Low-latency Network Monitoring via Oversubscribed Port Mirroring

Jeff Rasley, Brent Stephens,
Colin Dixon, Eric Rozner,
Wes Felter, Kanak Agarwal,
John Carter, Rodrigo Fonseca

BROWN

IBM Research

RICE

Self-Tuning Networks

Control Loop Examples

- Traffic Engineering
- Failure Detection

How fast can we do this?

Measurement

Control

Decision

Self-Tuning Networks

Control Loop Examples

- Traffic Engineering
- Failure Detection

How fast can we do this?

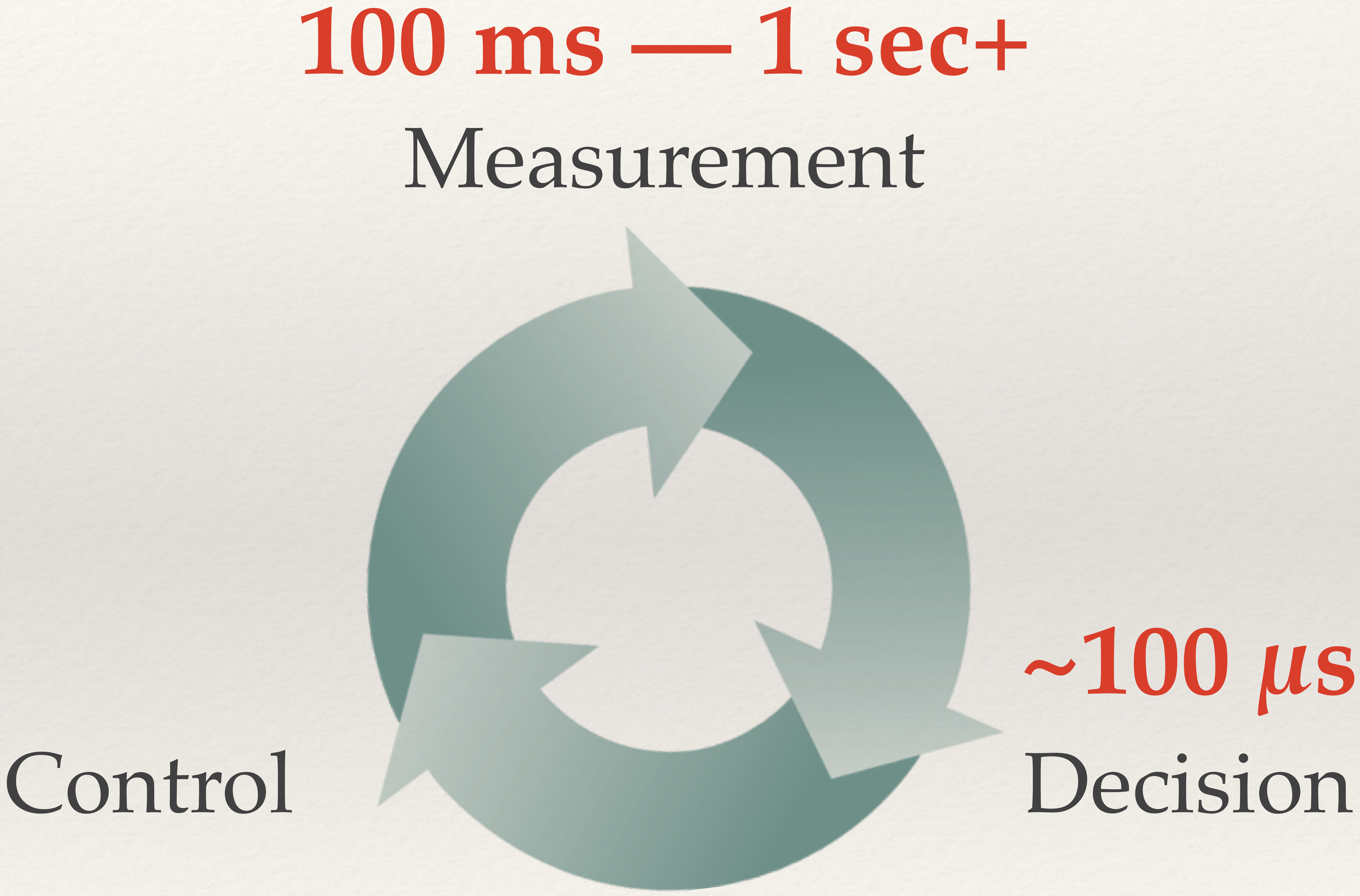
100 ms — 1 sec+

Self-Tuning Networks

Control Loop Examples

- Traffic Engineering
- Failure Detection

How fast can we do this?



Self-Tuning Networks

100 ms — 1 sec+
Measurement

Control Loop Examples

- Traffic Engineering
- Failure Detection

How fast can we do this?

~10 ms
Control

~100 μs
Decision

Self-Tuning Networks

Control Loop Examples

- Traffic Engineering
- Failure Detection

100 ms — 1 sec+
Measurement

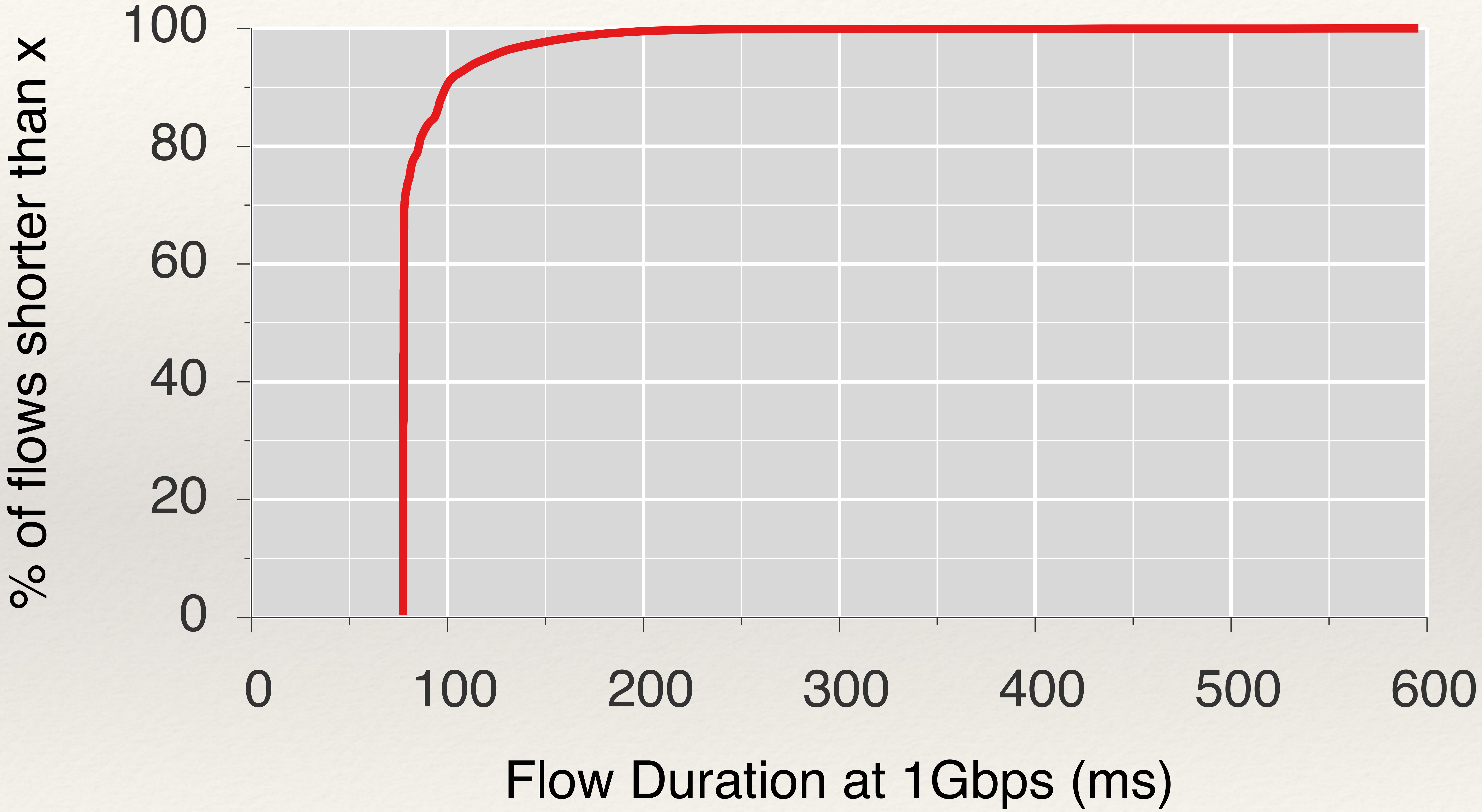
How fast can we do this?

~10 ms
Control

~100 μs
Decision

Motivation for Faster Control Loops

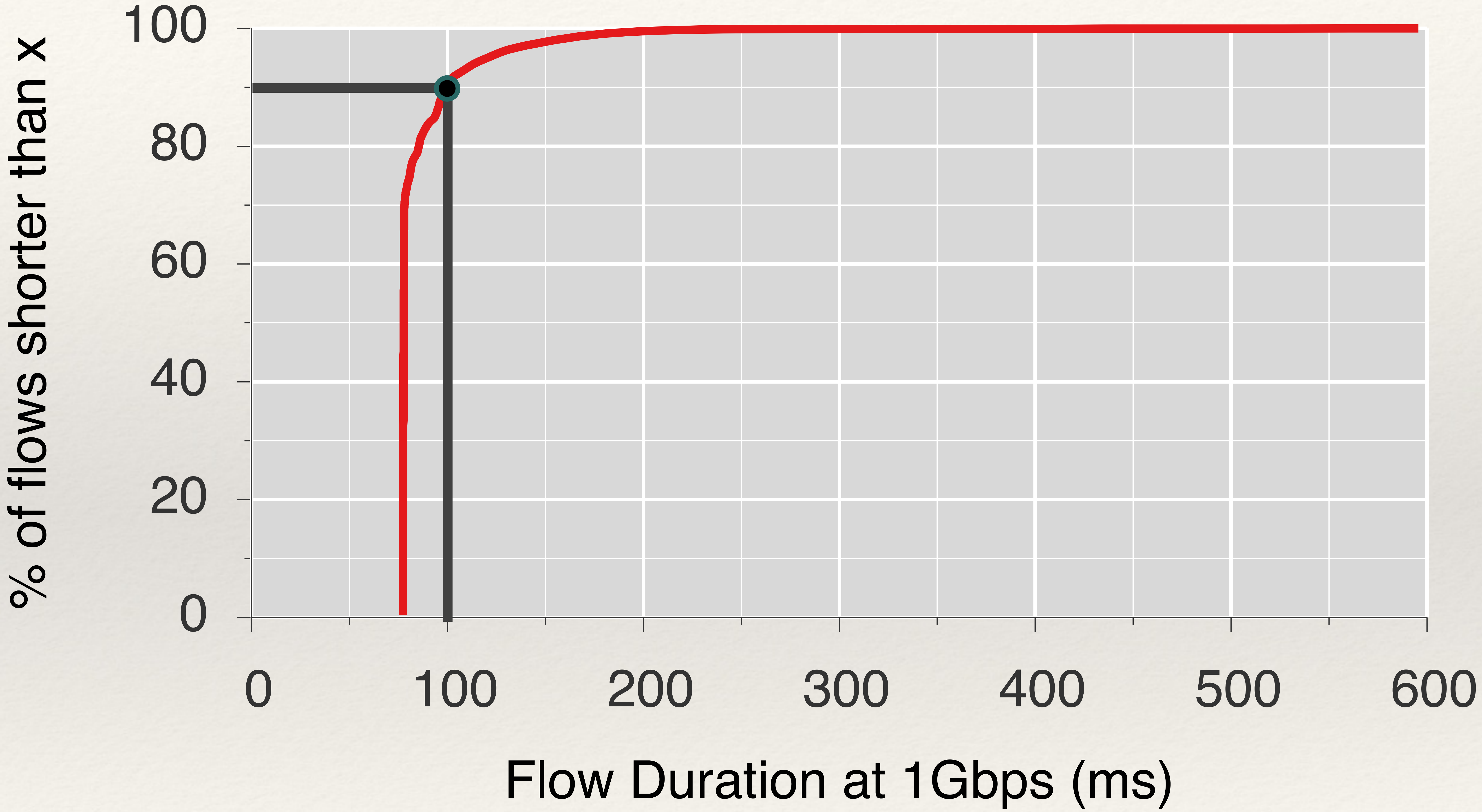
This gets much worse as you go from 1 Gbps –> 10 Gbps



Background TCP flows, Microsoft data center
DCTCP, Alizadeh et al. Sigcomm '10

Motivation for Faster Control Loops

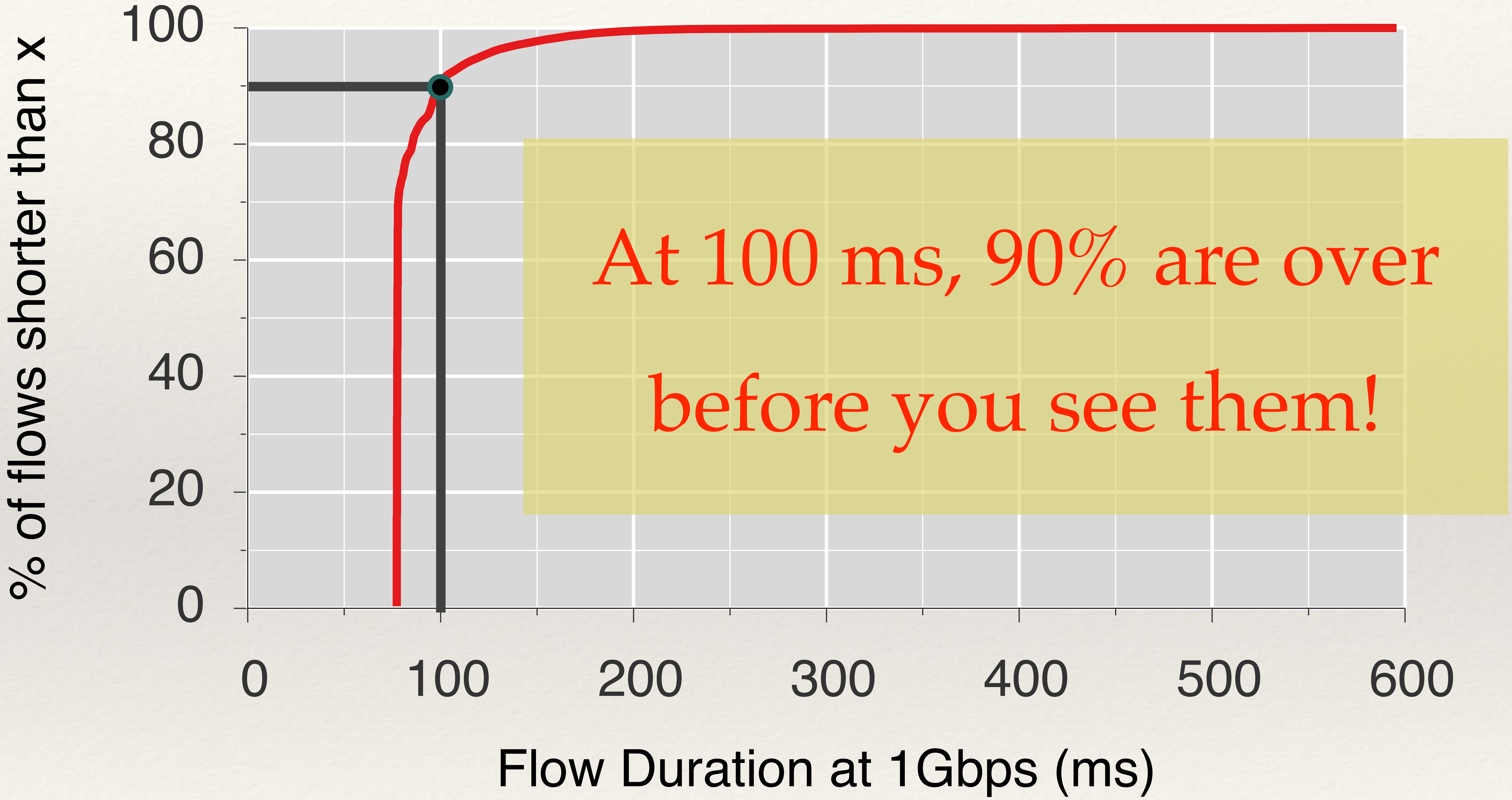
This gets much worse as you go from 1 Gbps –> 10 Gbps



Background TCP flows, Microsoft data center
DCTCP, Alizadeh et al. Sigcomm '10

Motivation for Faster Control Loops

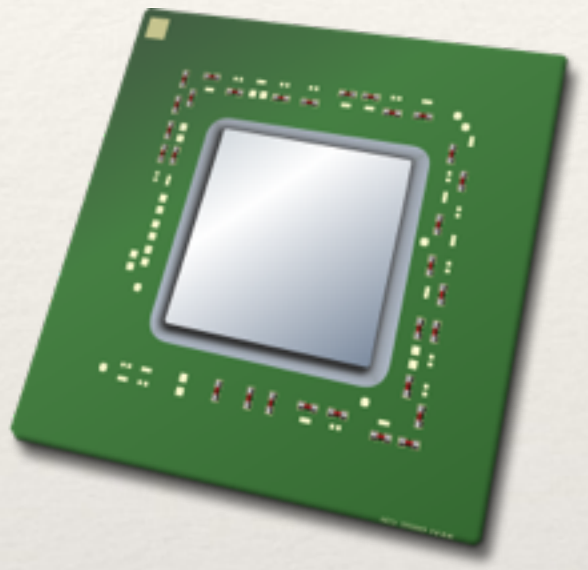
This gets much worse as you go from 1 Gbps –> 10 Gbps

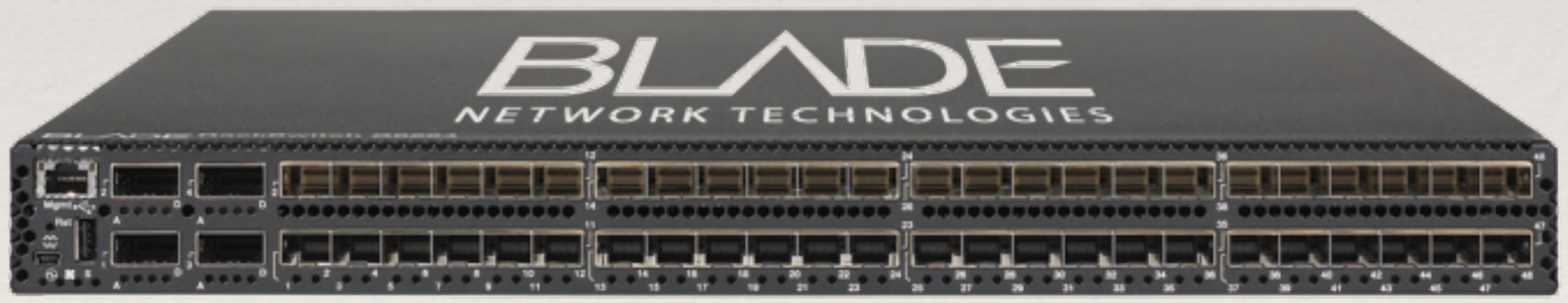


Background TCP flows, Microsoft data center
DCTCP, Alizadeh et al. Sigcomm '10

Why is Measurement Slow?

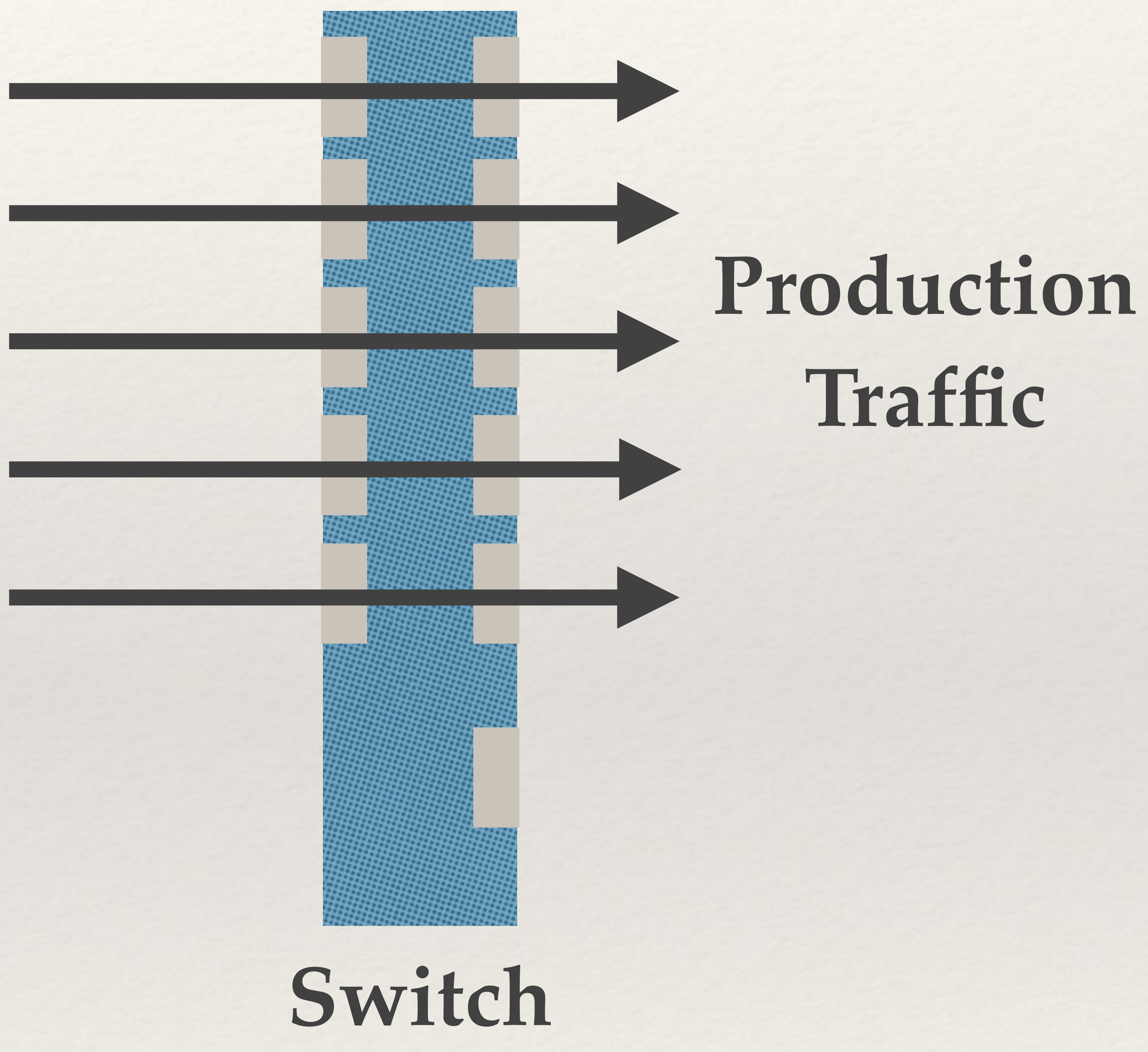
- Traditionally, this didn't need to be fast
- Control plane CPUs are typically slow
  - Sampling or port counter polling
- Is this likely to get better? Maybe
- Faster control plane CPUs could help, still a big gap between CPUs and ASICs





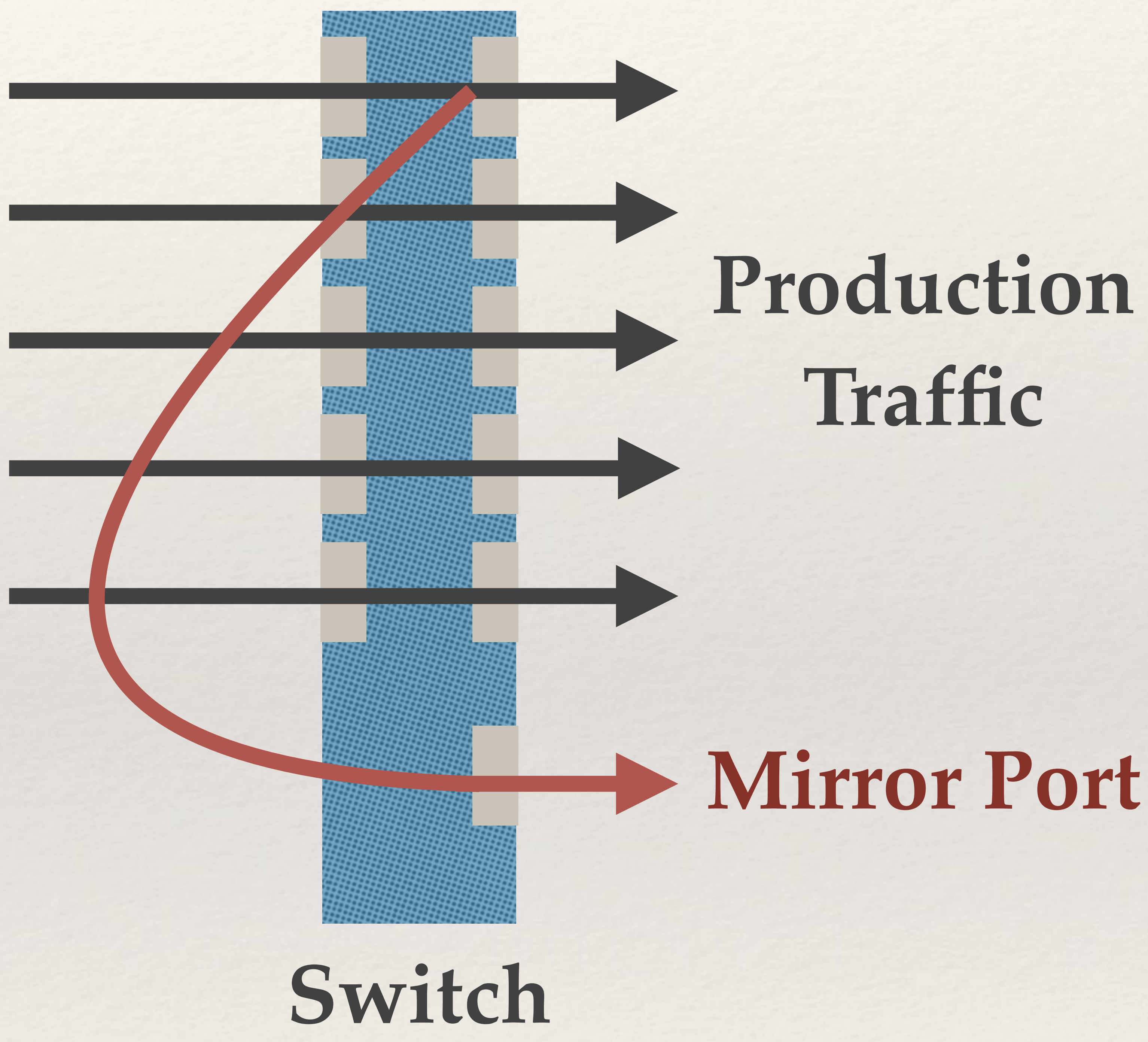
Our Solution: Abuse Port Mirroring

- Modern switches support port-mirroring
  - Copies all packets e.g. going out a port to a designated mirror port
- We abuse port mirroring to radically increase the number of samples/sec we get from a switch
- We mirror all ports to a single mirror port
  - Oversubscription approximates sampling (in the data plane) at much higher rates



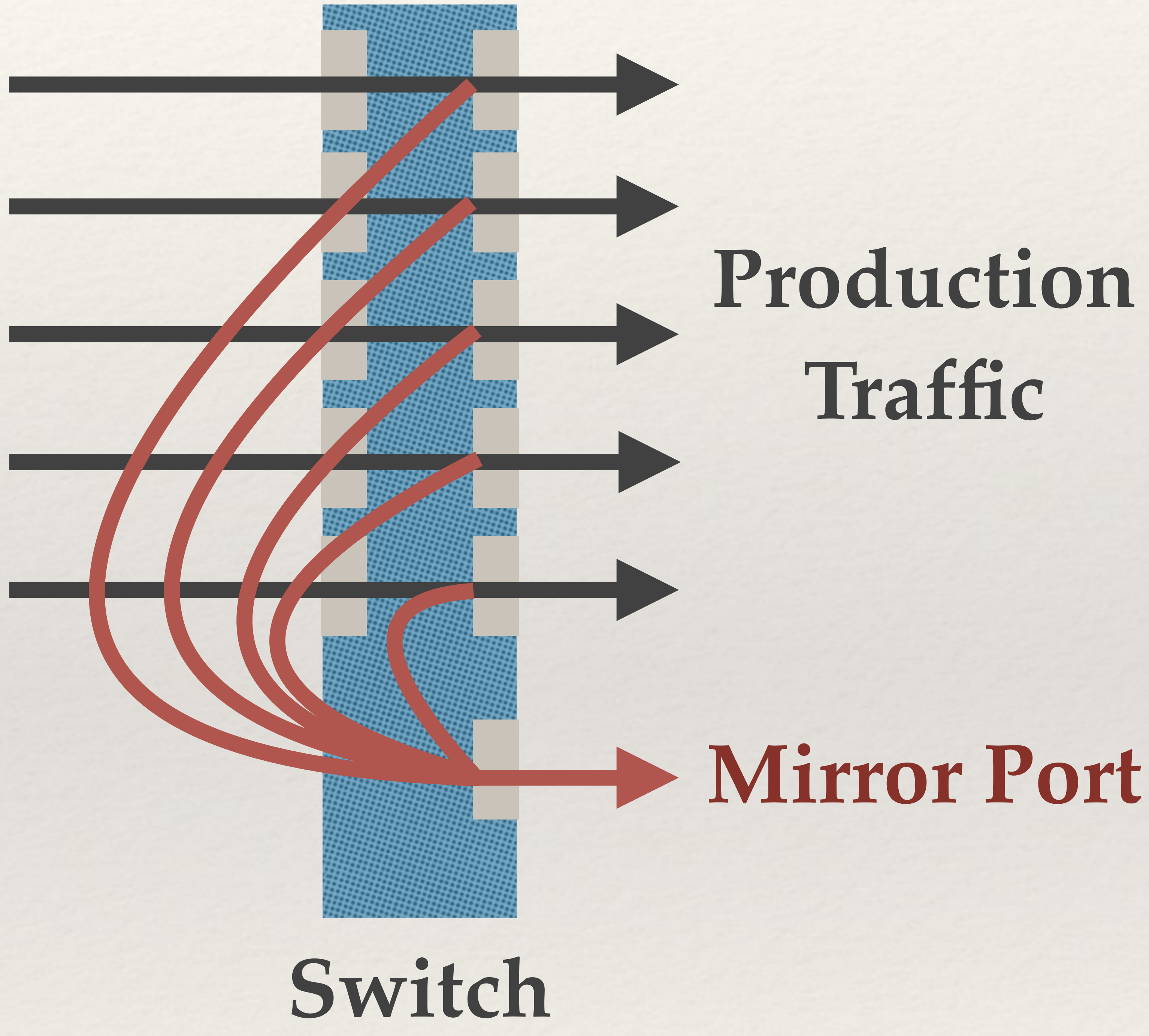
Our Solution: Abuse Port Mirroring

- Modern switches support port-mirroring
  - Copies all packets e.g. going out a port to a designated mirror port
- We abuse port mirroring to radically increase the number of samples/sec we get from a switch
- We mirror all ports to a single mirror port
  - Oversubscription approximates sampling (in the data plane) at much higher rates



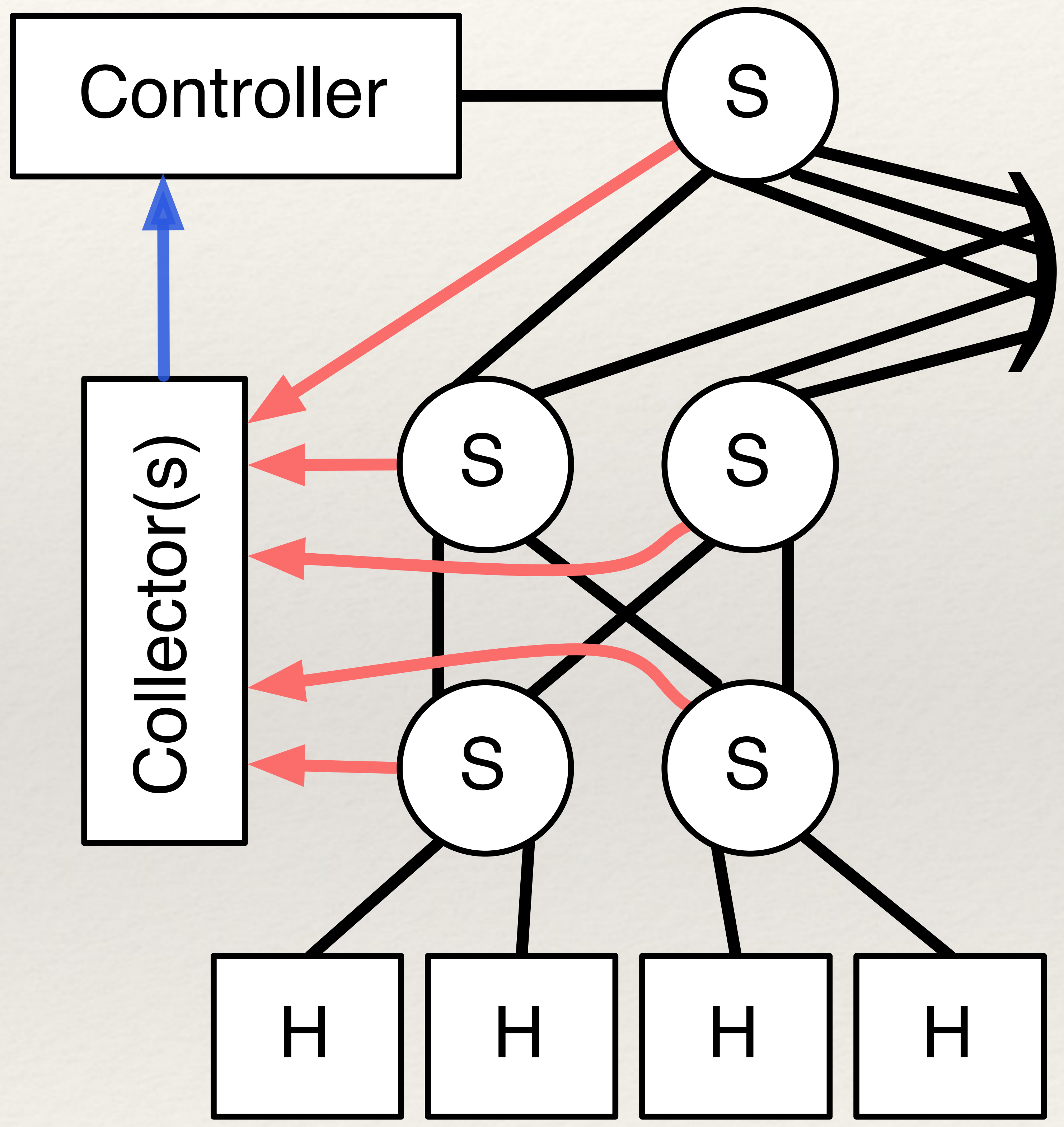
Our Solution: Abuse Port Mirroring

- Modern switches support port-mirroring
  - Copies all packets e.g. going out a port to a designated mirror port
- We abuse port mirroring to radically increase the number of samples/sec we get from a switch
- We mirror all ports to a single mirror port
  - Oversubscription approximates sampling (in the data plane) at much higher rates



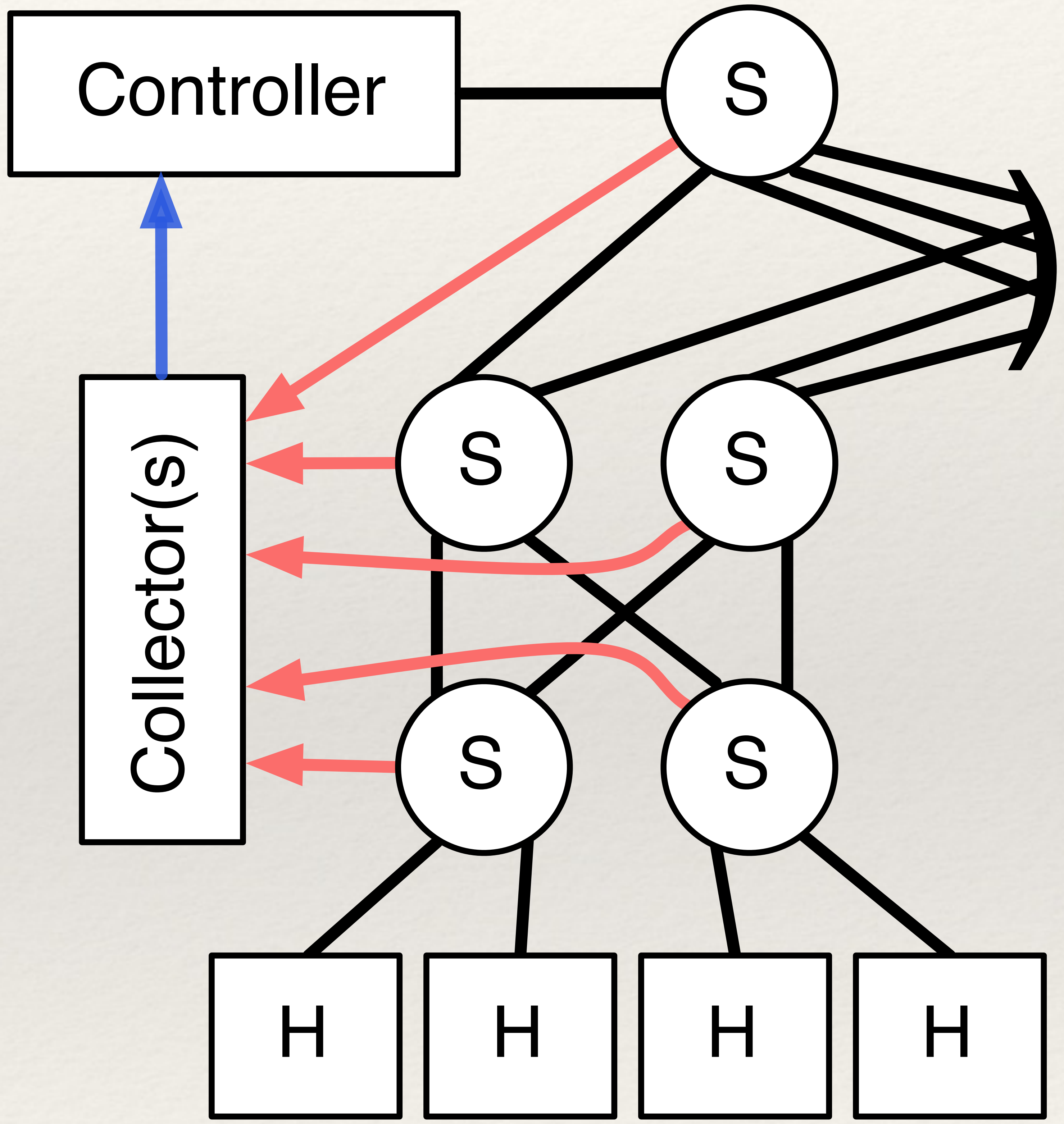
Architecture

- A set of collectors receives a stream of samples from mirror ports
  - Netmap or Intel DPDK for fast processing
- Reconstruct flow information across all flows in the network
  - e.g. flow throughput and port congestion
- Collectors can interact with an SDN controller to implement various applications
  - e.g. traffic engineering



What Can Go Wrong?

- Lose input/output port information from packets
  - Recover meta-data about packets by sharing topology state from the controller
- When mirror port fills, its drop policy is unknown thus making it hard to calculate throughout
  - Rate estimation via TCP sequence numbers
- Oversubscribed port may occupy switch buffer space, taking away from production traffic
  - Indeed, buffers were reduced. Latency of production traffic decreased. Negligible increase in packet loss (~0.1%).

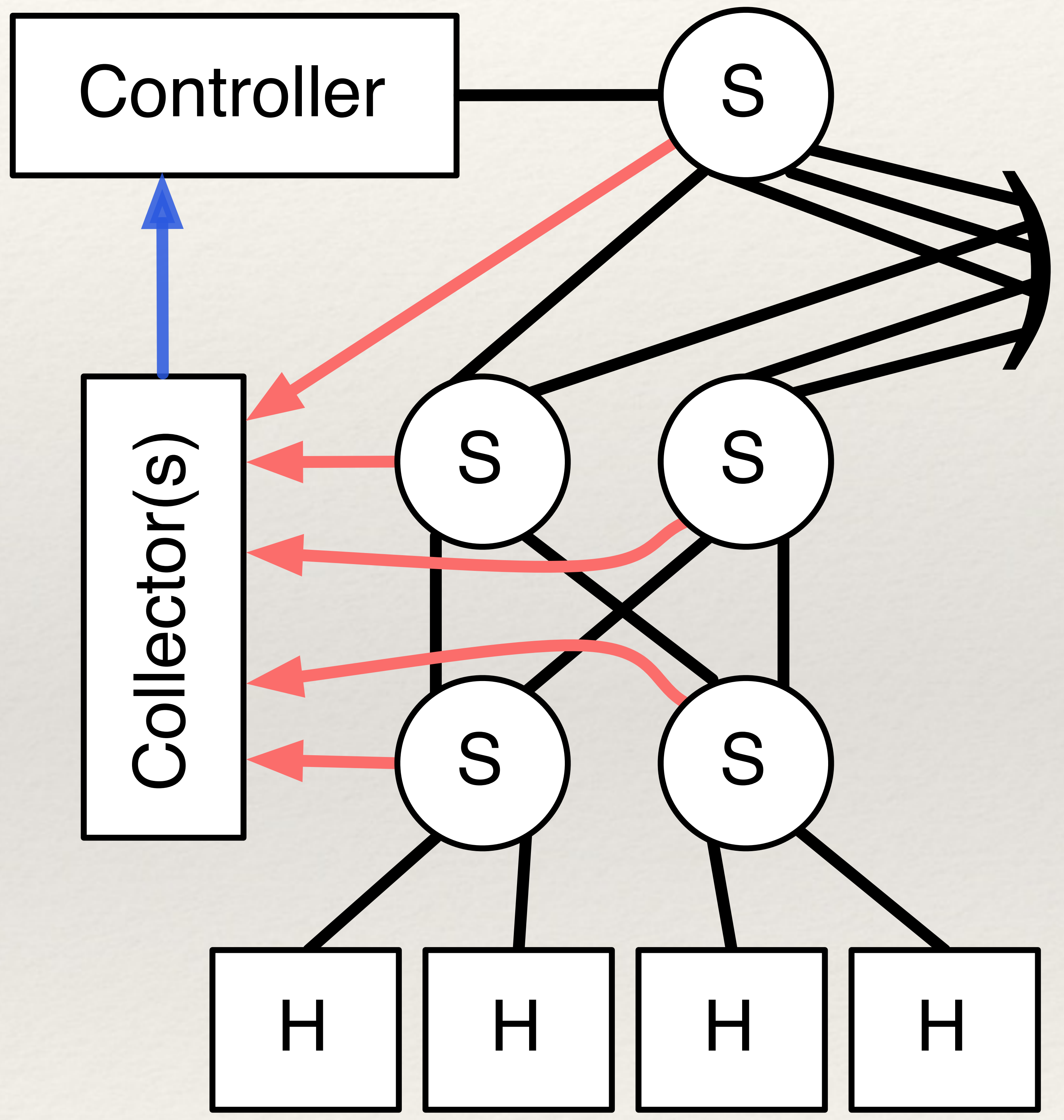


What Can Go Wrong?

- Lose input/output port informa
  - Recover me
    state
- When m
  making it
  - Rate est
- Oversubscri py switch buffer space,
  taking away roduction traffic
  - Indeed, buffers were reduced. Latency of production traffic decreased. Negligible increase in packet loss (~0.1%).

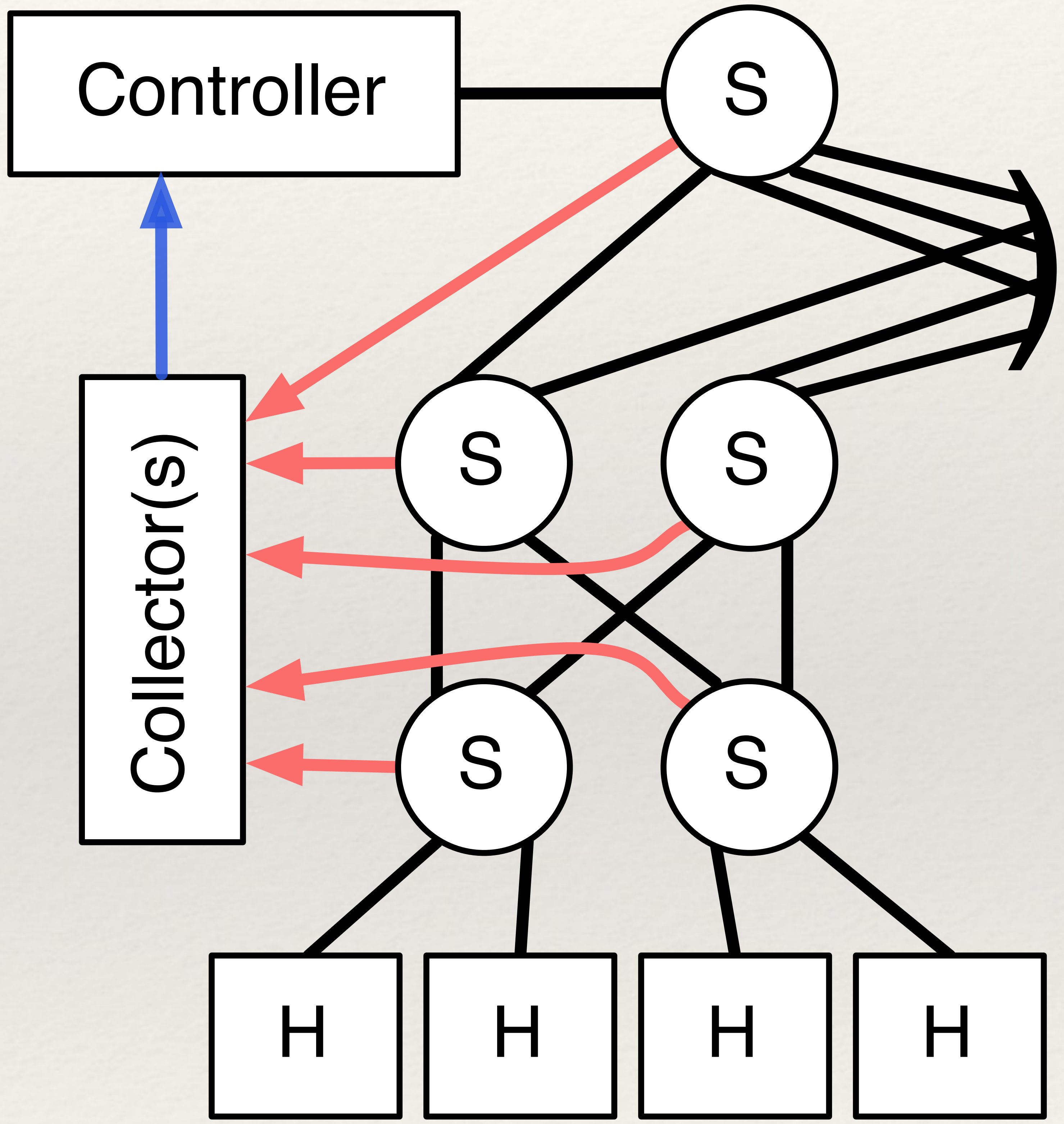
What are you crazy?!

By oversubscribing a mirror port there's a possibility of degrading production traffic.
— Cisco

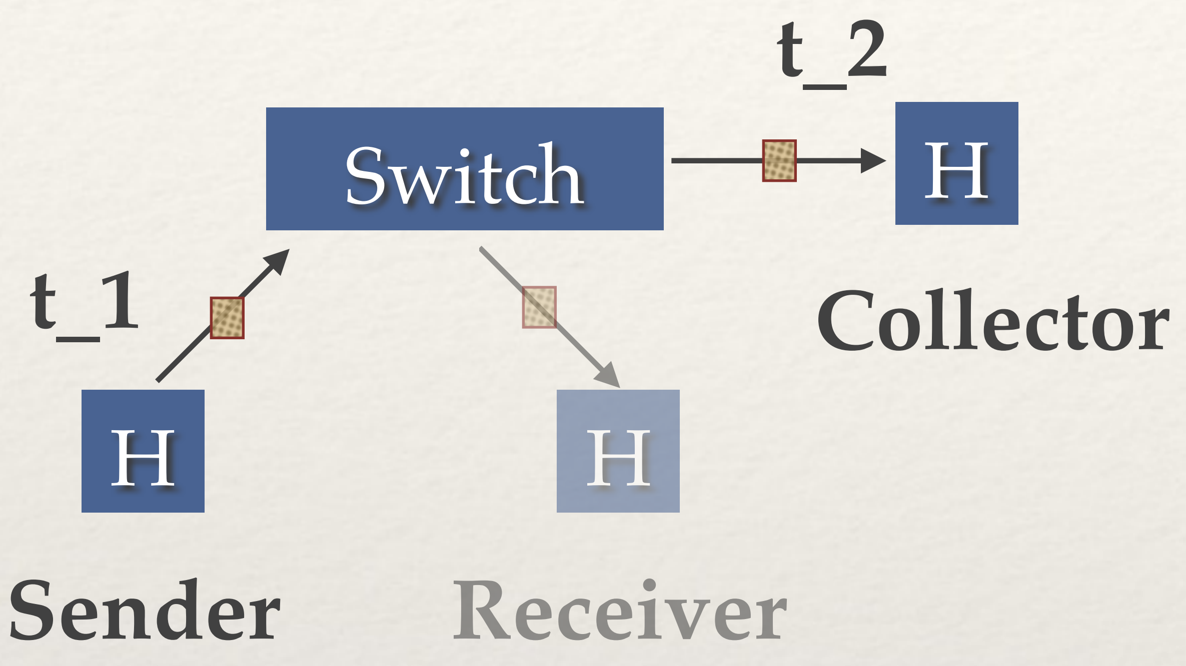


What Can Go Wrong?

- Lose input/output port information from packets
  - Recover meta-data about packets by sharing topology state from the controller
- When mirror port fills, its drop policy is unknown thus making it hard to calculate throughout
  - Rate estimation via TCP sequence numbers
- Oversubscribed port may occupy switch buffer space, taking away from production traffic
  - Indeed, buffers were reduced. Latency of production traffic decreased. Negligible increase in packet loss (~0.1%).

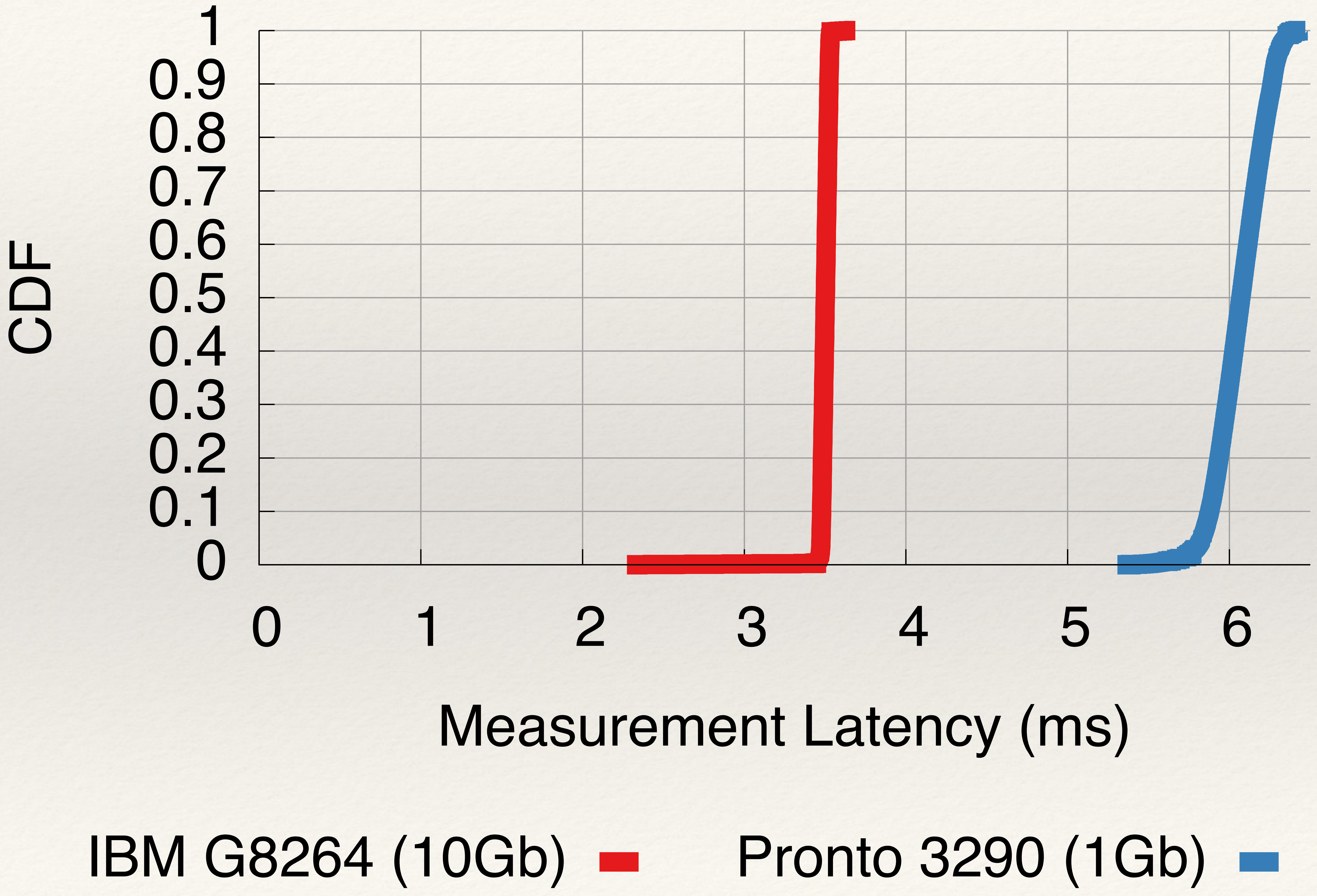


Results: Sample Latency (high congestion)



Latency = t_2 - t_1

Low Congestion Sample Latency: 75–150 μs



What Can You Do With This? TE!

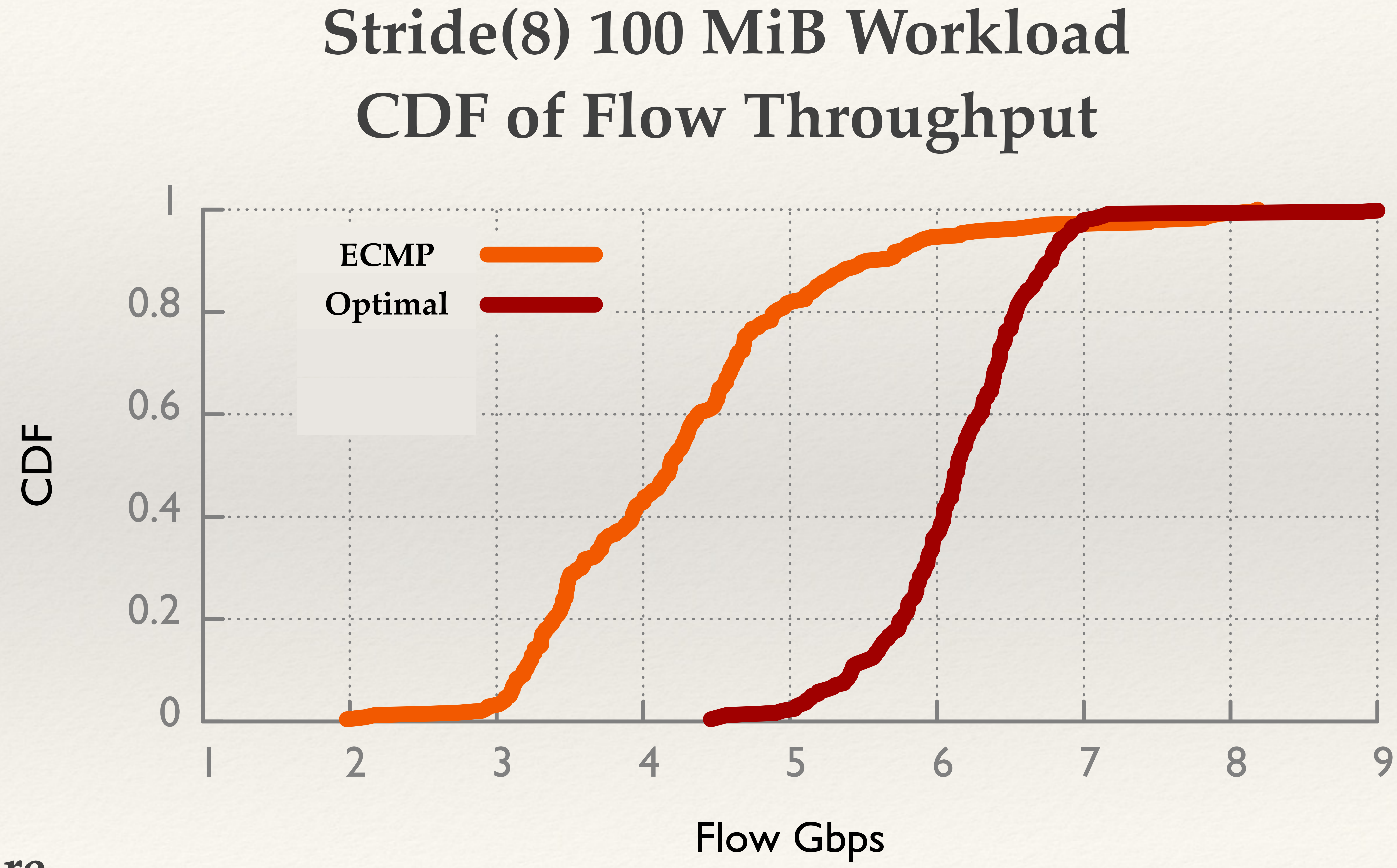
Setup

- 16 hosts
- k=4 Fat Tree

Stride(8)

Permutation of hosts, such that each flow traverses the core

IBM G8264 (10 Gb) Hardware



What Can You Do With This? TE!

Setup

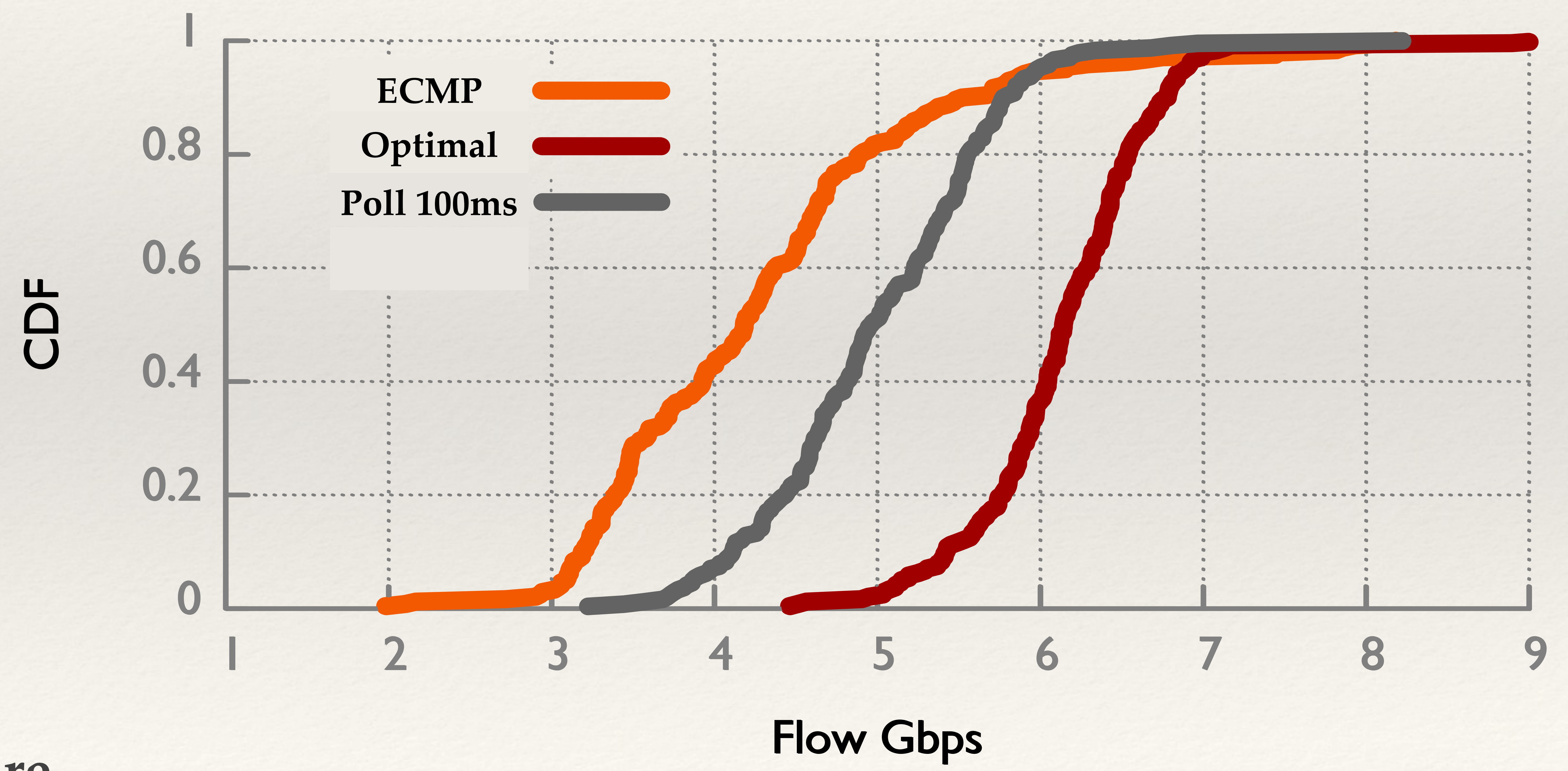
- 16 hosts
- k=4 Fat Tree

Stride(8)

Permutation of hosts, such that each flow traverses the core

IBM G8264 (10 Gb) Hardware

Stride(8) 100 MiB Workload CDF of Flow Throughput



What Can You Do With This? TE!

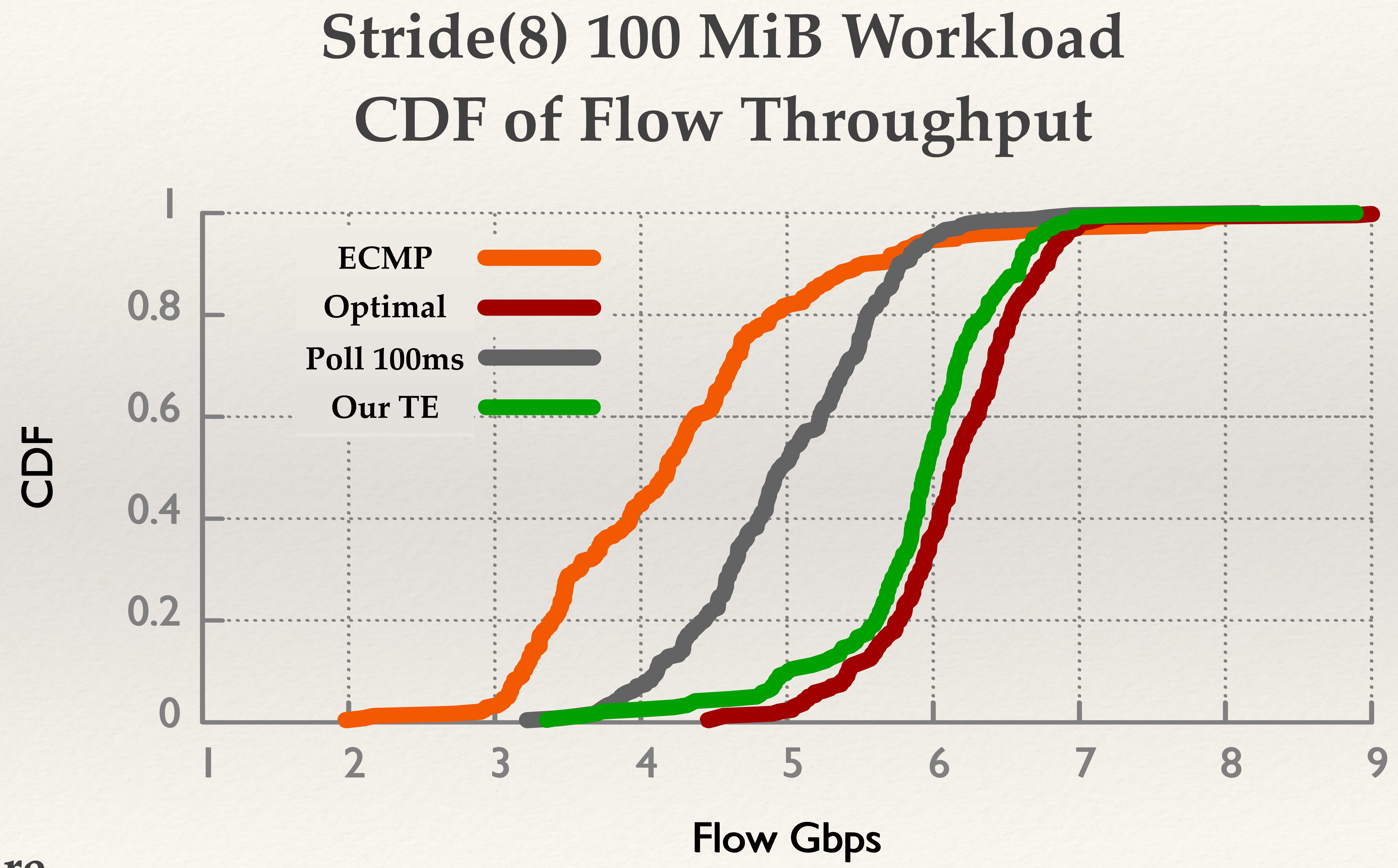
Setup

- 16 hosts
- k=4 Fat Tree

Stride(8)

Permutation of hosts, such that each flow traverses the core

IBM G8264 (10 Gb) Hardware

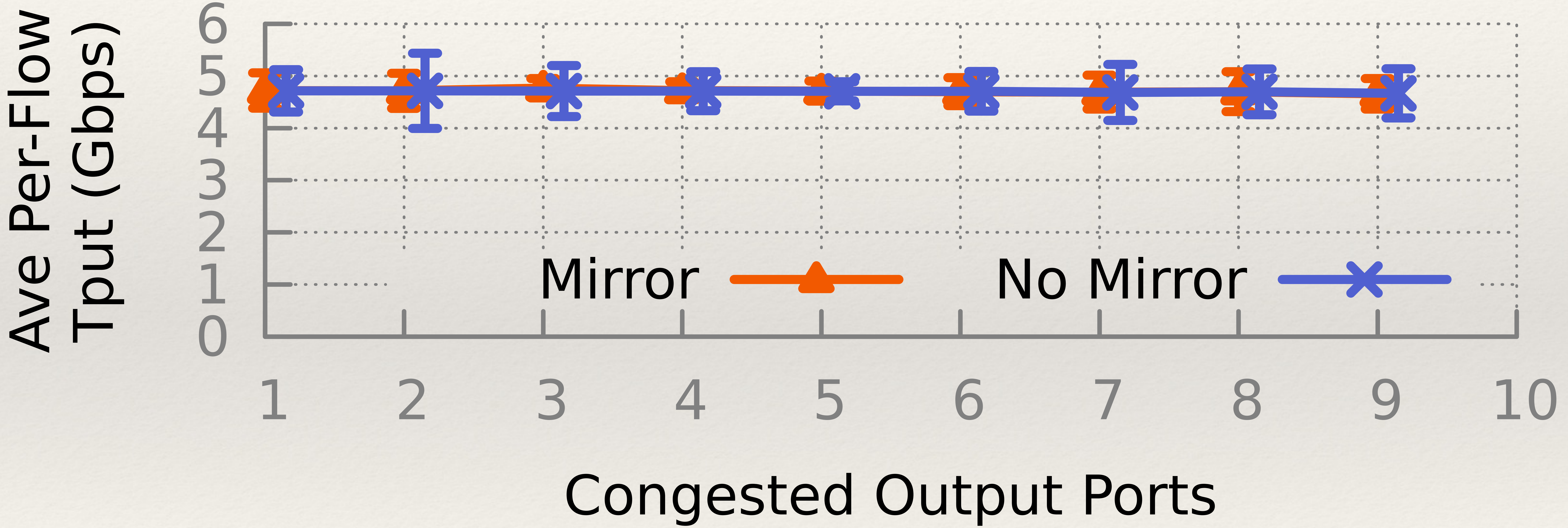


Conclusion

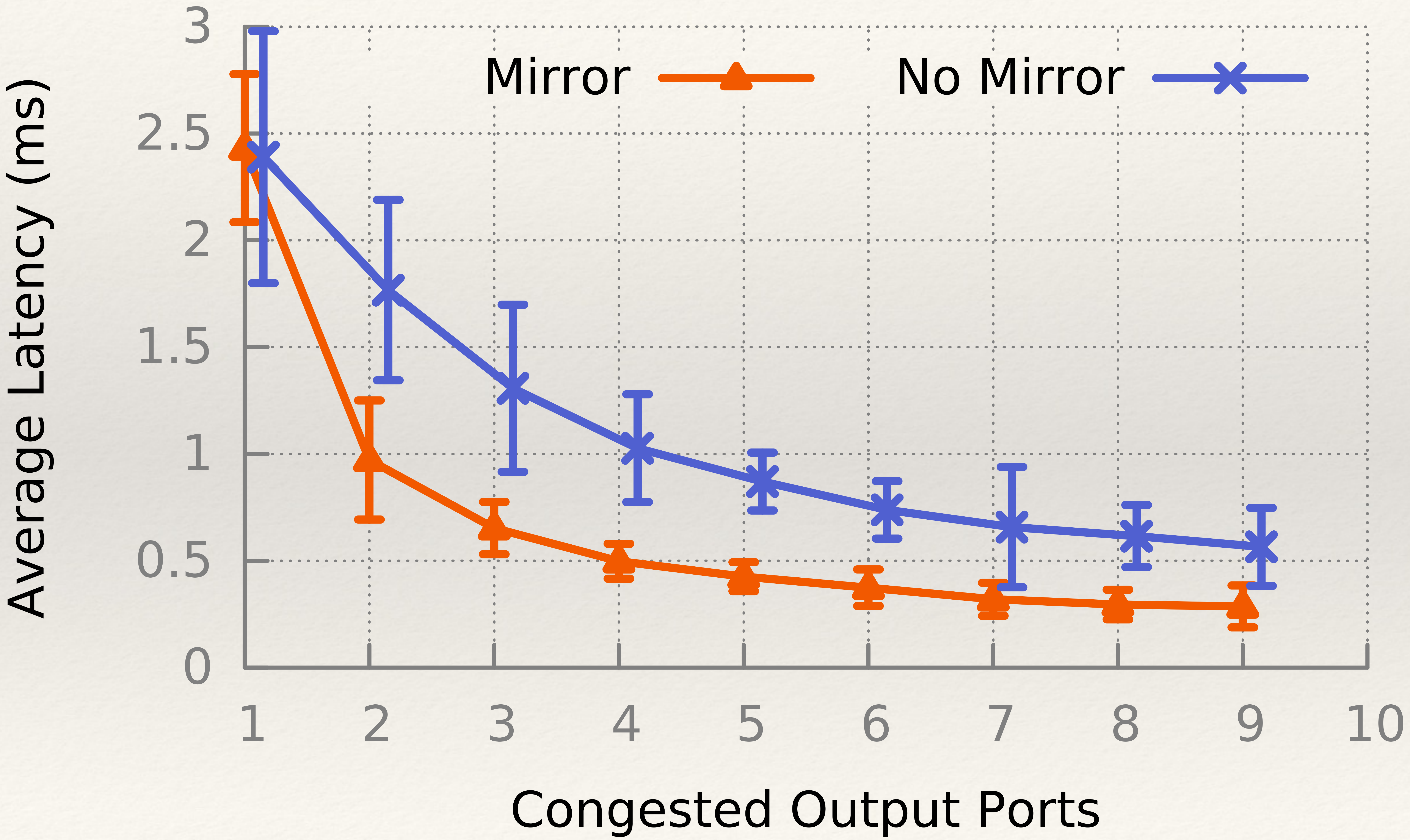
- Using oversubscribed port mirroring we get **~1 million samples / sec**.
- We get sampling latencies between **100 μs – 6ms** on real hardware, today.
    - We improve this by 3–4 orders of magnitude, the state of the art is 100 ms – 1 sec+

Questions? Thank you!

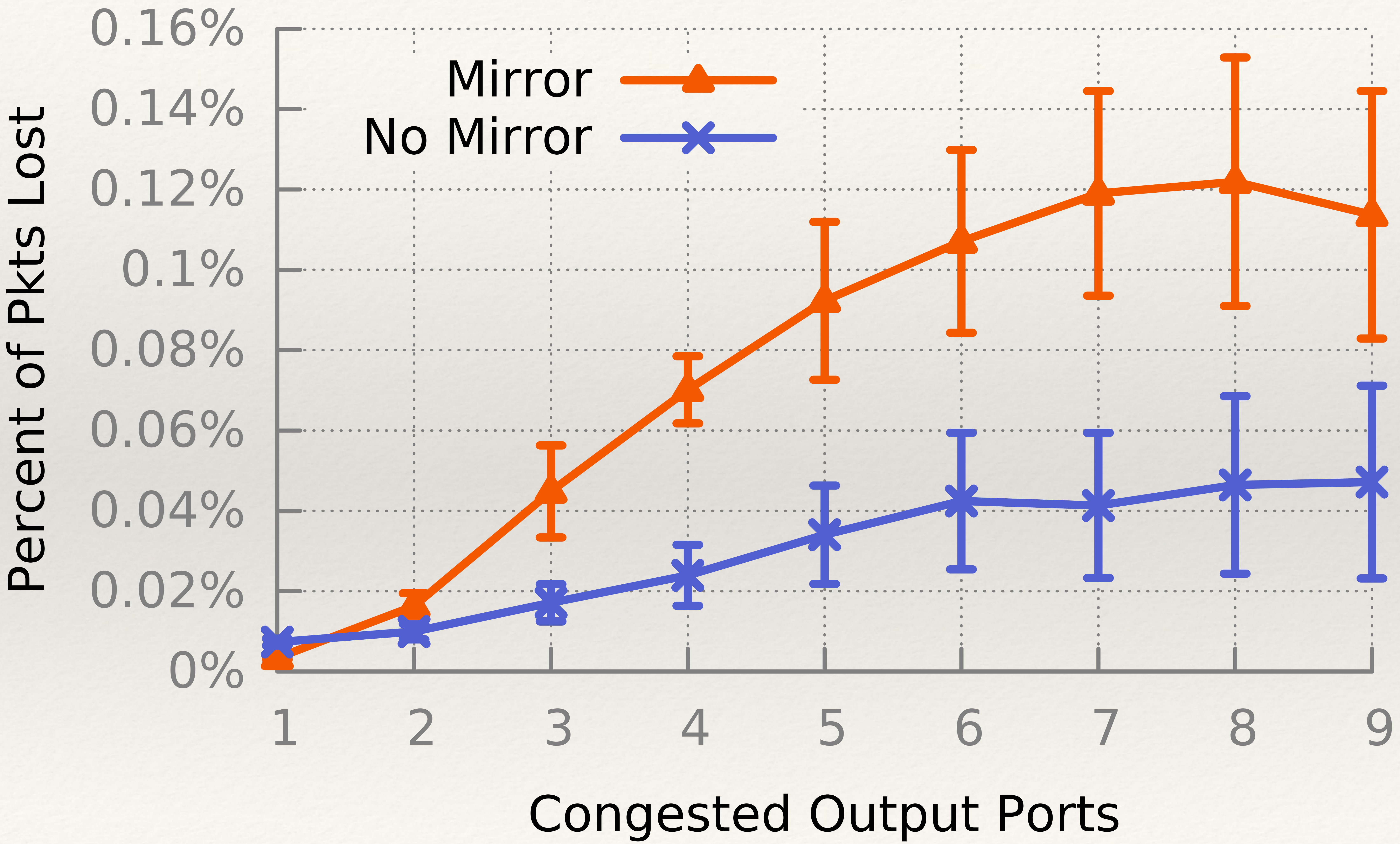
Production Traffic Throughput



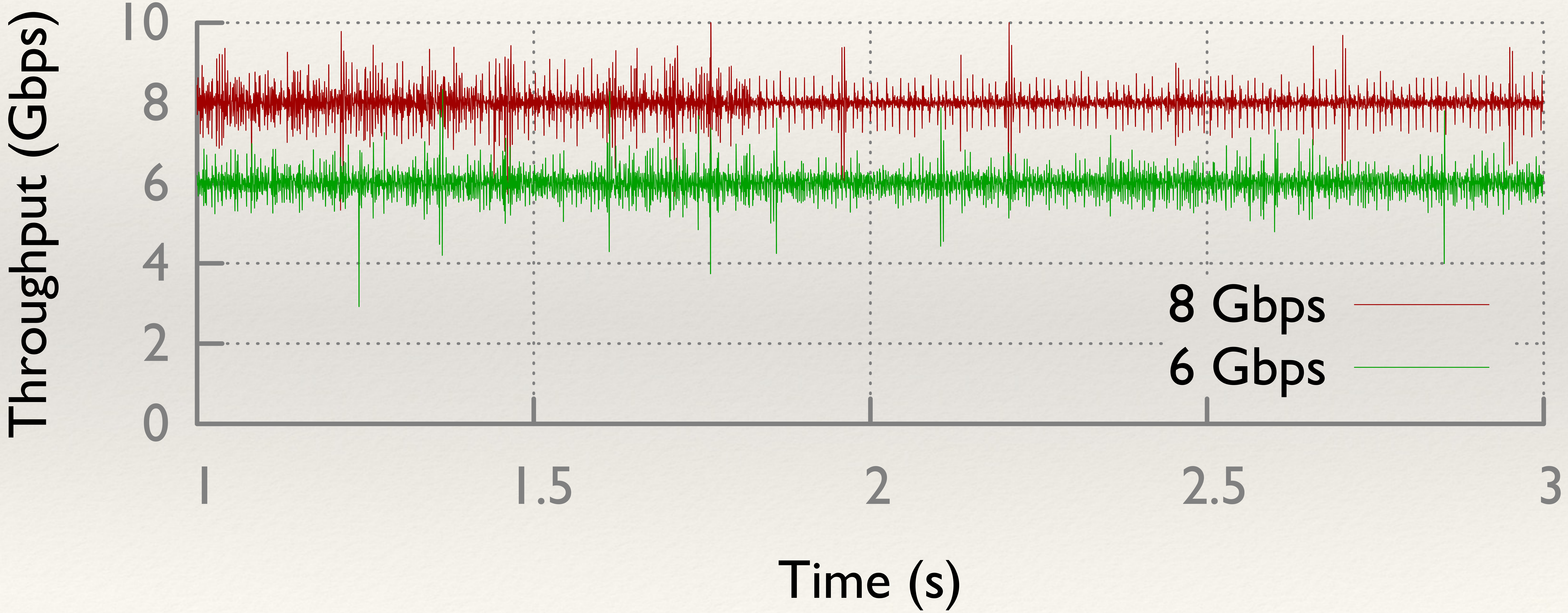
Production Traffic Latency



Production Traffic Packet Loss



Flow Rate Estimation



Rate Estimation Error