Design and Implementation for Demand-Responsive Cross-Layer Networking (part II)

Paper review on optical networking in ACM/USENIX community

Zhizhen Zhong

Tsinghua University & UC Davis

zhongzz14@mails.tsinghua.edu.cn , zzzhong@ucdavis.edu

11 May 2018

Networks Lab Group Meeting

Tsinghua University

UCDAVIS UNIVERSITY OF CALIFORNIA

NSDI'18 Overview

- NSDI'18 Overview
- Overview of optical networking in ACM and USENIX
- Optical networking paper review
- Some takeaways

NSDI’18 @ Renton, WA

- NSDI: Networked System Design and Implementation
- Organized by USENIX
  - The **USENIX** Association is the Advanced Computing Systems Association. It was founded in 1975 under the name “Unix Users Group,” focusing primarily on the study and development of Unix and similar systems.
- NSDI’18, 40 papers in 12 topics: new hardware, distributed systems, **traffic management**, NFV and hardware, web and video, performance isolation and scaling, **congestion control**, cloud, diagnosis, **fault tolerance**, **physical layer**, configuration management.

NSDI’18 research spotlights

[NetChain: Scale-Free Sub-RTT Coordination]

Xin Jin, *Johns Hopkins University;* Xiaozhou Li, *Barefoot Networks;* Haoyu Zhang, *Princeton University;* Nate Foster, *Cornell University;* Jeongkeun Lee, *Barefoot Networks;*Robert Soulé, *Università della Svizzera italiana;* Changhoon Kim, *Barefoot Networks;* Ion Stoica, *UC Berkeley*

Using programmable switches to design new coordination protocol

[zkLedger: Privacy-Preserving Auditing for Distributed Ledgers]

Neha Narula, *MIT Media Lab;* Willy Vasquez, *University of Texas at Austin;* Madars Virza, *MIT Media Lab*

Distributed ledgers for financial systems enabled by networking

Fastpass: A Centralized “Zero-Queue” Datacenter Network

Jonathan Perry, Amy Ousterhout, Hari Balakrishnan, Devavrat Shah, and Hans Fugal. "Fastpass: A centralized zero-queue datacenter network." *ACM SIGCOMM* 2014.

清華大學 Tsinghua University

UCDAVIS UNIVERSITY OF CALIFORNIA

Fastpass concept

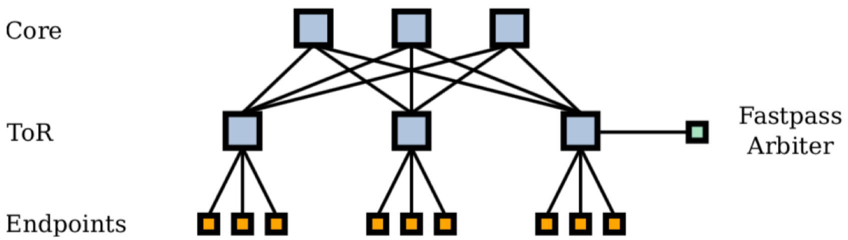


Figure 1: Fastpass arbiter in a two-tier network topology.

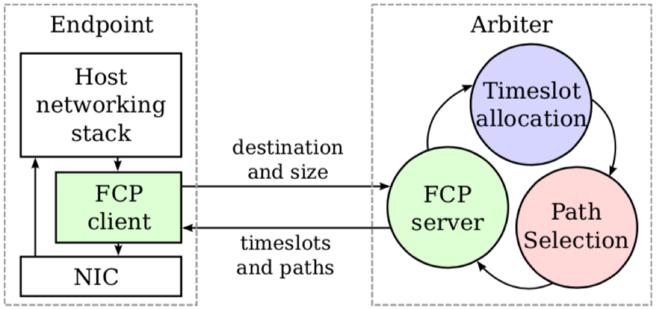


Figure 2: Structure of the arbiter, showing the timeslot allocator, path selector, and the client-arbiter communication.

- In Fastpass, a logically centralized arbiter controls all network transfers.
- Because the arbiter knows about all current and scheduled trans- fers, it can choose timeslots and paths that yield the "*zero-queue*" property: the arbiter arranges for each packet to arrive at a switch on the path just as the next link to the destination becomes available.
- Fastpass incorporates two fast algorithms: the first determines the time at which each packet should be transmit- ted, while the second determines the path to use for that packet.
- Network scale time synchronization is required.

Tsinghua University

UC DAVIS
UNIVERSITY OF CALIFORNIA

Path selection

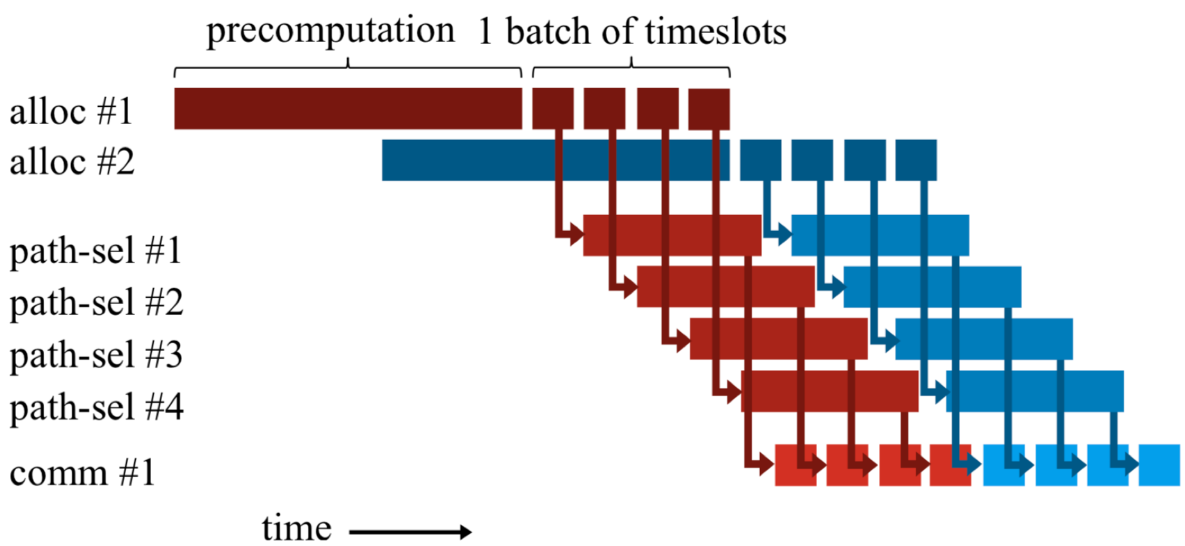


Figure 6: Multicore allocation: (1) allocation cores assign packets to timeslots, (2) path selection cores assign paths, and (3) communication cores send allocations to endpoints.

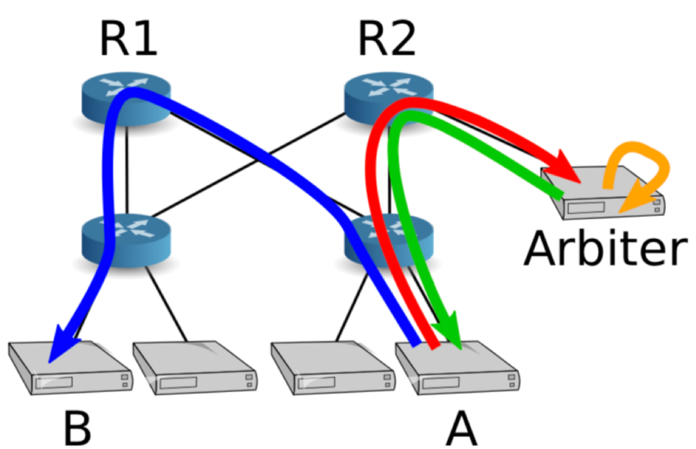
Tsinghua University

UCDAVIS UNIVERSITY OF CALIFORNIA

Path selection

Example: Packet from A to B

5µs	A → Arbiter	"A has 1 packet for B"
1-20µs	Arbiter	timeslot allocation & path selection
15µs	Arbiter → A	"@t=107: A → B through R1"
no queuing	A → B	sends data



Tsinghua University 清華大學

UC DAVIS UNIVERSITY OF CALIFORNIA

Queueing performance

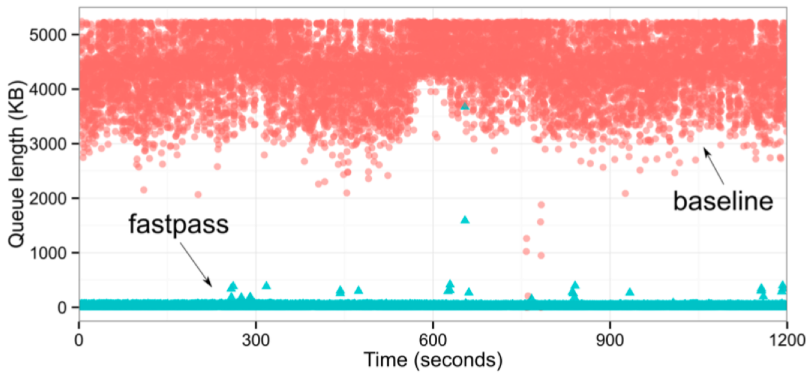


Figure 7: Switch queue lengths sampled at 100ms intervals on the top-of-rack switch. The diagram shows measurements from two different 20 minute experiments: baseline (red) and Fastpass (blue). Baseline TCP tends to fill switch queues, whereas Fastpass keeps queue occupancy low.

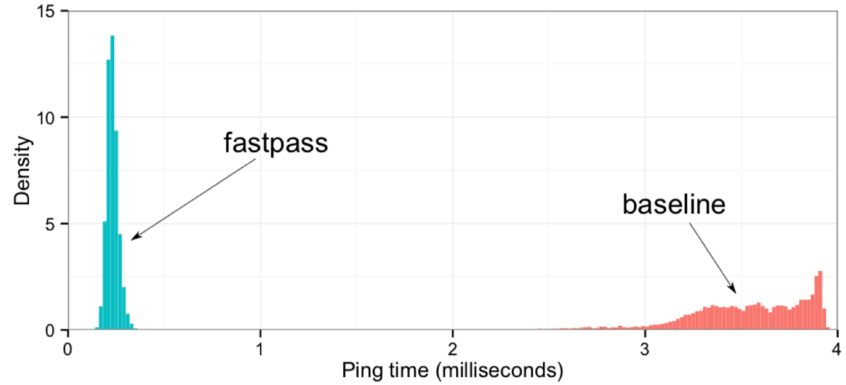


Figure 8: Histogram of ping RTTs with background load using Fastpass (blue) and baseline (red). Fastpass's RTT is 15.5× smaller, even with the added overhead of contacting the arbiter.

- Fastpass reduces the median switch queue occupancy from 4.35 Megabytes in the baseline to just 18 kilobytes with Fastpass, a reduction of a factor of 242×
- Fastpass reduces the end-to-end round-trip time (RTT) for in- teractive traffic when the network is heavily loaded by a factor of 15.5×, from a median of 3.56 ms to 230 μs

throughput

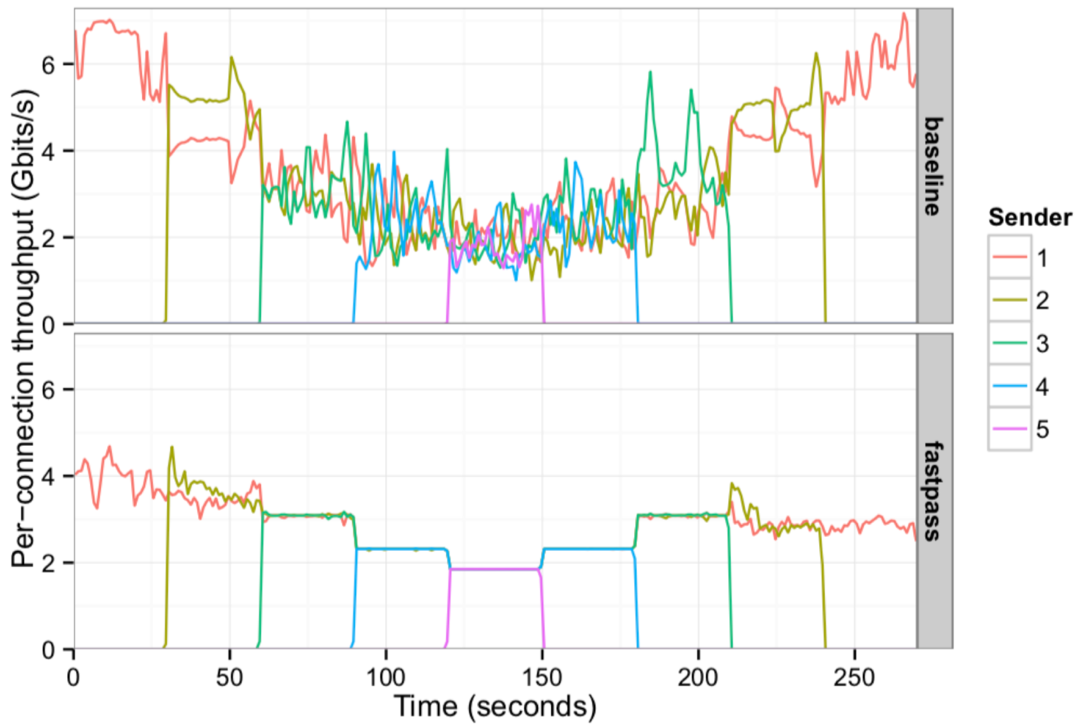


Figure 9: Each connection's throughput, with a varying number of senders. Even with 1s averaging intervals, baseline TCP flows achieve widely varying rates. In contrast, for Fastpass (bottom), with 3, 4, or 5 connections, the throughput curves are on top of one another. The Fastpass max-min fair timeslot allocator maintains fairness at fine granularity. The lower one- and two-sender Fastpass throughput is due to Fastpass qdisc overheads (§7.2).

Tsinghua University

UC DAVIS
UNIVERSITY OF CALIFORNIA

10

Experiment: request queueing

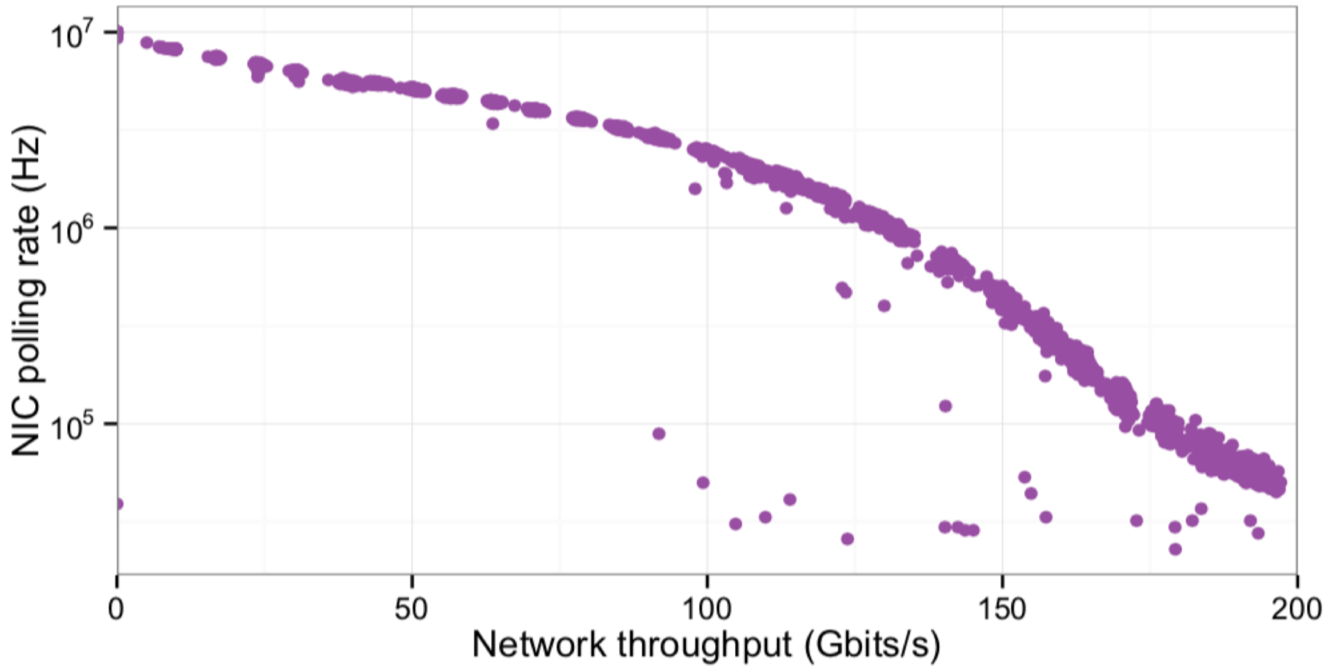


Figure 10: As more requests are handled, the NIC polling rate decreases. The resulting queueing delay can be bounded by distributing request-handling across multiple comm-cores.

清華大學
Tsinghua University

UCDAVIS
UNIVERSITY OF CALIFORNIA

11

Experiment: communication control overhead

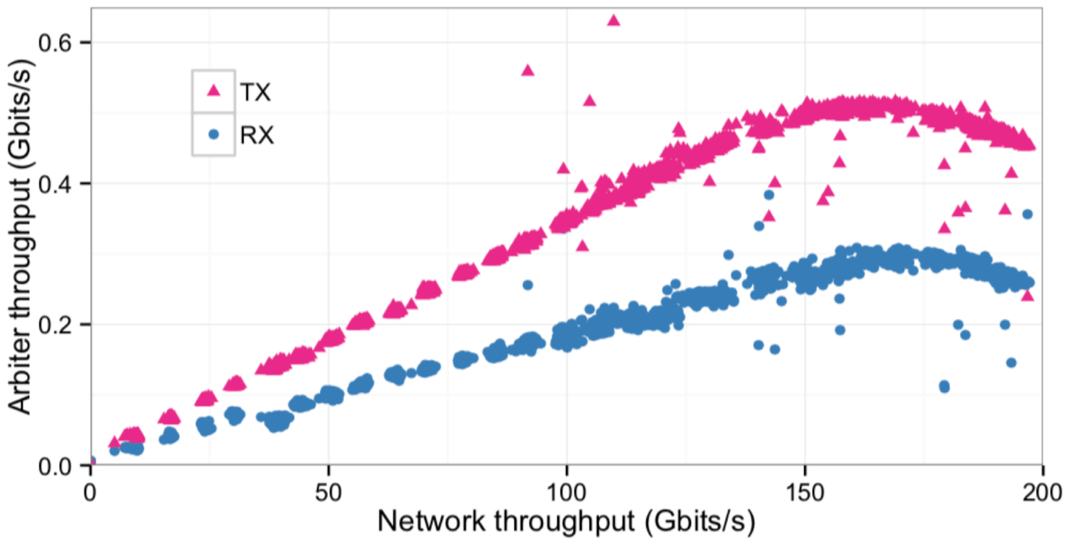


Figure 11: The arbiter requires 0.5 Gbits/s TX and 0.3 Gbits/s RX bandwidth to schedule 150 Gbits/s: around 0.3% of network traffic.

- The network overhead of communication with the arbiter is 1-to-500 for request traffic and 1-to-300 for allocations for the tested workload

Tsinghua University

UCDAVIS UNIVERSITY OF CALIFORNIA

12

Experiment: path selection

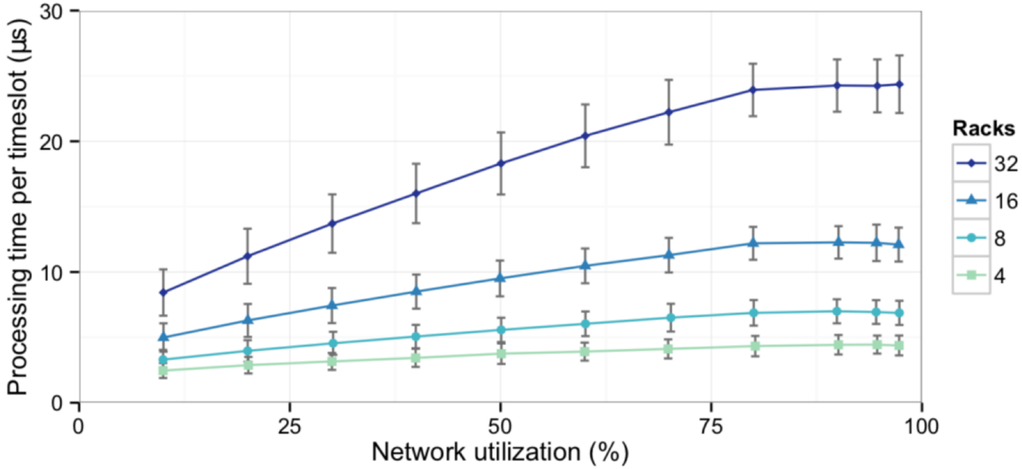


Figure 12: Path selection routes traffic from 16 racks of 32 endpoints in <12 µs. Consequently, 10 pathsel-cores would output a routing every <1.2 µs, fast enough to support 10 Gbits/s endpoint links. Error bars show one standard deviation above and below the mean.

- Fig. 12 shows that the processing time increases with network utilization until many of the nodes reach full degree (32 in the tested topology), at which point the cost of pre-processing[4] the graph decreases, and path selection runs slightly faster.

13

Experiment: Facebook experiment

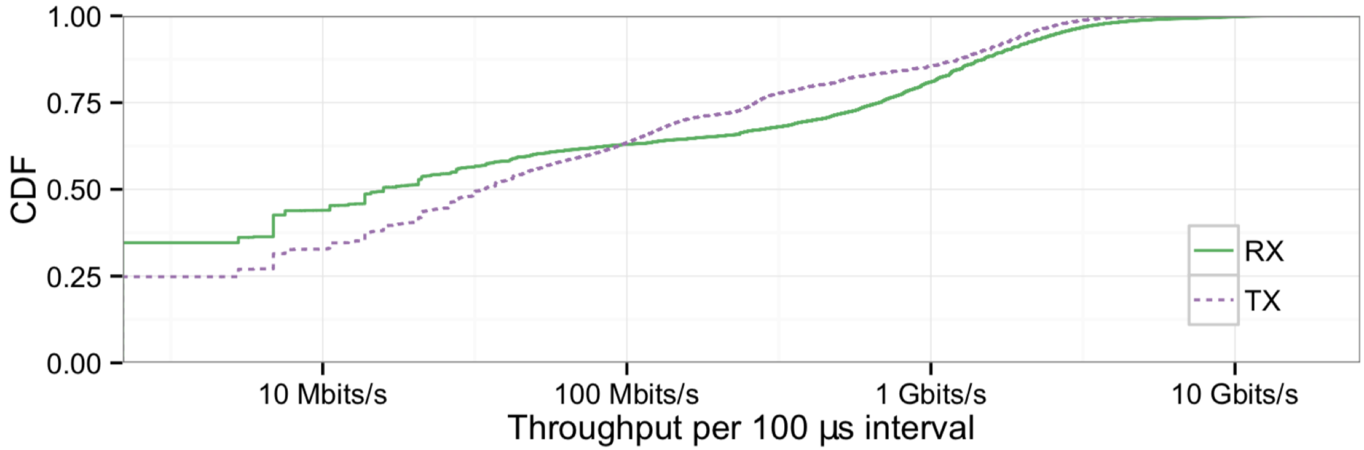


Figure 13: Distribution of the sending and receiving rates of one production server per 100 microsecond interval over a 60 second trace.

- Cluster traffic is bursty, but most of the time utilizes a fraction of network capacity

Experiment: Facebook experiment

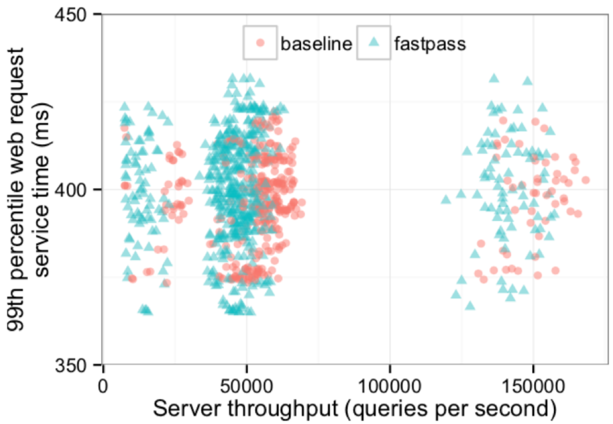


Figure 14: 99th percentile web request service time vs. server load in production traffic. Fastpass shows a similar latency profile as baseline.

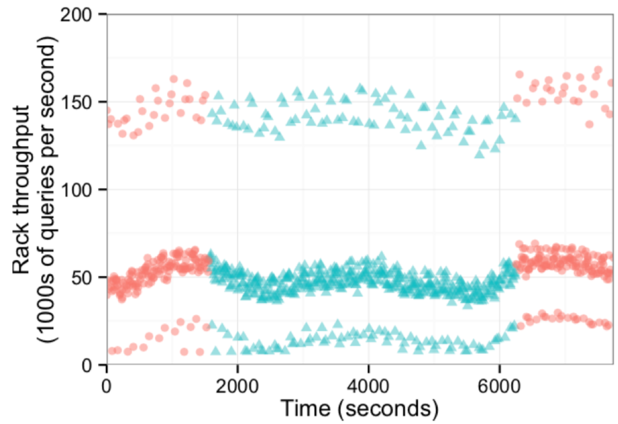


Figure 15: Live traffic server load as a function of time. Fastpass is shown in the middle with baseline before and after. The offered load oscillates gently with time.

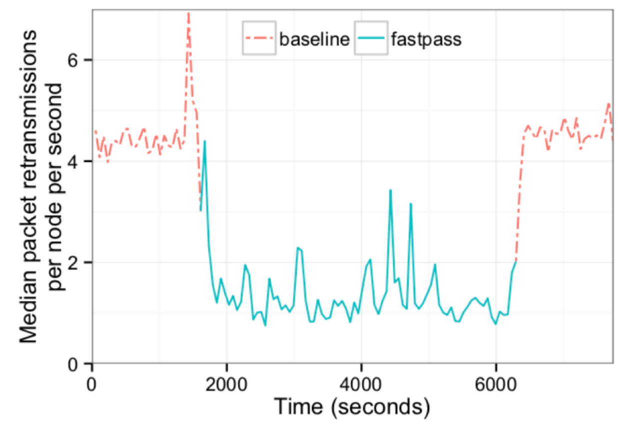


Figure 16: Median server TCP retransmission rate during the live experiment. Fastpass (middle) maintains a 2.5× lower rate of retransmissions than baseline (left and right).

- Fig. 14 shows that the 99th percentile web request service time using Fastpass is very similar to the baseline's. The three clusters pertain to groups of machines that were assigned different load by the load-balancer.
- Fig. 15 shows the cluster's load as the experiment progressed, showing gentle oscillations in load. Fastpass was able to handle the load without triggering the aggressive load-reduction.

Tsinghua University

UC DAVIS
UNIVERSITY OF CALIFORNIA

Some takeaways

- Research approach
  - Theoretical analysis: resource allocation.
  - Experimental demonstration: system performance.
- Scalability can be evaluated by real-world implementation.
- Distributed systems.
- Touch the boundary of optical networking and L3-L4 networking.

16

Thank you for attention!

Zhizhen Zhong

Tsinghua University & UC Davis

zhongzz14@mails.tsinghua.edu.cn , zzzhong@ucdavis.edu

11 May 2018

Networks Lab Group Meeting

清華大學 Tsinghua University

UCDAVIS UNIVERSITY OF CALIFORNIA