Cutting Tail Latency in Commodity Datacenters with CloudBurst

Gaoxiong Zeng[1], Li Chen[1,2], Bairen Yi[1,3], Kai Chen[1]

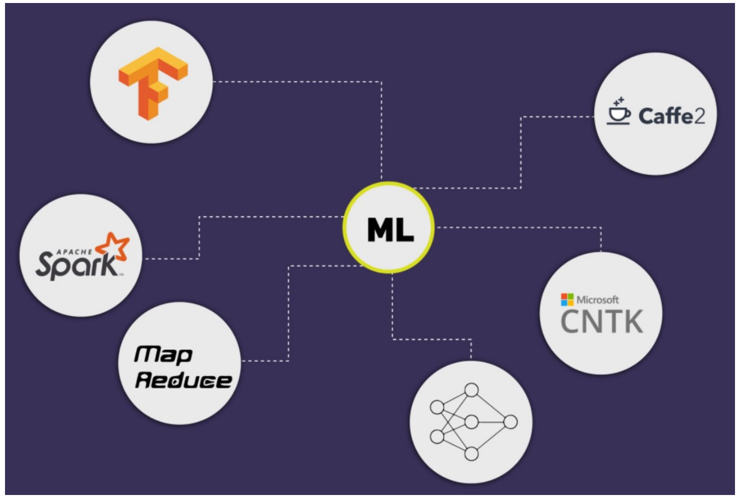
[1]iSINGLab @ Hong Kong University of Science & Technology

[2]Huawei, [3]ByteDance

DCN applications are latency-sensitive

- Datacenter applications & services require low network latency.

Big Data / Machine Learning System



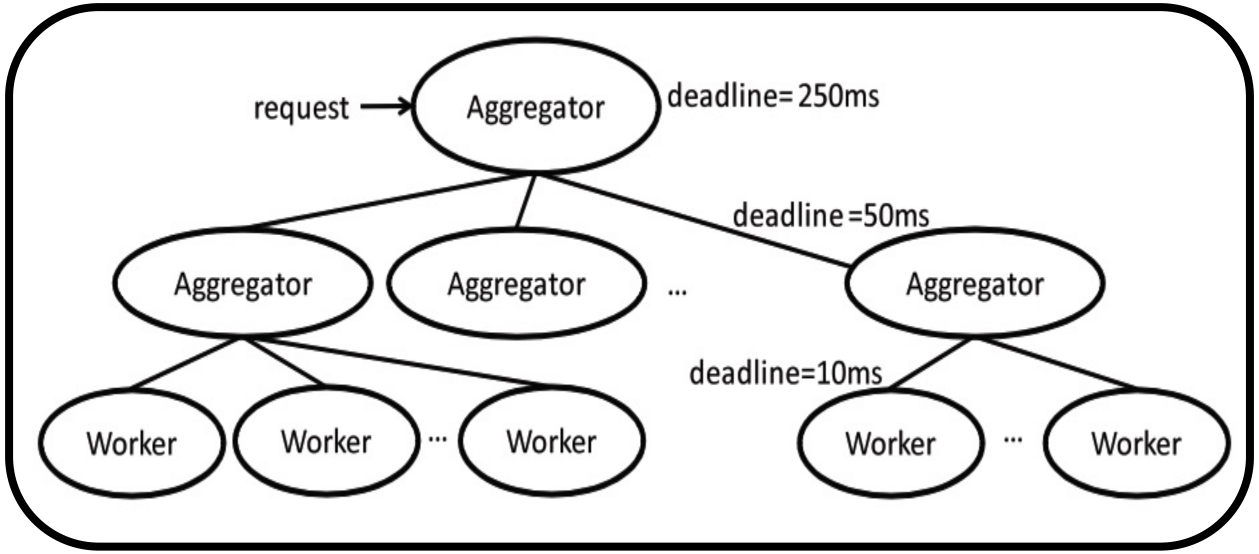
Faster computing imposes harsh requirement on network latency: **CPU -> GPU -> TPU**

Distributed Storage / Database System

Faster storage media imposes harsh requirement on network latency: **HDD -> SSD -> NVMe**

Long tail latency – Why does it happen?

1. High fan-in bursts (Incast)
   - Model synchronization (especially barrier-synchronized) for DNN training;
   - Application partition/aggregate pattern [*]…



[*] Data Center TCP (DCTCP). SIGCOMM 2010.

Long tail latency – Why does it happen?

2. Shallow shared-buffer switch
   ✔ Buffers are often shared to absorb bursts. However,
   ✗ Trend: Buffer per port per Gbps is decreasing as link speed grows [*].

ASIC	Broadcom 56538	Broadcom Trident+	Broadcom Trident II	Broadcom Tomahawk	Barefoot Tofino
Capacity (ports × BW)	48 p × 1 Gbps	48 p × 10 Gbps	32 p × 40 Gbps	32 p × 100 Gbps	64 p × 100 Gbps
Total buffer	4MB	9MB	12MB	16MB (4 MMUs)	22MB
Buffer per port	85KB	192KB	384KB	512KB	344KB
Buffer per port per Gbps	85KB	19.2KB	9.6KB	5.12KB	3.44KB

   ✗ Besides, guaranteed buffer per port (>α*BDP) is required for high throughput, thus leaving limited shared buffer for burst tolerance.

[*] One More Config is Enough: Saving (DC)TCP for High-speed Extremely Shallow-buffered Datacenters. INFOCOM 2020.

Long tail latency – Why does it happen?

3. Error handling and retransmission timeout

   Fast recovery (e.g., with duplicate ACK) requires at least one RTT.

   ✗ Timeout-based recovery is inevitable for tail dropping and small flows.

   ✗ RTO_{min} in DCN can be as low as 5ms (200ms for the Internet), still several orders of magnitud

   How to address the long tail latency?

4. Malfunctioning hardware
5. Imperfect traffic load balancing
6. …

Long tail latency – How to address it?

1. Reducing network queueing

- Fine-grained load balancing
  - ➢ CONGA, FastPass, etc.
  - Complex network control or switch modification.
- Rate control
  - ➢ DCTCP, Timely, etc.
  - Still need a good load balancing scheme to work well.
- Traffic prioritization
  - ➢ pFabric, Qjump, etc.
  - Limited priority queues in practice and rely on accurate configuration.

Long tail latency – How to address it?

2. Recovering from packet losses

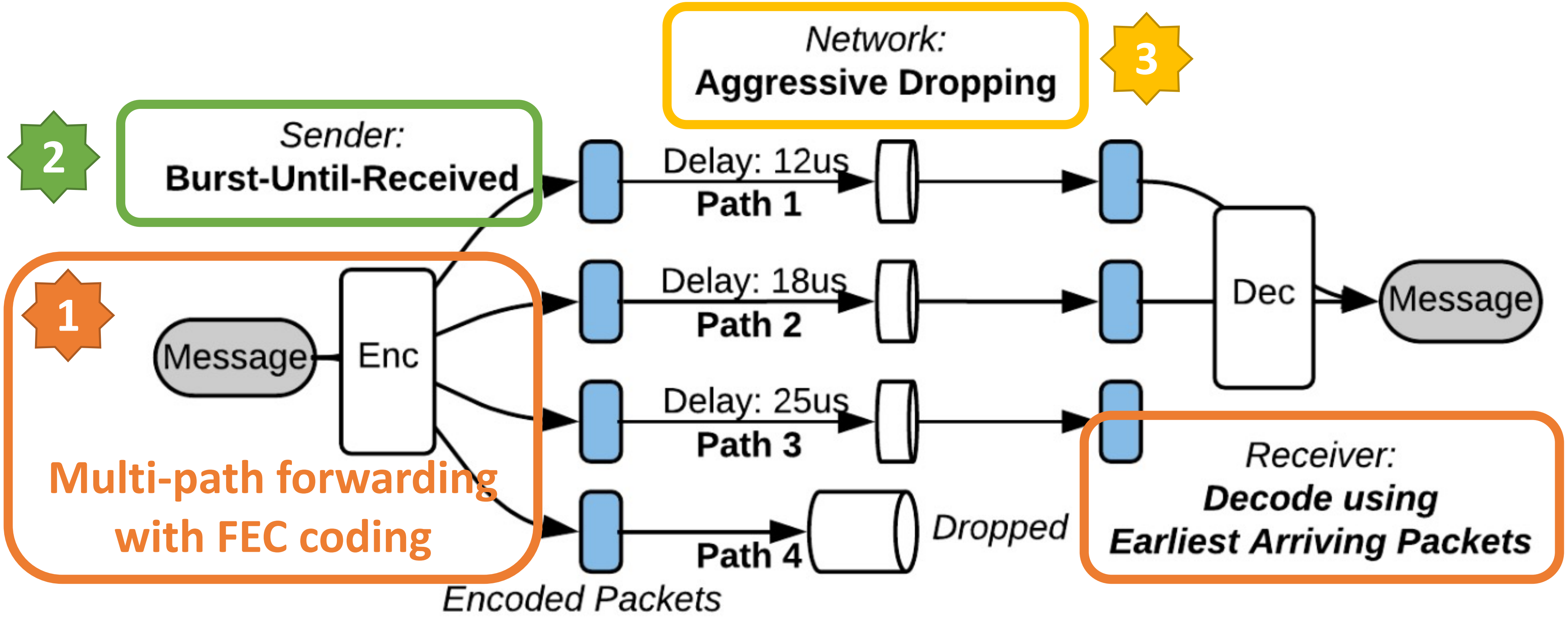
- Fast in-network feedback: non-trivial hardware modification.
- Lossles k, etc.).

Can we address the long tail latency problem with a *simple, readily deployable yet effective* solution?

3. Proactive transport solutions

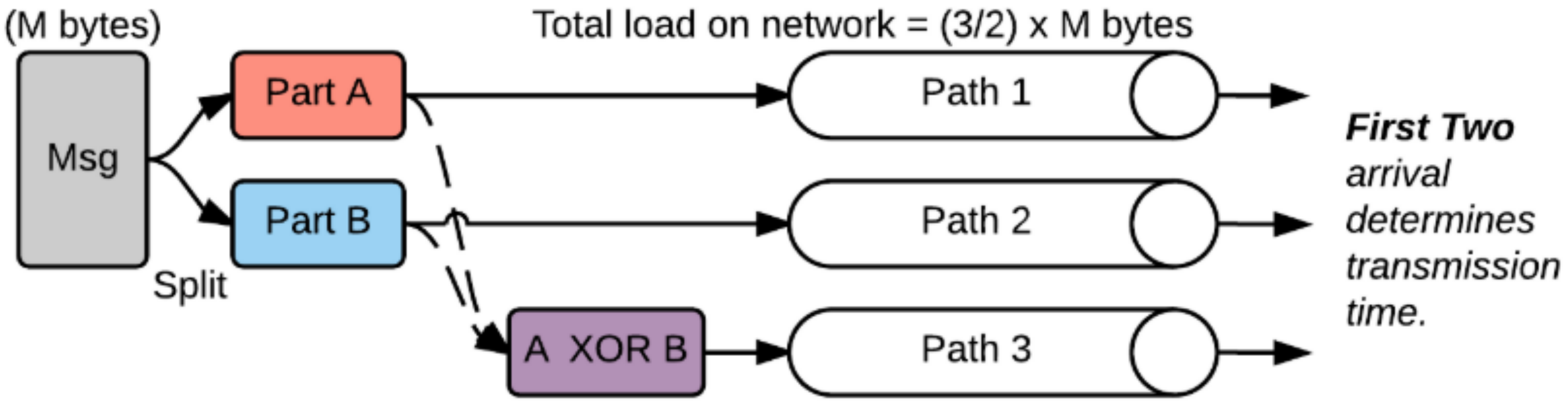
- NDP, ExpressPass, Homa, etc.
- Existing problems unsolved (e.g., first-RTT delay) for real deployment.

CloudBurst: a simple yet effective solution



CloudBurst: a simple yet effective solution

1. Multi-path forwarding with forward error correction (FEC)



- An example of 3-path FEC:
  - Original message can be reconstructed with the earliest arrived 2 of 3 parts.

Fixed-rate block codes:
In this case, code rate = 3/2.

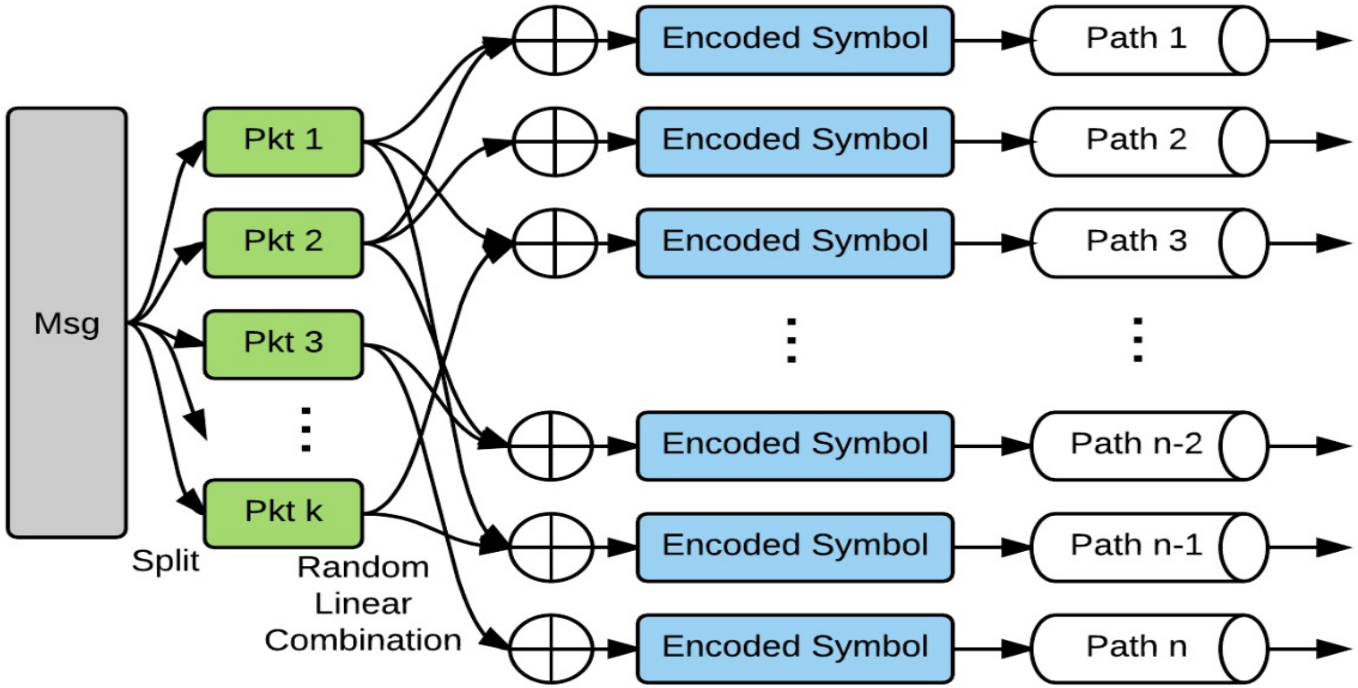
CloudBurst: a simple yet effective solution

1. Multi-path forwarding with forward error correction (FEC)

- **LT Code (LTC)** is recommended & prototyped in our work.

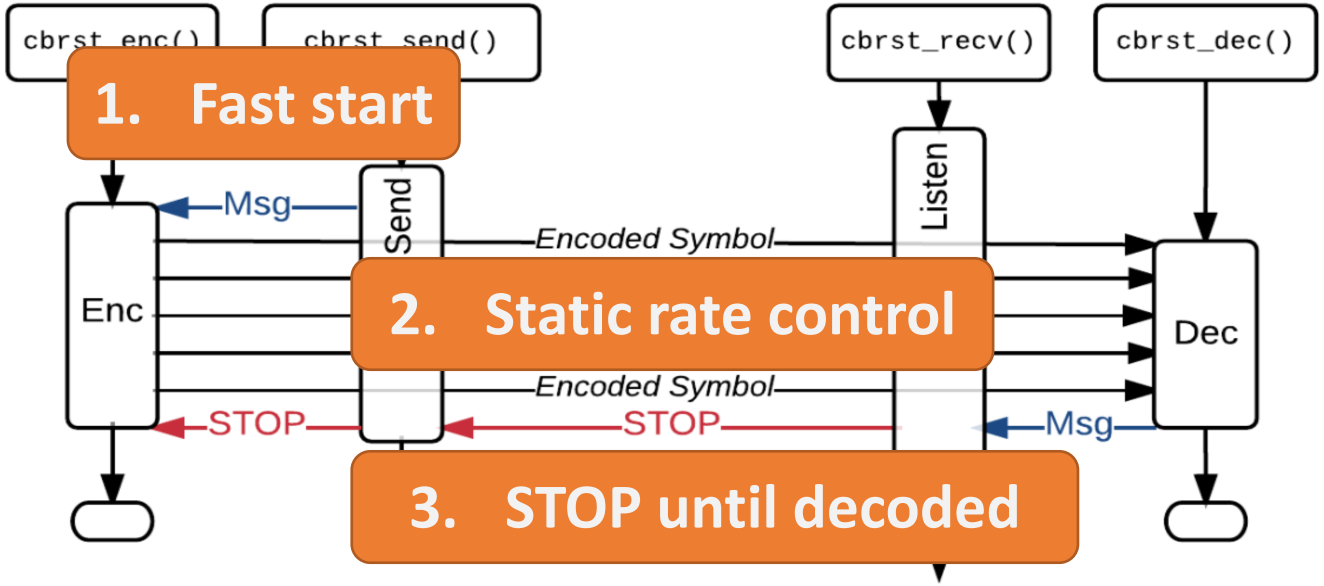
✓ **Adaptable code rate**

✓ **Low complexity**



CloudBurst: a simple yet effective solution

2. Burst-until-received at the end-host

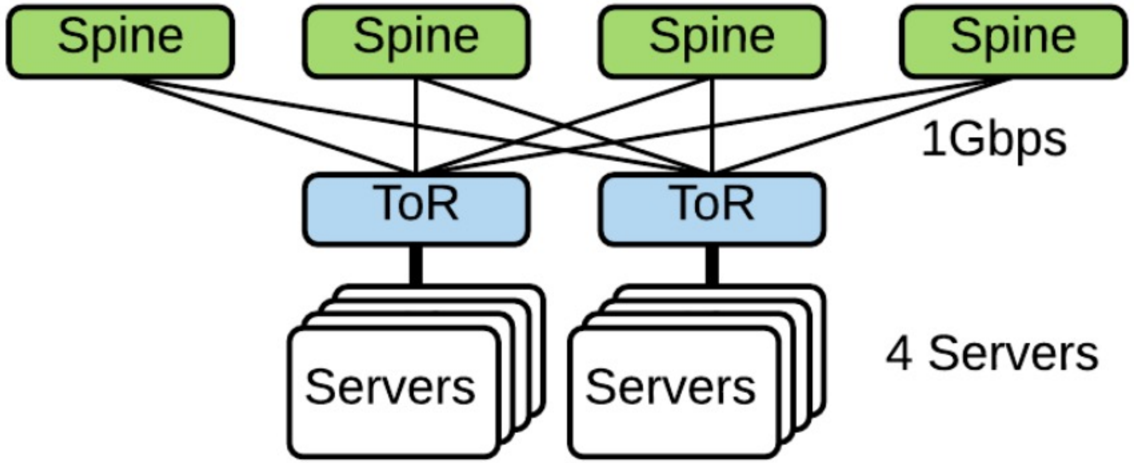


3. Aggressive dropping in the network

- Separate CloudBurst flows on a small-buffered queue.

Implementation

- CloudBurst is prototyped as a user-space library with Rust 1.6.
  - Multi-path routing is achieved with XPath [*].
- A testbed is built with commodity switches & servers as shown below.



[*] Explicit Path Control in Commodity Data Centers: Design and Applications. NSDI 2015.

Evaluation

- How effective is the design choices of CloudBurst?
  - We compare DCTCP with 4 variants of CloudBurst:

  A: FEC; B: FEC + multipath; C: FEC + aggressive dropping; D: Full CloudBurst.

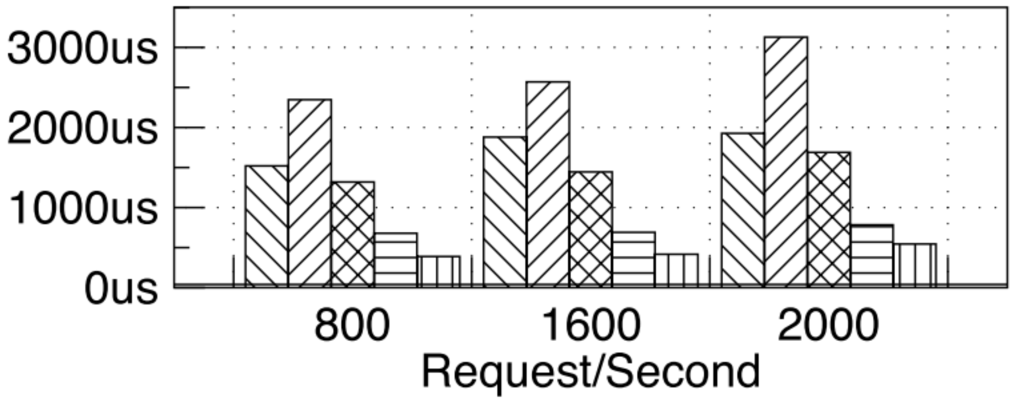


Fig. 9. p99 Completion Time (5KB)

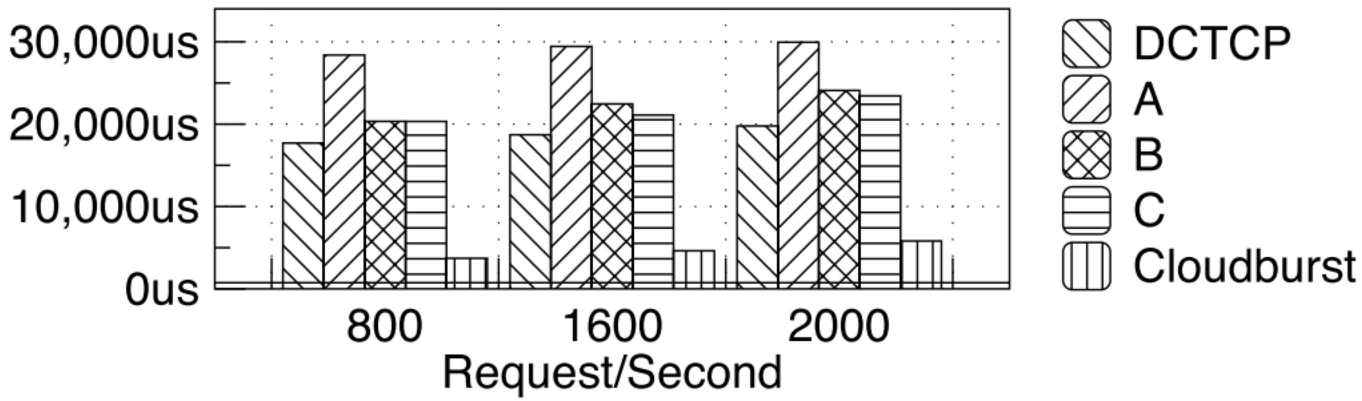


Fig. 11. p99 Completion Time (93KB)

Evaluation

- How does CloudBurst perform compared with prior schemes?
    - We compare PIAS, DCTCP (with replication on multipath), MPTCP with CloudBurst.

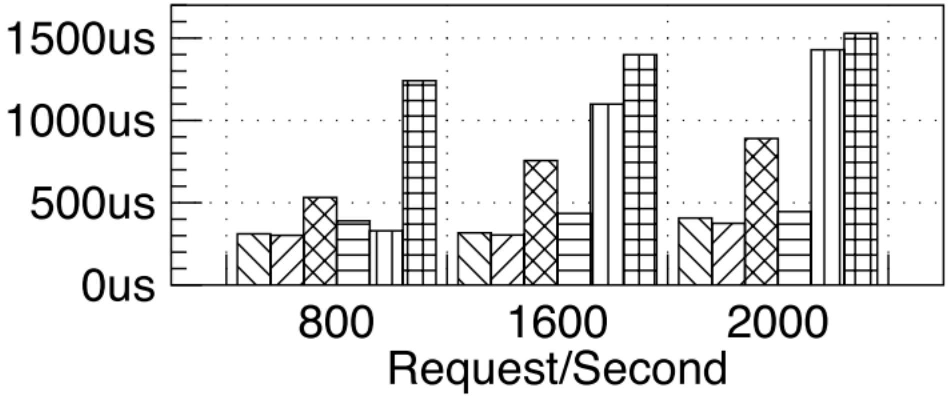


Fig. 12. Average Message Completion Time

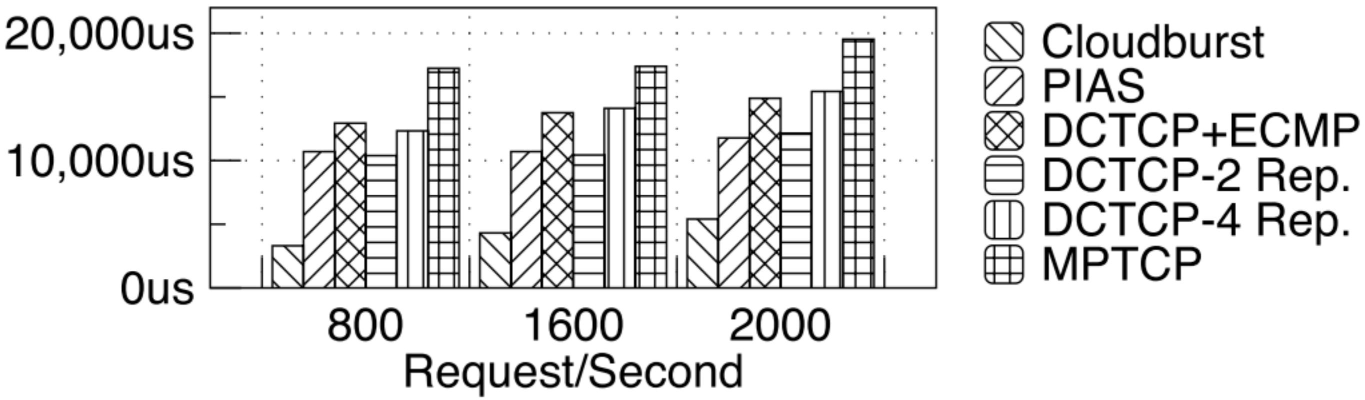


Fig. 13. p99 Message Completion Time

Evaluation

- How does CloudBurst perform under large-scale networks?
  - We conduct large-scale DCN experiments with ns-2 simulation.
  - CloudBurst outperforms other schemes without hardware modification.

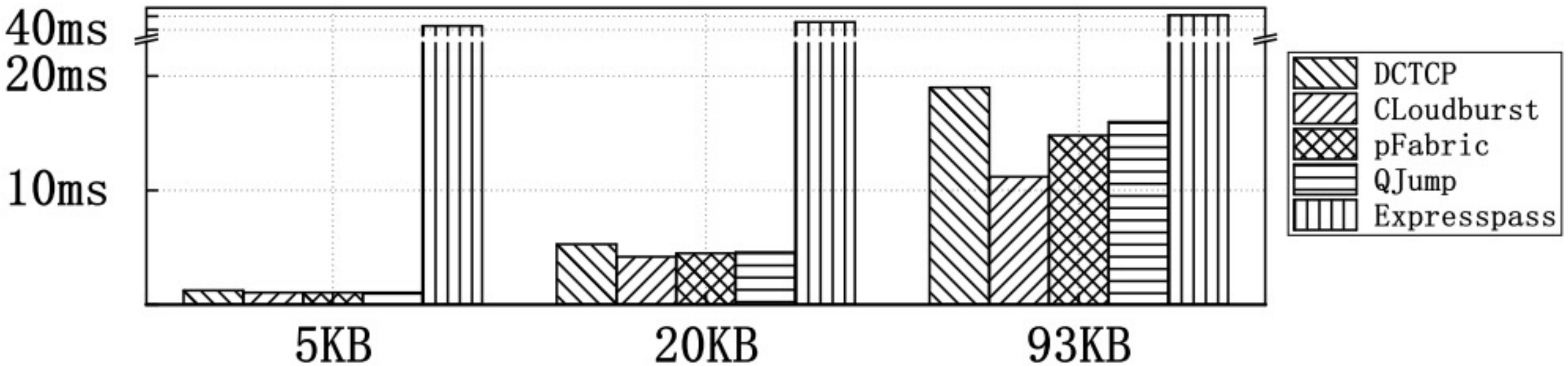


Fig. 17. p99 Completion Time

Conclusion

- A comprehensive study of the long tail latency problem:
  - Why does it happen?
  - How to address it?
- Our design: CloudBurst
  - Key idea: Multi-path forwarding with forward error correction.
- Implementation & Evaluation:
  - 63.69% and 60.06% reduction in 99th tail FCT compared to DCTCP & PIAS.

Thanks!