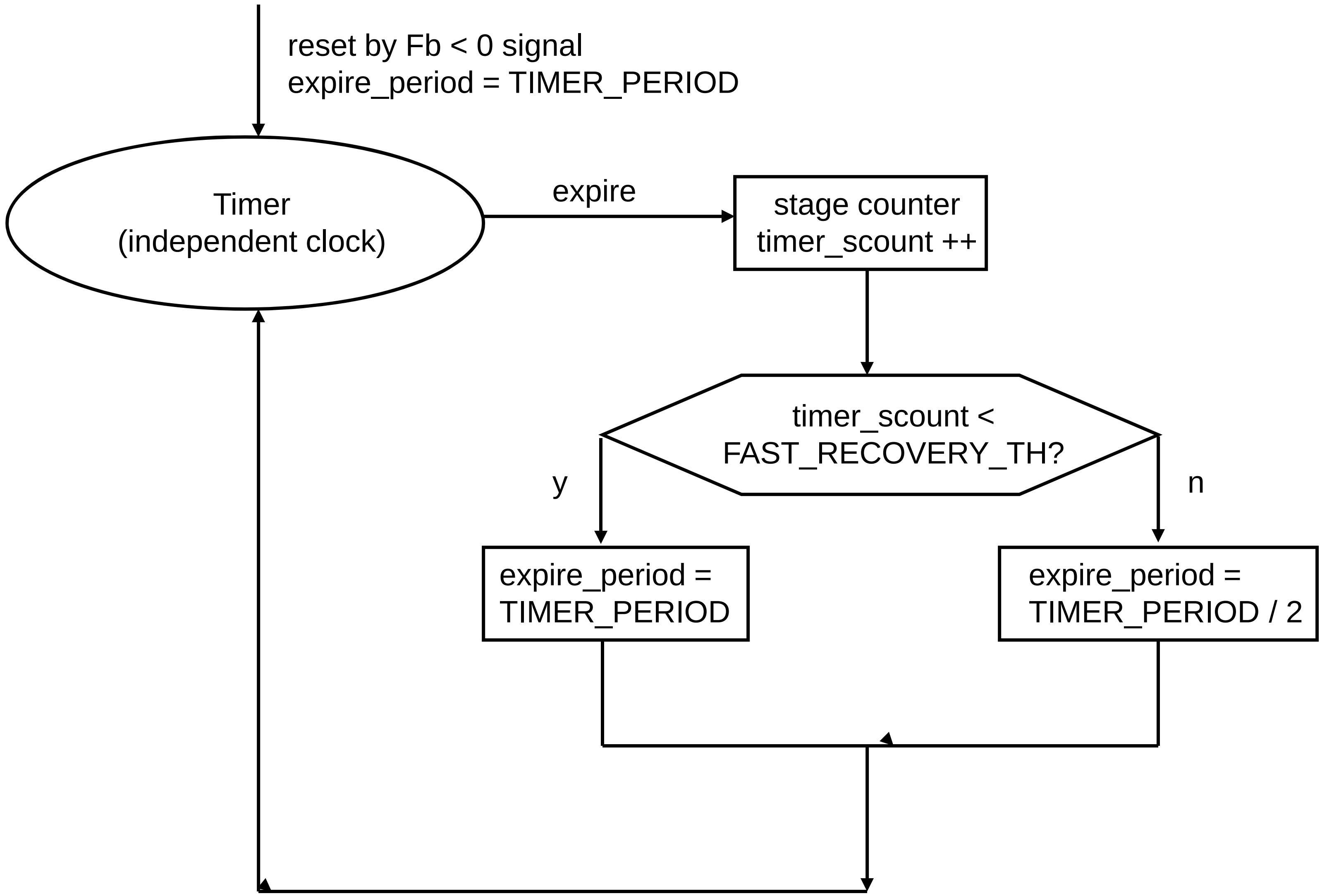
Two counters: byte-counter and timer cycle through independently
Both reset by Fb < 0 signal

- Byte-Counter
  - Fb < 0 resets byte-counter
  - Stage counter is incremented whenever a certain amount of data is accumulated
- Timer
  - Fb < 0 resets timer
  - Stage counter is incremented whenever a certain amount of time has passed
- Rate Limter (RL)
  - Depending the states of Byte-Counter's and Timer's stage counters to decide rate adjustment actions

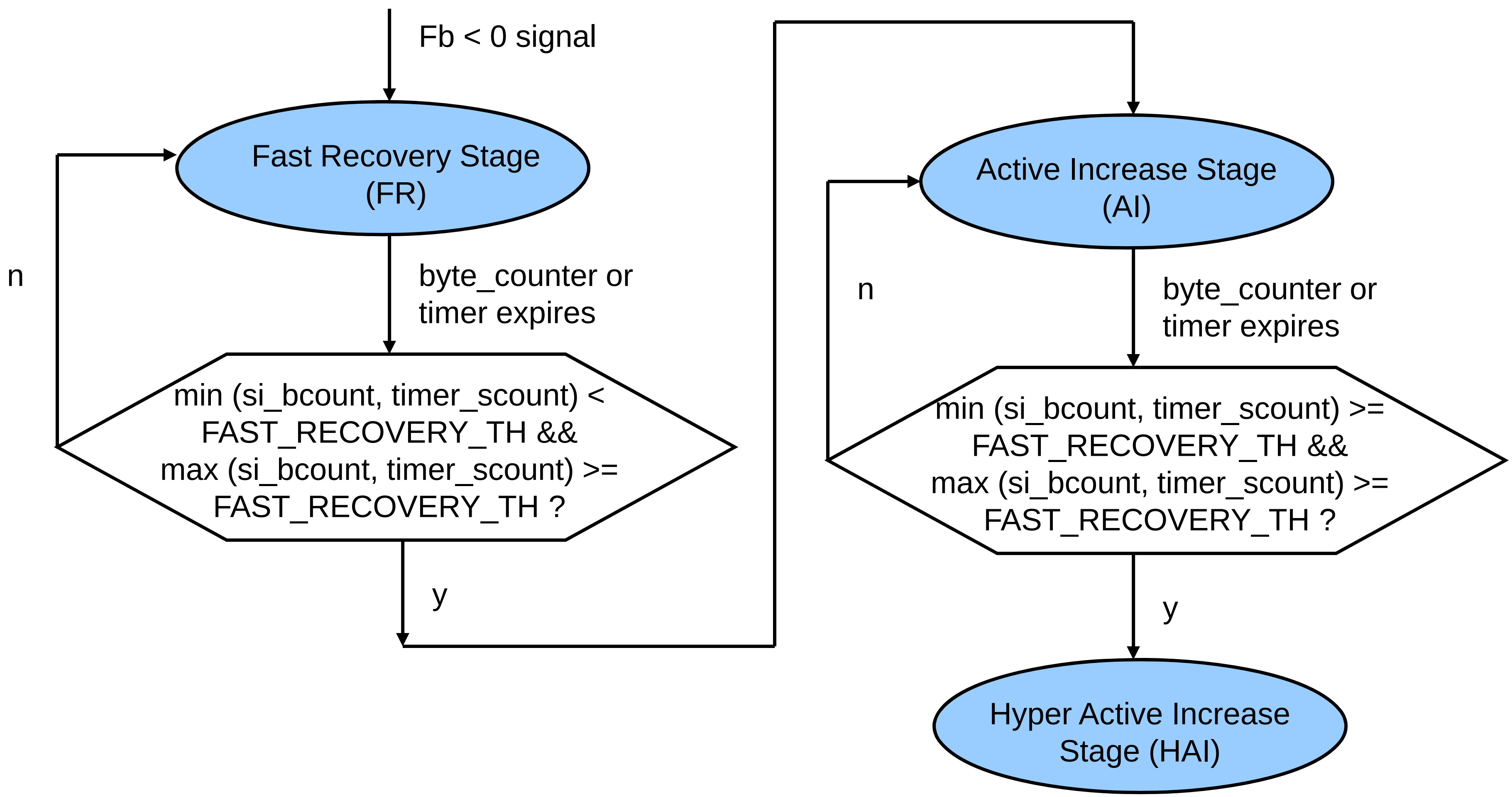
Byte Counter



Timer


reset by Fb < 0 signal
expire_period = TIMER_PERIOD
Timer
(independent clock)
expire
stage counter
timer_scount ++
timer_scount <
FAST_RECOVERY_TH?
y
n
expire_period =
TIMER_PERIOD
expire_period =
TIMER_PERIOD / 2

Rate Limiter State Diagram



Rate Changes
- Upon a Fb < 0 message

- Target Rate (TR) is decreased implicitly
  - TR = Current Rate (CR)

- Current Rate (CR) is multiplicatively decreased
  - CR = CR * (1 – Gd*|Fb|)

Rate Changes
- Upon byte_counter or timer expires

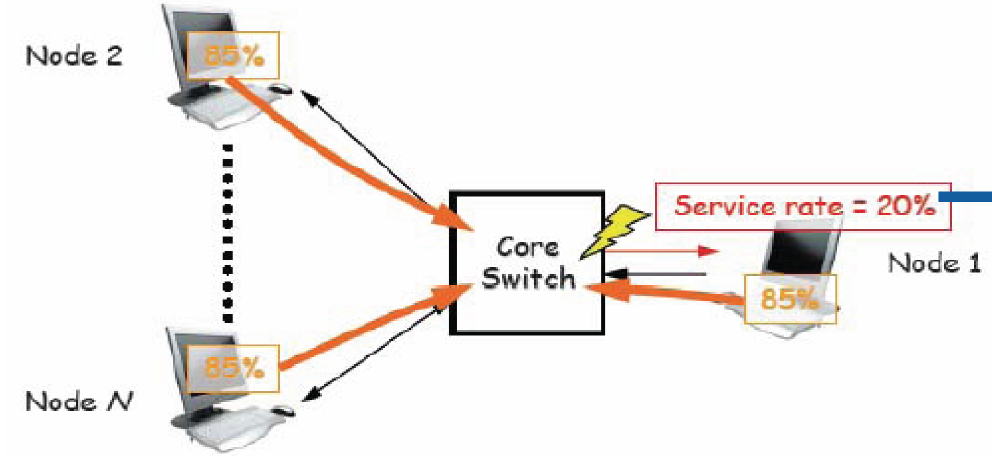
1) Fast Recovery
- TR = CR
- CR = (CR+TR)/2

2) Additive Increase
-TR = TR+R_AI; CR = (CR+TR)/2

3) Hyper Additive Increase (event numbered i = 1, 2, …)
- At the end of event number i:
TR = TR + (i*R_HAI); CR = (CR+TR)/2

Baseline #1

1. Output Generated Hot Spot Single Hop



Changed to 10%

Workload:

- All Nodes (10) : Uniform Distribution, load = 8.5Gbps
- Node 1 Service Rate = 1Gbps
- One Congestion Point
  - Hotspot:
    - Degree: 9, Severity = 8.5:1,
    - Duration: 80 mS from ti=10 to 90 mS

- Scenarios: 2Gbps, 1Gbps, 0.5Gbps OG service rates

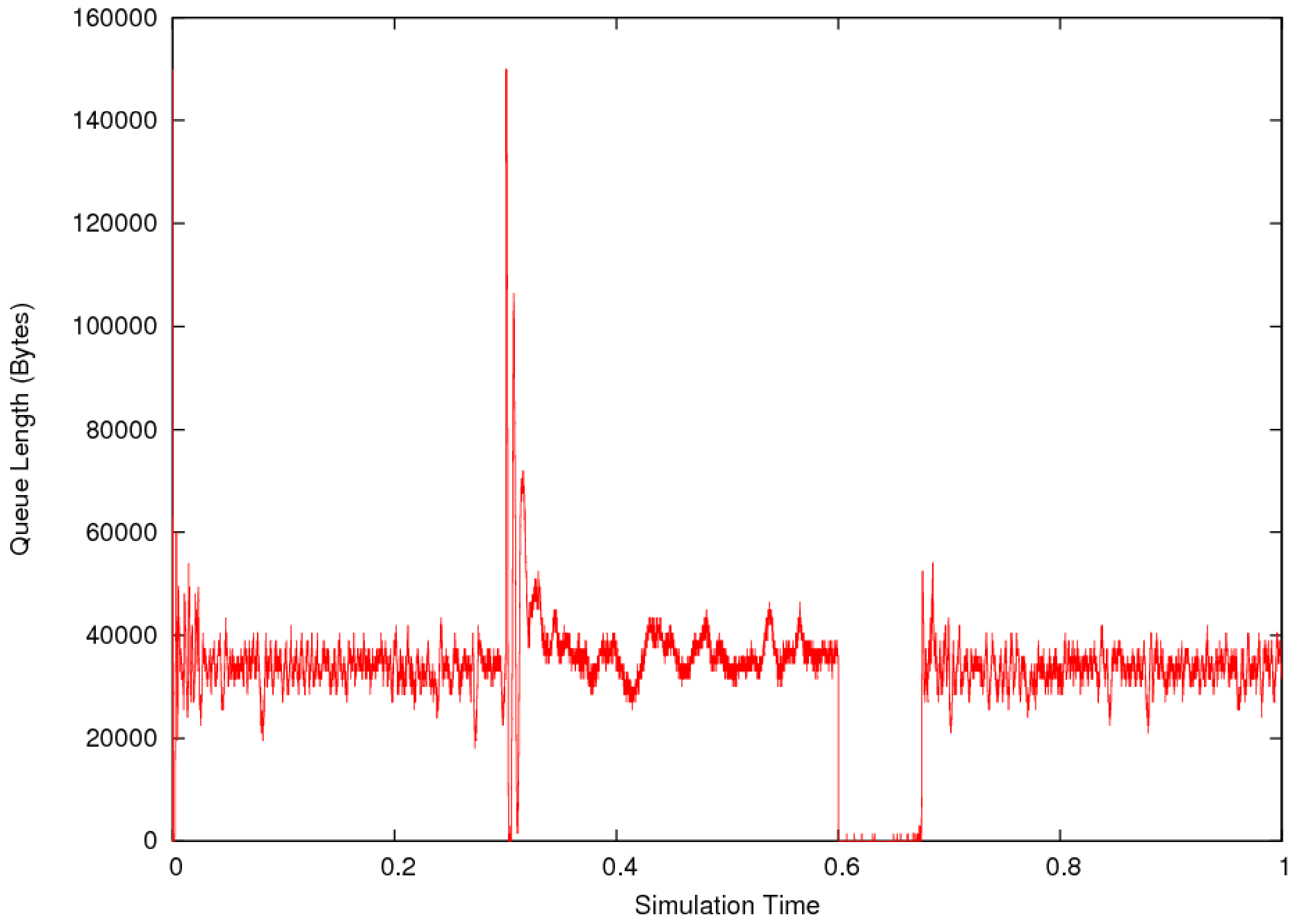
Required

Verdana regular 7pt.
Legal text goes here

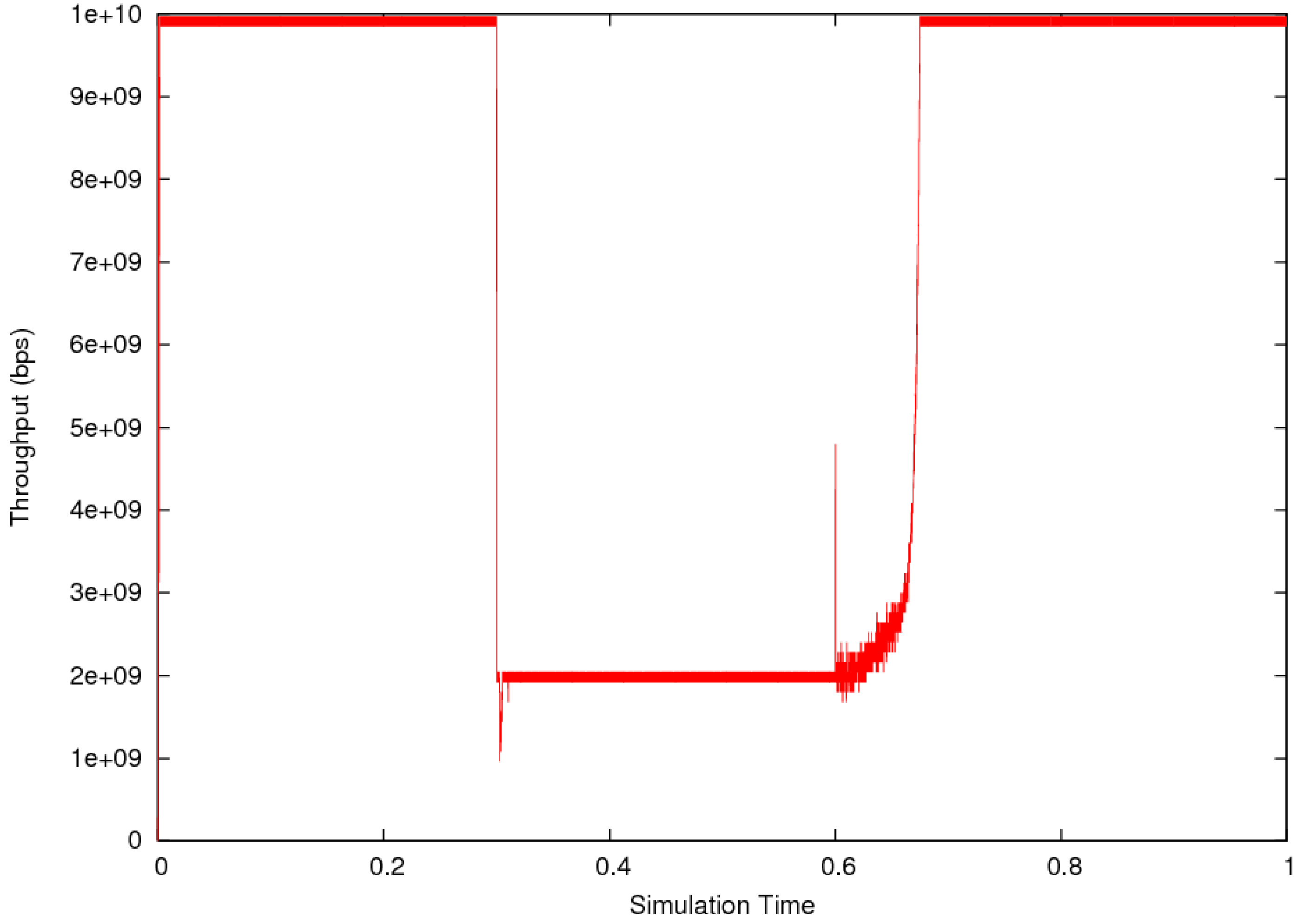
Simulation Parameters

- Traffic
  - i.i.d. Bernoulli arrivals
  - Uniform destination distribution (to all nodes except self)
  - Fixed frame size = 1500 B
- Switch
  - VOQ with 2.4MB shared mem
  - Partitioned memory per input, shared among all outputs
  - No limit on per-output memory usage
- Adapter
  - RLT: VOQ and single; RR service
  - One rate limiter per destination
  - Egress buffer size = 1500 KB,
  - Ingress buffer size = Unlimited
- QCN
  - W = 2.0
  - Q_EQ = 33 KB
  - GD = 0.0078125
  - Base marking: once every 150 KB
  - Margin of randomness: 30%
  - R_{unit} = 1 Mb/s
  - MIN_RATE = 10 Mb/s
  - BC_LIMIT = 150 KB
  - TIMER_PERIOD = 15 ms
  - R_AI = 5 Mbps
  - R_HAI = 50Mbps
  - FAST_RECOVERY_TH = 5
  - Quantized_Fb: 6 bits

Service Rate: 2.0Gbps
- Queue Size

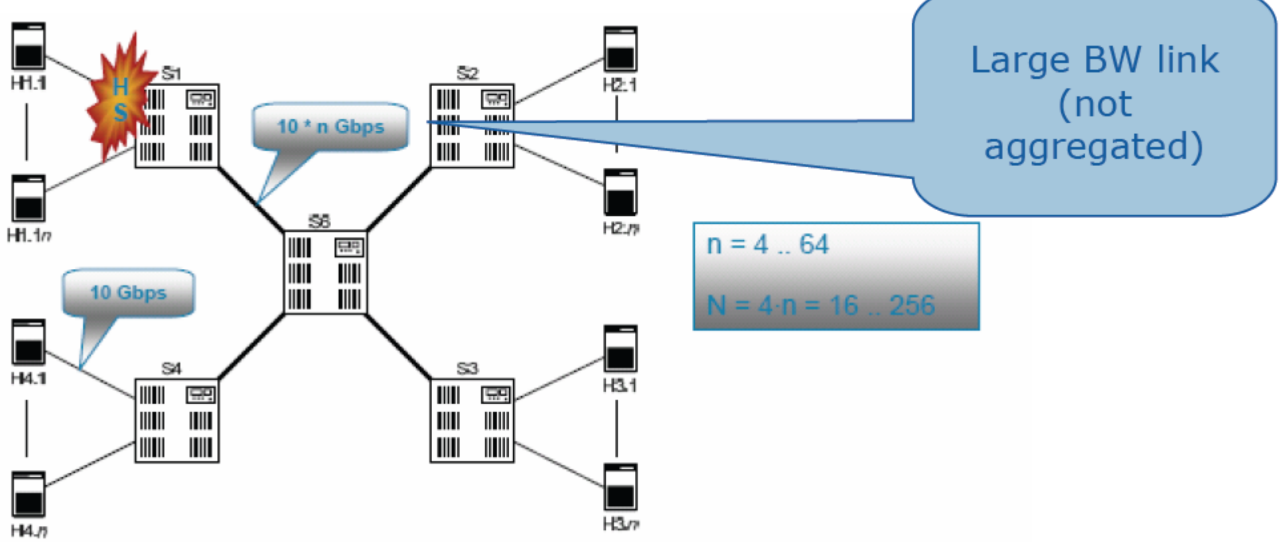


Service Rate: 2.0Gbps
- Throughput



Baseline #2

2. Multi-Hop Single HS Large Network



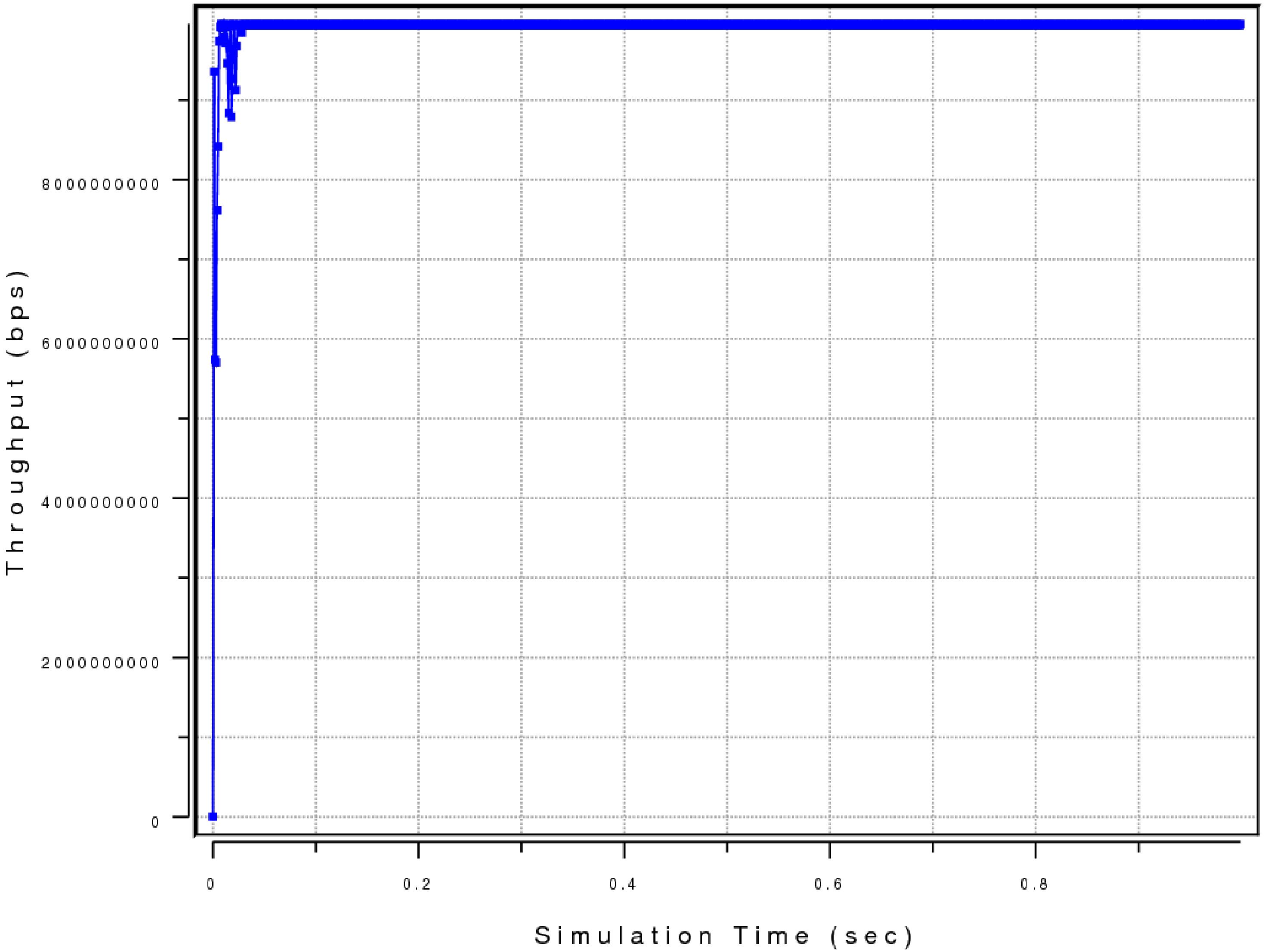
Workload:

- Load: H1.1 -- H4.n λ = 85%, Skewed Uniform
  - H1.1 is targeted with 2λ
  - All other nodes with λ (N-2)/(N-1)
- Congestion Point:
  - Node H1.1
  - HS degree = N
  - HS severity = 1.7 : 1

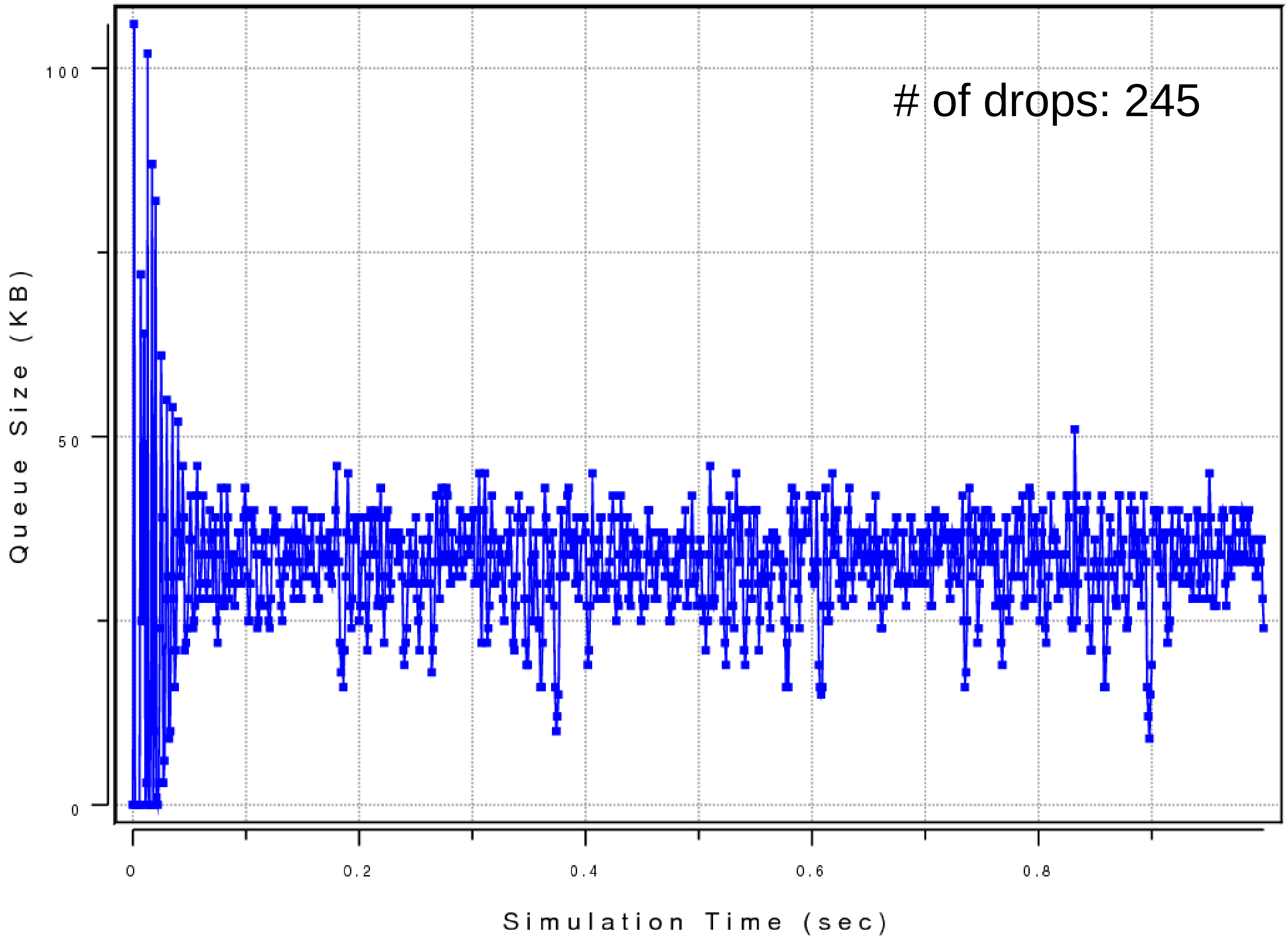
Required

Verdana regular 7pt.
Legal text goes here

Link Throughput
- The Congested Link (n = 4)



- Congested Queue Size (n = 4)



Summary

- QCN is a congestion control mechanism for data center Ethernet
- Congestion points send negative feedback signals
- Reaction point responds to congestion signals by cutting its rate, lack of such signals would trigger reaction point to increase its rate
  - fast recovery, additive increase and hyper additive increase.