Towards Efficient Networked Systems

IHSAN AYYUB QAZI
SBA SCHOOL OF SCIENCE & ENGINEERING
LUMS, LAHORE, PAKISTAN

NetSeminar, Stanford University (Dec 9, 2015)
Cloud Computing & Data Centers

Wireless Networks

Internet Censorship
Cloud Computing & Datacenters

- **Low latency** datacenter networks
  [SIGCOMM’14, INFOCOM’13]

- **Data center** architecture design
  [SIGCOMM’15 (poster), SIGCOMM’14 (poster)]
  - Efficient and resilient **data plane** and **control plane** designs
Wireless Networks

- New challenges as we scale to Gbps WiFi
  [INFOCOM’14, INFOCOM’13 (poster)]
  - User-level throughput degrades dramatically
  - QoS differentiation becomes challenging

- Low cost data channels for developing regions
  [INFOCOM’14]
  - Example: Data over SMS
  - Long distance WiFi
Internet Censorship

- How to build incentive-aligned censorship measurement systems? [ACM HotNets’14]

- Measuring the impact of censorship on various stakeholders (e.g., users, ISPs) [ACM IMC’14]
Networks & Systems Group @ LUMS
Networks & Systems Group @ LUMS

Cloud Computing & Data Centers

Wireless Networks

Internet Censorship
Low Latency Data Center Networks

- **L²DCT**
  - Context: Private DCs
  - Transport Framework
  - (INFOCOM'13)

- **PASE**
  - Context: Private DCs
  - Transport Framework
  - (SIGCOMM'14)

- **MulBuff**
  - Context: Private & Cloud DCs
  - Queue Management Scheme
  - (SIGCOMM'14) Poster
Low Latency Data Center Networks

- **L²DCT**
  - (INFOCOM’13)
  - Context: Private DCs
    - Transport Framework

- **PASE**
  - (SIGCOMM’14)
  - Context: Private DCs
    - Transport Framework

- **MulBuff**
  - (SIGCOMM’14) Poster
  - Context: Private & Cloud DCs
    - Buffer Management Scheme
Friends, not Foes – Synthesizing Existing Datacenter Transport Strategies in PASE

Ihsan Ayyub Qazi (LUMS), Ali Munir (MSU), Ghufran Baig (LUMS), M. Irteza (LUMS), Alex Liu (MSU), Fahad Dogar (MSR)

This work appeared in ACM SIGCOMM 2014
User Facing Online Services

- Online services have become extremely popular
  - e.g., web search, social networking, advertising systems

Key goal: Minimize user response time!
Why response time matters?

- Every 100ms latency costs 1% in business revenue
  [Speed matters, G. Lindan]

- Traffic reduced by 20% due to 500ms increase in latency
  [M. Mayer at Web 2.0]

- An extra 400ms reduced traffic by 9%
  [YSlow 2.0, S. Stefanov]

Impacts user experience & operator revenue
Why is the network important?

- Typical applications have distributed components
  - e.g., a bing search query touches > 100 machines

- Network can impact application performance
  - “10% of search responses observe 1ms to 14 ms of network queuing delays” [Bing, SIGCOMM’10]
DC Network Resource Allocation

- **Fair Sharing**
  Equal bandwidth sharing among flows (e.g., TCP, DCTCP)
  - Sub-optimal completion times
  - Traditional “fairness” metrics less relevant

- **Priority Scheduling (QoS-Aware)**
  Prioritize some flows over other flows based on their requirements
  - Minimize flow completion times (e.g., pFabric, L^2DCT, PIAS)
  - Meet flow deadlines (e.g., D^3, D^2TCP)
Challenges in Data Centers

- Partition/Aggregate Structure of Applications
  - Leads to synchronized responses

- Traffic workloads
  - Short flows — *delay-sensitive*
  - Long flows — *throughput-sensitive*

- TCP does not meet the demands of applications
  - Requires *large queues* for achieving high throughput
DC Transports

- DCTCP (SIGCOMM'10)
- $D^2$TCP (INFOCOM’13)
- $L^2$DCT (SIGCOMM’12)
- $D^3$ (SIGCOMM’11)
- pFabric (SIGCOMM’13)
- PDQ (SIGCOMM’12)
DC Transports

Deployment Friendly but Sub-optimal
- DCTCP
  - SIGCOMM’10
- L3DCT
  - SIGCOMM’12

Near-optimal but not Deployment Friendly (Data plane Changes)
- D³
  - SIGCOMM’11
- pFabric
  - SIGCOMM’13
- PDQ
How can we design a deployment friendly and near optimal DC transport while leveraging the insights offered by existing proposals?
How can we design a deployment friendly and near optimal DC transport while leveraging the insights offered by existing proposals?
Two Design Approaches

Centralized

Challenges
1. Latency to the controller
2. Processing and network overheads

Distributed

Challenges
- In switches (e.g., PDQ, D3)
- At the end-hosts (e.g., DCTCP, L2DCT)

Challenges
- Changes in switches
- Sub-optimal

e.g., Fastpass, SDT
which is important because many datacenter applications launch
providing a solution to several specific problems observed in datacenter
distributed congestion control, even with router support. By contrast,
tives within the same network infrastructure is challenging using
minimize job completion time. Supporting these different objec-
tives, some want high bulk throughput, and some want to
low latency, some want to
only partially
mates and assumptions about request RTTs, and solve the problem
missions
queues, response packets are lost, triggering delay-inducing retrans-
tions. With small router
congested traffic. Fastpass does not incur this delay penalty.

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

The arbiter responds to requests within tens of microseconds

2

pass reduced the occurrence of TCP retransmissions by

2

The arbiter also chooses a path through the

Path selection
each timeslot (§


Path selector, and the client-arbiter communication.

Figure 2: Structure of the arbiter, showing the timeslot allocator,

Figure 2: Structure of the arbiter, showing the timeslot allocator,

Arbitrator

Arbitrator

Arbitrator

Arb
Arb
Arb
Arb
Arb
Arb
Arb
Arb
Arb
Arb

The arbiter must achieve high throughput and low latency for both

Because the arbiter knows about all current and scheduled trans-

rare and can be used as an indication of component failure. End-

provides reliability using timeouts and ACKs of aggregate demands

FCP must balance conflicting
	to isolate faulty links or switches and compute fault-free paths. The

take over within a few milliseconds if the primary arbiter fails.

The resulting RTT reduced from 3.56 ms to 230 µs.

2. FASTPASS ARCHITECTURE

3. Requirements: it must consume only a small fraction of network

5200

8-core allocator handling 2.21 Terabits/s

1.6% lower than baseline TCP, while

2.2

1. Requires no changes in switches

3. Achieves near-optimal performance

4. Multiple apps and user objectives

PASE implements a distributed control plane for resource allocation

1. Requires no changes in switches

2. Achieves near-optimal performance

3. Multiple apps and user objectives
Outline

- DC Transport Strategies
- PASE Design
- Evaluation
Outline

- DC Transport Strategies
- PASE Design
- Evaluation
Transport Strategies

- **Self-Adjusting End-Points** e.g., TCP, DCTCP, D²TCP
  - Senders make independent decisions and adjust rate by themselves

- **Arbitration** e.g., D³, PDQ, Baraat
  - A common entity allocates rate to each flow given other network flows

- **In-network Prioritization** e.g., pFabric
  - Switches independently schedules/drops packets based on priorities
Transport Strategies

- **Self-Adjusting End-Points** e.g., TCP, DCTCP, D²TCP
  - Senders make independent decisions and adjust rate by themselves

- **Arbitration** e.g., D³, PDQ, Baraat
  - A common entity allocates rate to each flow given other network flows

- **In-network Prioritization** e.g., pFabric
  - Switches independently schedules/drops packets based on priorities

Existing DC transports use only one of these strategies
<table>
<thead>
<tr>
<th>Transport Strategy</th>
<th>Pros</th>
<th>Cons</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self-Adjusting Endpoints</td>
<td>Ease of deployment</td>
<td>No strict priority scheduling</td>
<td>DCTCP, L</td>
</tr>
<tr>
<td>Arbitration</td>
<td>Strict priority scheduling</td>
<td>High flow switching overhead</td>
<td>PDQ, D</td>
</tr>
<tr>
<td>In-network Prioritization</td>
<td>Low flow switching overhead</td>
<td>Switch-local decisions</td>
<td>pFabric, PIAS</td>
</tr>
<tr>
<td>In-network Prioritization</td>
<td>Work conserving</td>
<td>Limited # of priority queues</td>
<td></td>
</tr>
</tbody>
</table>
Transport Strategies in Unison

In-network Prioritization Alone

Limited # of queues
More # of flows (priorities)

Flow Multiplexing
Limited performance gains!

Flows
1
2
3
4

High Priority
Low Priority

Any static mapping mechanism degrades performance!
Transport Strategies in Unison

In-network Prioritization + Arbitration

Arbitrator
Dynamic mapping of flows to queues

Idea
As a flows’ turn comes, map it to the highest priority queue!
## Combining Strategies

### Arbitration alone

<table>
<thead>
<tr>
<th>Flows</th>
<th>Arbitrator</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Un-paused (1Gbps)</td>
<td>0.8Gbps</td>
</tr>
<tr>
<td>20</td>
<td>Paused (0)</td>
<td>Paused (0)</td>
</tr>
</tbody>
</table>

**Problem:** Calculating precise rates is hard

### Arbitration + Self-adjusting Endpoints

<table>
<thead>
<tr>
<th>Flows</th>
<th>Arbitrator</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Un-paused (1Gbps)</td>
<td>0.8Gbps</td>
</tr>
<tr>
<td>20</td>
<td>Paused (0)</td>
<td>0.2Gbps</td>
</tr>
</tbody>
</table>

Self-adjustment keeps link utilization high even if arbitration decision was imperfect
Transport Strategies in Unison

- In-network Prioritization + Arbitration
  - In-network prioritization helps in reducing the high flow switching overhead with arbitration-based approaches

PASE leverages these insights in its design!
Outline

- DC Transport Strategies
- PASE Design
- Evaluation
PASE Design Principle

*Each transport strategy should focus on what it is best at doing*

- **Arbitrators**
  - Do inter-flow prioritization at coarse time-scales

- **Endpoints**
  - Probe for any spare link capacity on their own

- **In-network prioritization**
  - Do per-packet prioritization at sub-RTT timescales
Arbitration: Control plane (decoupled from the data plane)
PASE Overview

- **Arbitration**: Control plane (decoupled from the data plane)
  Assign flows’ a “reference rate” and “priority queue”
**PASE Overview**

- **Arbitration**: Control plane (decoupled from the data plane)
  Assigns each flow a “reference rate” and “priority queue”

- **Self-Adjusting Endpoints**: Guided rate control
PASE Overview

- **Arbitration**: Control plane
  Assign each flow a “reference rate” and “priority queue”
- **Self-Adjusting Endpoints**: Guided rate control
- **In-network Prioritization**: Uses existing queues
➢ Distributed Arbitration per link arbitration done in control plane

➢ Arbitration can be implemented at the end hosts (e.g., for their own links to the switch) or on dedicated hosts inside the DC

Sends data with min rate & priority
PASE Arbitration – Challenges

Challenges

- Arbitration latency
- Processing overhead
- Network overhead
PASE Arbitration – Challenges

**Challenges**

- Arbitration latency
- Processing overhead
- Network overhead

**Solution:** Leverage the tree-like structure of typical DC topologies
Bottom Up Arbitration

➢ **Leverage Tree Structure**
from leaves up to the root

Intra-Rack

**No external arbitrators required!**

Inter-Rack

Facilitates optimizations \(\text{(early pruning \& delegation)}\) for inter-rack scenarios!
Key Idea: **Divide a link into virtual links and delegate responsibility to child arbitrators**

**Algorithm**
- Link capacity $C$ is split in $N$ virtual links
- Parent arbitrator delegates virtual link to child arbitrator
- Child arbitrator does arbitration for virtual link
- Virtual link capacity is periodically updated based on the top $k$ flows of all child arbitrators
Delegation

**Key Idea:** Divide a link into virtual links and delegate responsibility to child arbitrators

**Algorithm**
- Link capacity $C$ is split in $N$ virtual links
- Child arbitrator does arbitration for virtual link
- Virtual link capacity is periodically updated based on the top $k$ flows of all child arbitrators

**Reduces Arbitration Latency**
Make arbitration decisions close to the sources!
PASE Endhost Transport

➢ **Rate Control**
  ➢ Uses the *reference rate* (from the arbitrators) as the pivot rate
  ➢ Follows DCTCP control law

➢ **Loss Recovery Mechanism**
  ➢ Packets in lower priority queues can be delayed for several RTTs
    ➢ *large RTO or small probes* for reducing spurious retransmissions
PASE - Putting Together

- Efficient arbitration control plane
- Simple TCP-like transport
- Existing priority queues inside switches
Outline

- DC Transport Strategies
- PASE Design
- Evaluation
Evaluation

➢ Platforms
  ▶ Small-scale testbed
  ▶ ns2 simulations

➢ Workloads
  ▶ Web search (DCTCP), Data mining (VL2)

➢ Comparison with deployment friendly
  ▶ DCTCP, D^2TCP, L^2DCT

➢ Comparison with best performing
  ▶ pFabric
Basic Simulation Setup

<table>
<thead>
<tr>
<th>Queue Size</th>
<th>250KB (per queue)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTT</td>
<td>300usec</td>
</tr>
<tr>
<td>RTO</td>
<td>1 msec</td>
</tr>
<tr>
<td>L</td>
<td>40</td>
</tr>
</tbody>
</table>
Comparison with Deployment Friendly

Settings similar to $D^2$TCP
- Flow Sizes: 100-500KB
- Deadlines: 5-25msec

PASE is deployment friendly yet BETTER than existing protocols!
Comparison with Best Performing

PASE performs comparable and does not require changes to data plane
Summary

➢ Key Strategies for existing DC Transport
  ▶ Arbitration, in-network Prioritization, Self-Adjusting End-points
  ▶ Complimentary rather than substitutes

➢ PASE
  ▶ Combines the three strategies
  ▶ Efficient arbitration control plane; simple TCP-like transport; leverages existing priority queues

➢ Performance
  ▶ Comparable to or better than earlier proposals that require changes to the network fabric
Networks & Systems Group @ LUMS
A Case for Marrying Censorship Measurements with Circumvention

This work appeared in ACM HotNets 2015

Aqib Nisar    Aqsa Kashaf                       Zartash A. Uzmi
Ihsan Ayyub Qazi

SBA School of Science and Engineering, LUMS, Lahore, Pakistan
Uninterrupted Access:

- Facebook YES/NO?
- Google YES/NO?
- Twitter YES/NO?
- Wikipedia YES/NO?
- YouTube YES/NO?

Many would say NO to at least one!
(Censorship in 70+ countries)
Censorship Research Landscape

- **Measuring** censorship
  - What is blocked?
  - Where is it blocked?
  - How it is blocked?

- **Circumventing** censorship
  - How to **bypass** blocking?
Censorship Research Challenges

- **Censorship Measurements**
  - Need widespread measurement points
  - Especially within censored regions

- **Circumvention**
  - Not adaptive to the blocking mechanism
  - One-size-fits-all solutions
What if we combine?

C-Saw
Circumvention + Measurements

Circumvention System
Measurement System

Consumers
Free-speech NGOs
Human Rights Orgs
Civil Societies
Political Activists
Researchers
The C-Saw Feedback Cycle

1. Bootstrap
2. Circumvent
3. Incentivize
4. Crowdsourcing & Measure
5. Analyze & Learn
6. Repeat
Adaptive Circumvention: Example

C-Saw:
just chooses the BEST method
(Data-driven & adaptive)
How much benefit can a system like C-Saw bring?
Case Study: Pakistan

- ISP-A and ISP-B: only international providers

![Diagram showing HTTP and HTTPS traffic]

HTTP Traffic: 
- ISP A: Blocking
- ISP B: Blocking

HTTPS Traffic: 
- ISP A: Accessible
- ISP B: Accessible (HTTPS with Domain Fronting)
Case Study: Pakistan

- ISP-A and ISP-B: only international providers

- Measurements taken at:
  1) University campus (Lahore)
     Served by ISP-A and ISP-B
  2) Home users (Karachi)
     Served by ISP-B only
Insights from Case Study: Pakistan

- Existing **circumvention methods impose widely different overheads**
  - Choose the best

- Censorship **mechanism varies with content type**
  - Even within the same ISP
  - Adapt circumvention

- Blocking **mechanisms may differ by ISPs**
  - Even for the same site; and
  - For ISPs in the same region
  - Explore cross-ISP links
C-Saw: Preliminary Design

Proxy [local] modules
- Learning
- Distribution
- Circumvention

Learning Module
Measure NOW, use LATER... for better browsing experience [local]

Distribution Module
Measurements made now will help future browsing experience [others]
Proposed Design Features

► Using history information
  ▶ Measurements at user U1 benefit browsing at U1
  ▶ Local database of “previous” measurements

► Community help
  ▶ Measurements at U1 benefit U2
  ▶ Distributed database

► Informed user consent
  ▶ Only those URLs “tested” that user visits

► Exploiting cross-ISP links (explicit consent)
Results from the Case Study
C-Saw vs. Static proxies

Fetched:
YouTube (0.36MB)
200 runs

ISP-B:
Blocking: HTTP & HTTPS
C-Saw used HTTPS/DF

Measurement point:
Campus network

All static proxies exhibited longer page load times
C-Saw vs. Tor

Fetched:
YouTube (0.36MB)
200 runs

ISP-A:
HTTPS did unblock
C-Saw used HTTPS

Measurement point:
Campus network
Tor exit point shown

All Tor results indicate longer page load times
Using cross-ISP links in C-Saw

Fetched:
YouTube (0.36MB)
200 runs

Home user: direct
on ISP-B in Karachi
(used HTTPS/DF)

For Cross-ISP (P2P):
used ISP-A in Lahore
(used HTTPS)

Cross-ISP links may reduce the page load times
Summary

- **Censorship Measurements**
  - Challenge: widespread vantage points
  - C-Saw provides rich measurements

- **Circumvention Performance**
  - Improved by rich monitoring data
    - Judicious selection of circumvention
  - C-Saw incentivizes users to opt-in
Discussion points

• Centralized censorship
  – What can be done there? Cross-ISP links (inter-national)?

• Malicious users (*posing* as victims/altruists)
  – False measurements (Can be eliminated by majority vote!)

• If C-Saw decides to use a relay, how to select that?
  – Load balancing across relays?

• Deployment risks
  – End-users (How to anonymize? Where to store measurement data?)
  – Why wouldn't an adversary block our IPs / take measures against C-Saw?
Few slides on NSG...
Considerable focus on undergraduate research

- **A Case for Marrying Censorship Measurements with Circumvention.** Aqib Nisar, Aqsa Kashaf, Zartash Uzmi, Ihsan Ayyub Qazi. *ACM HotNets 2015* (Acceptance Rate = 18%)


- **Friends, not Foes - Synthesizing Existing Data Center Transport Strategies in PASE.** Ali Munir, Ghufran Baig, Syed Irteza, Ihsan Ayyub Qazi, Fahad Dogar, Alex Liu. *ACM SIGCOMM 2014* (Acceptance Rate = 18.9%)

- **Rethinking Buffer Management in Data Center Networks.** Aisha Mushtaq, Asad Khalid Ismail, Abdul Wasay, Bilal Mahmood, Ihsan Ayyub Qazi, Zartash Afzal Uzmi. *ACM SIGCOMM 2014* (poster)


- **On the Effectiveness of High-Speed WLAN Standards for Long Distance Communication.** Kamran Nishat, Ruwaifa Anwar, Ahmed Mehfooz, Bilal Zaidi, Haris Choudhary, Ihsan Ayyub Qazi. *IEEE INFOCOM 2014* (poster)

- **Minimizing Flow Completion Times in Data Centers.** Ali Munir, Ihsan Ayyub Qazi, Zartash Afzal Uzmi, Aisha Mushtaq, Saad Ismail, M. Safdar Iqbal, and Basma Khan. *IEEE INFOCOM 2013* (Acceptance Rate = 17%)
Placements/Admissions/Awards @ NSG

➡️ Ph.D. Admissions
  ➡️ UC Berkeley, Stanford, CMU, UIUC, UCLA, Georgia Tech, USC, UMass Amherst, UT Austin, Harvard, Wisconsin Madison, Rice University, Stony Brook, Boston University, Michigan, Max Planck, EPFL, Waterloo, etc

➡️ Masters Admissions
  ➡️ CMU, UCSD, UT Austin, Cornell, Columbia, NYU, USC, Brown University, University of Virginia, Virginia Tech, ETH Zurich, EPFL, UBC, University of Toronto, Dartmouth, etc

➡️ Industry
  ➡️ Microsoft, Google, Amazon
  ➡️ Internship(s): EPFL (Switzerland)

➡️ Awards
  ➡️ Facebook Grace Hopper Scholarship, 2014
Thank you!
ihsan.qazi@lums.edu.pk
nsg.lums.edu.pk