To switch or not to switch to TCP Prague?
Incentives for adoption in a partial L4S deployment

Fatih Berkay Sarpkaya, Ashutosh Srivastava, Fraida Fund, Shivendra Panwar

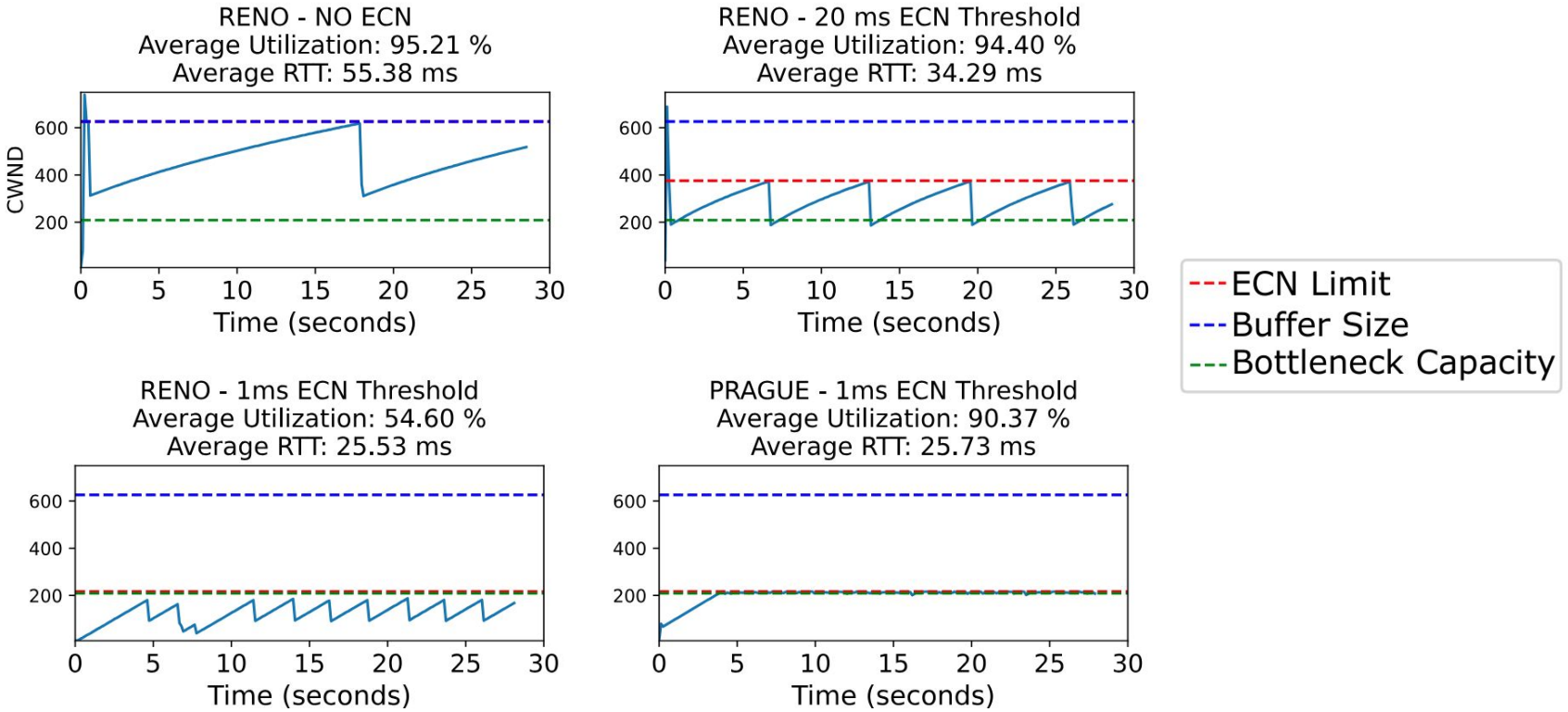
NYU TANDON SCHOOL OF ENGINEERING

L4S is an architecture for -

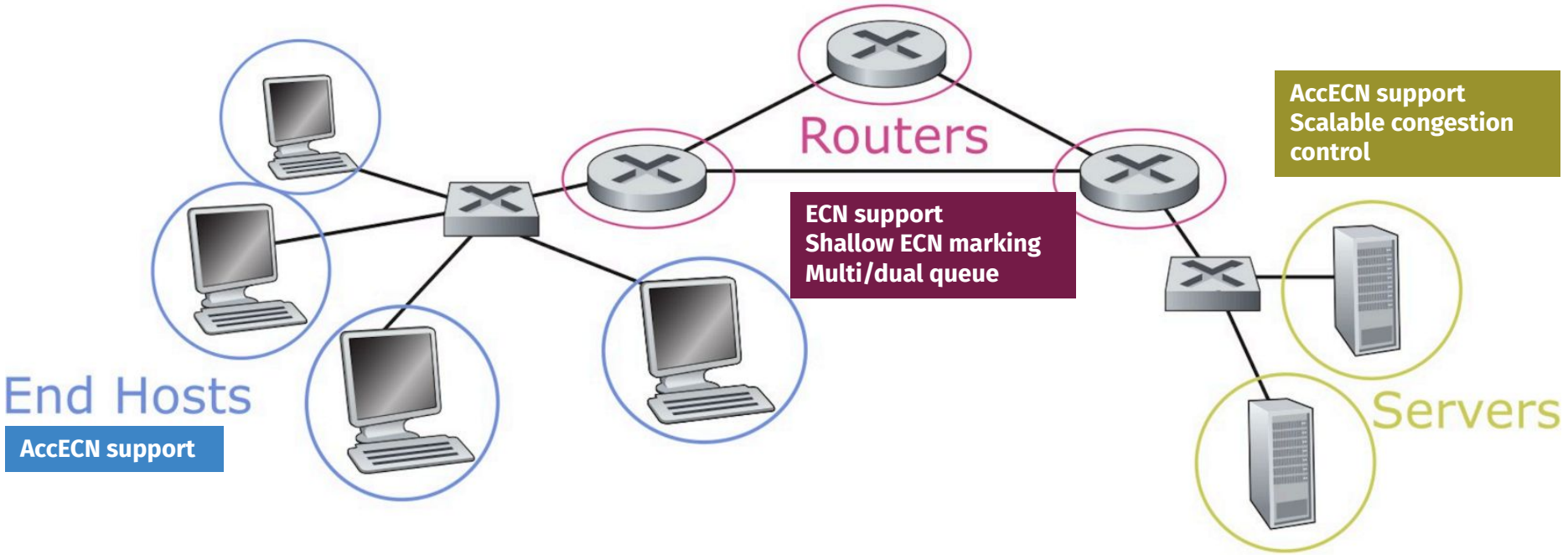
- **L**ow **L**atency
- **L**ow **L**oss
- **S**calable Throughput

1 B. Briscoe, K. De Schepper, M. Bagnulo, and G. White, “RFC 9330: Low latency, low loss, and scalable throughput (L4S) internet service: Architecture,” USA, 2023.

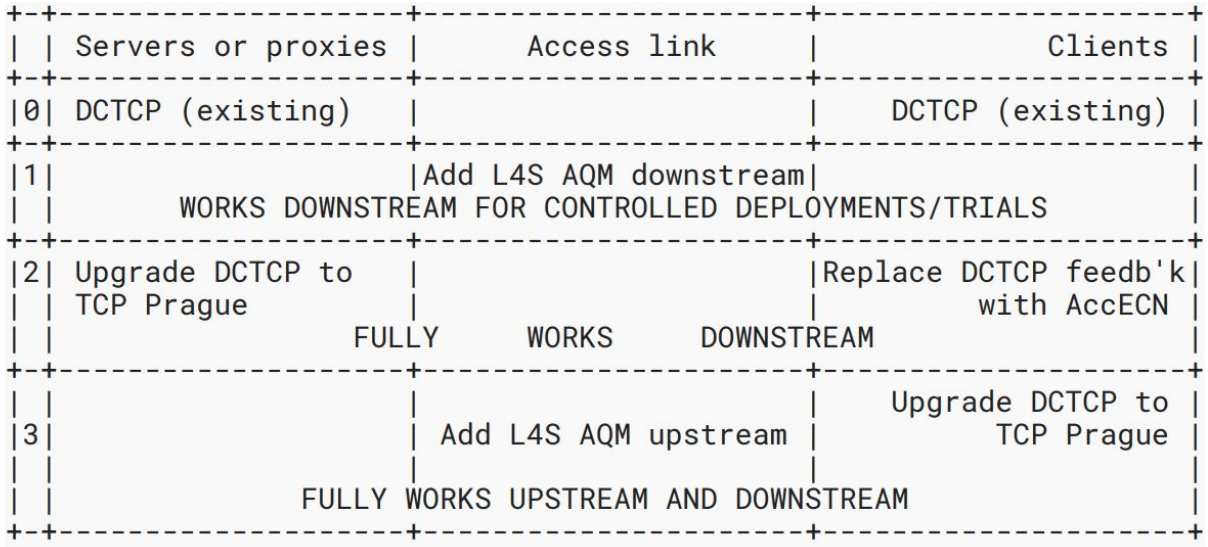
Scalable congestion control is key part of L4S architecture



But, L4S involves several more elements -



In best-case incremental deployment scenario, L4S flows remain isolated from classic flows…



Example L4S Deployment Sequence (RFC 9330)

1 B. Briscoe, K. De Schepper, M. Bagnulo, and G. White, “RFC 9330: Low latency, low loss, and scalable throughput (L4S) internet service: Architecture,” USA, 2023.

2 White, G. (2024, March 17). Operational guidance on coexistence with classic ECN during L4S deployment (draft-ietf-tsvwg-l4sops-06). Internet Engineering Task Force. https://datatracker.ietf.org/doc/draft-ietf-tsvwg-l4sops/06/

… but in other sequences, scalable flows can coexist with classic flows at L4S or non-L4S bottleneck routers.

However:

- There are likely to be scenarios in which L4S flows will traverse non-L4S AQMs. [peering points, non-cable access links (4G/5G/Wi-Fi/Satellite)]
- Upgrading legacy access network middleboxes, e.g., Wi-Fi routers or 3G/4G base stations worldwide, to support L4S will also be a challenge.

Performance during incremental deployment is essential for transition to widespread deployment

To encourage more widespread deployment,

➔ an L4S flow should have throughput and delay characteristics at least as favorable as a classic flow, even if some elements of the full architecture are missing.

➔ Also, an L4S flow should not be harmful to classic flows.

Early tests raise some concerns -

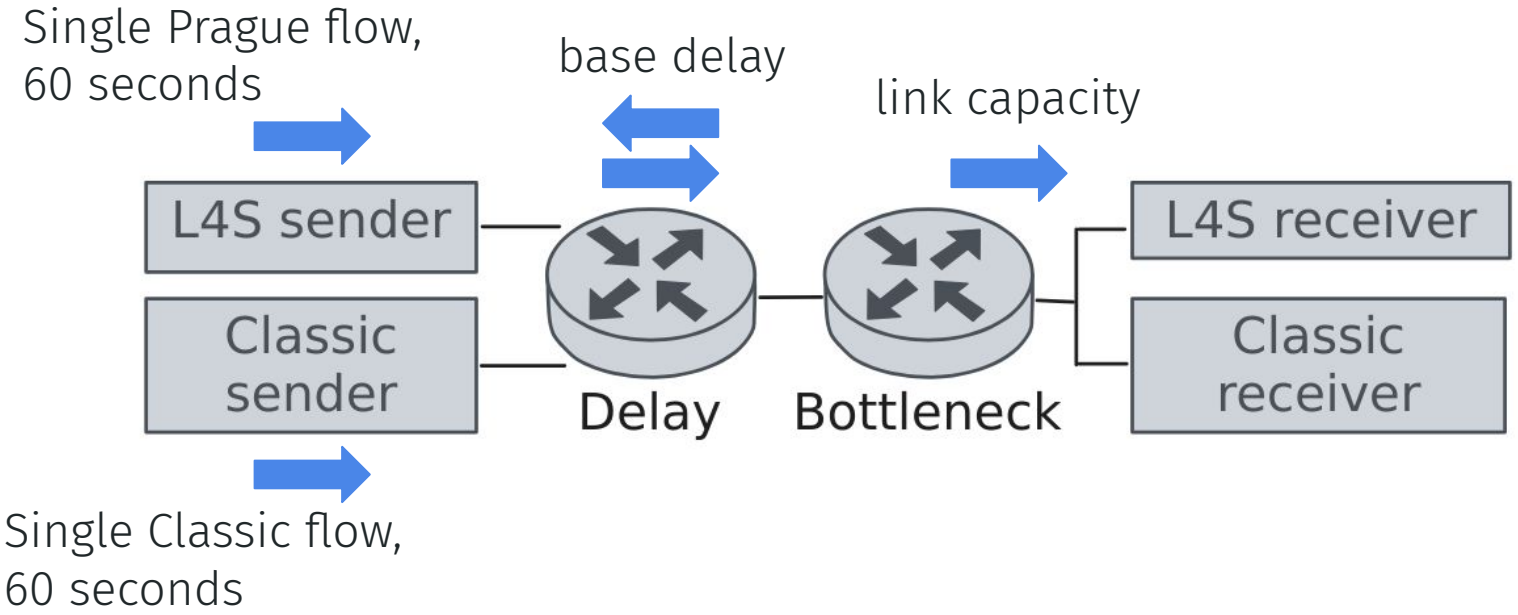
- “L4S flows dominate non-L4S flows, whether ECN enabled or not, when they occupy a shared RFC3168 signaling queue.” (Issue #16)
  - P. Heist, L4S Tests, https://github.com/heistp/l4s-tests, 2021
  - T. Henderson, O. Tilmans, and G. White, “Testbed and Simulation Results for TSVWG Scenarios,” 2019, https://l4s.cablelabs.com/l4s_issues.html
  - TSVWG IETF-106 Interim Feb 2020
- “The DualPI2 qdisc introduces a network bias for TCP Prague flows over existing CUBIC flows.”
  - P. Heist, L4S Tests, https://github.com/heistp/l4s-tests, 2021
- “TCP Prague behaves approximately like NewReno and, is outperformed by CUBIC in pFIFO bottleneck. It is difficult to see where L4S 'scalable throughput' claim is justified here.”
  - P. Heist, L4S Tests, https://github.com/heistp/l4s-tests, 2021

We try to evaluate TCP Prague (a scalable CC) with this question in mind -

Given that the bottleneck router may or may not have a dual queue AQM, and given that the other flows sharing the same bottleneck may not be TCP Prague flows, what benefit can a sender expect from unilaterally switching its own congestion control to TCP Prague?

Experiment design on FABRIC testbed: topology

- A flow is most likely to encounter a bottleneck either at a peering point, or at the access link. We emulate network conditions that are representative of an access link: **10 ms base RTT**, **100 Mbps bottleneck link capacity**.



FABRIC

Experiment design on FABRIC testbed: queues

	ECN	AQM	Multi/Dual Queue	Shallow ECN marking
FIFO : single drop tail queue				
FIFO (+ECN) : single drop tail queue with ECN	✅			
CoDel (ECN + AQM) : single queue with CoDel AQM	✅	✅		
FQ (ECN + Multi-Queue) : fair queue with flow isolation and ECN	✅		✅ (Multi)	
FQ-CoDel (ECN + Multi-Queue + AQM) : fair queuing with the CoDel AQM	✅	✅	✅ (Multi)	
DualPI2 (ECN + Dual-Queue + AQM)	✅	✅	✅ (Dual)	✅

Experiment design on FABRIC testbed: congestion controls

	Loss-Based	Rate Based	ECN Support
Cubic	✅		✅ (Classic)
BBRv1		✅	Does not support ECN
BBRv2		✅	✅ (Classic, AccECN)

Compete with TCP Prague (Scalable Congestion Control with AccECN)

Under what circumstances is TCP Prague performance

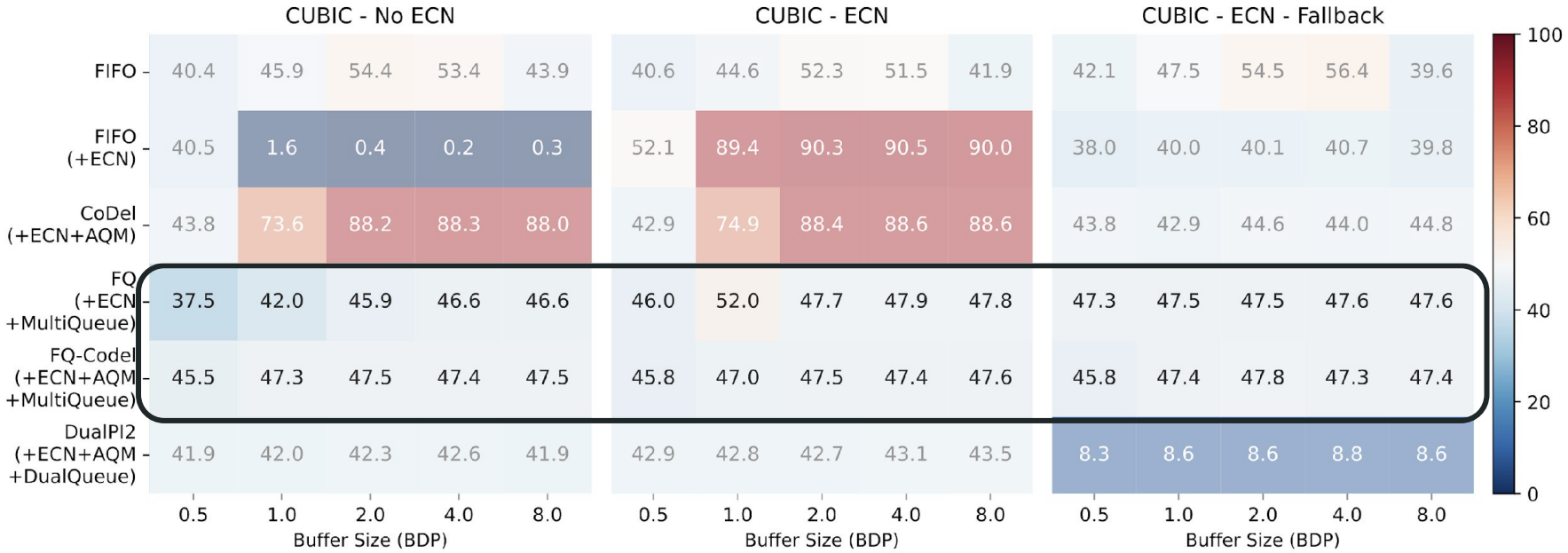
- **at least as favorable**
- and **not harmful**

to classic flows?

➔ With a queue that enforces fairness, TCP Prague coexists well with TCP CUBIC or TCP BBR.

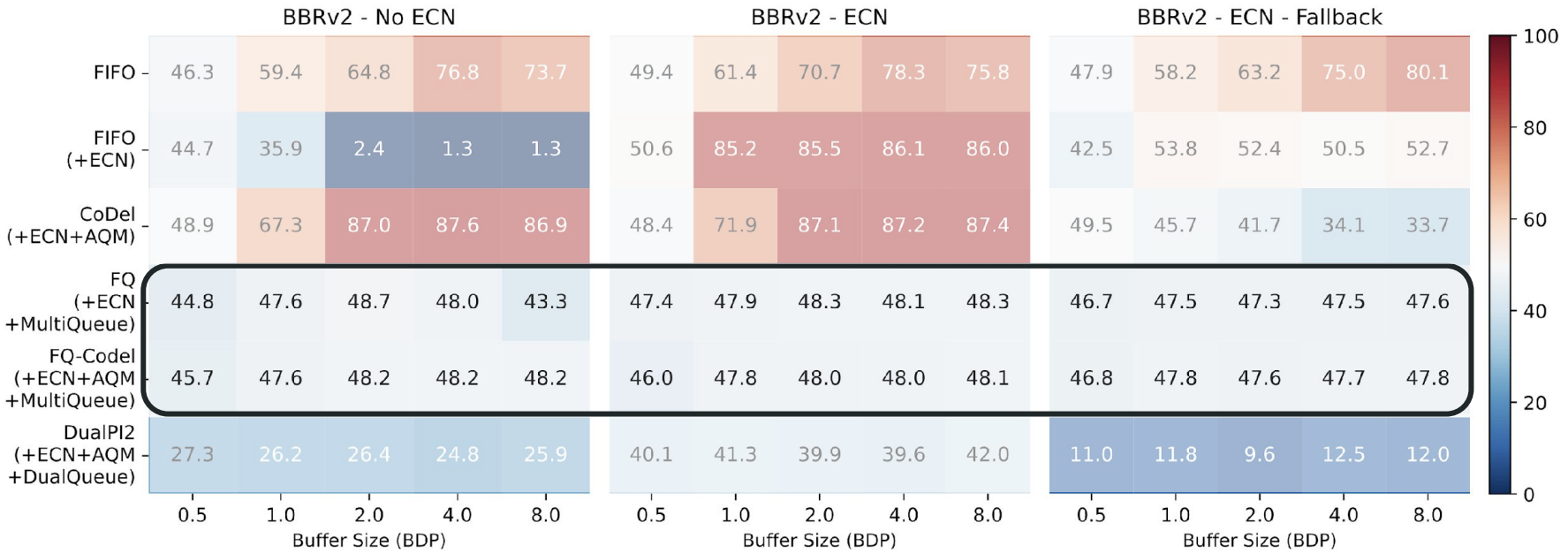
Prague Gets Its Fair Share of Throughput
Cubic Flow & FQ Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with Cubic flow.



Prague Gets Its Fair Share of Throughput
BBRv2 Flow & FQ Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with BBRv2 flow.



Under what circumstances is TCP Prague performance

- **not as favorable as**
- but still **not harmful**

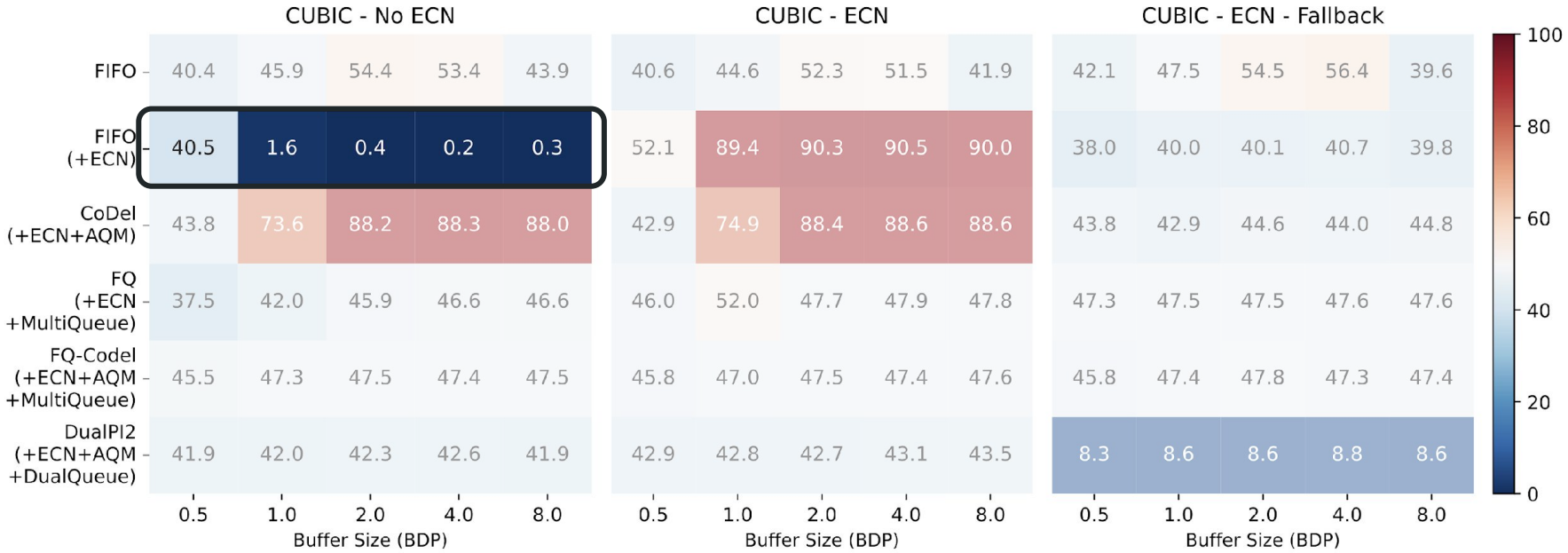
to classic flows?

➔ When sharing a single queue with a flow that does not respond to ECN.

➔ Also when sharing a DualPi2 queue with BBRv2.

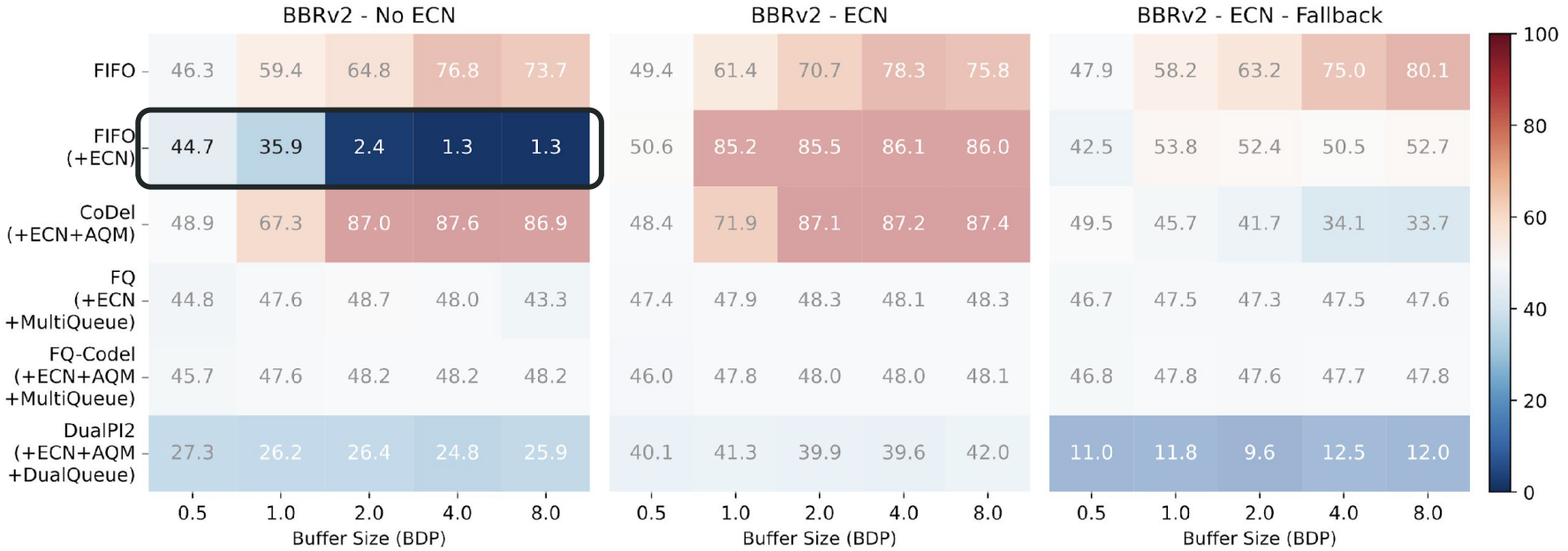
Prague Throughput is Degraded
No ECN Cubic Flow & Single Queue + ECN without AQM Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with Cubic flow.



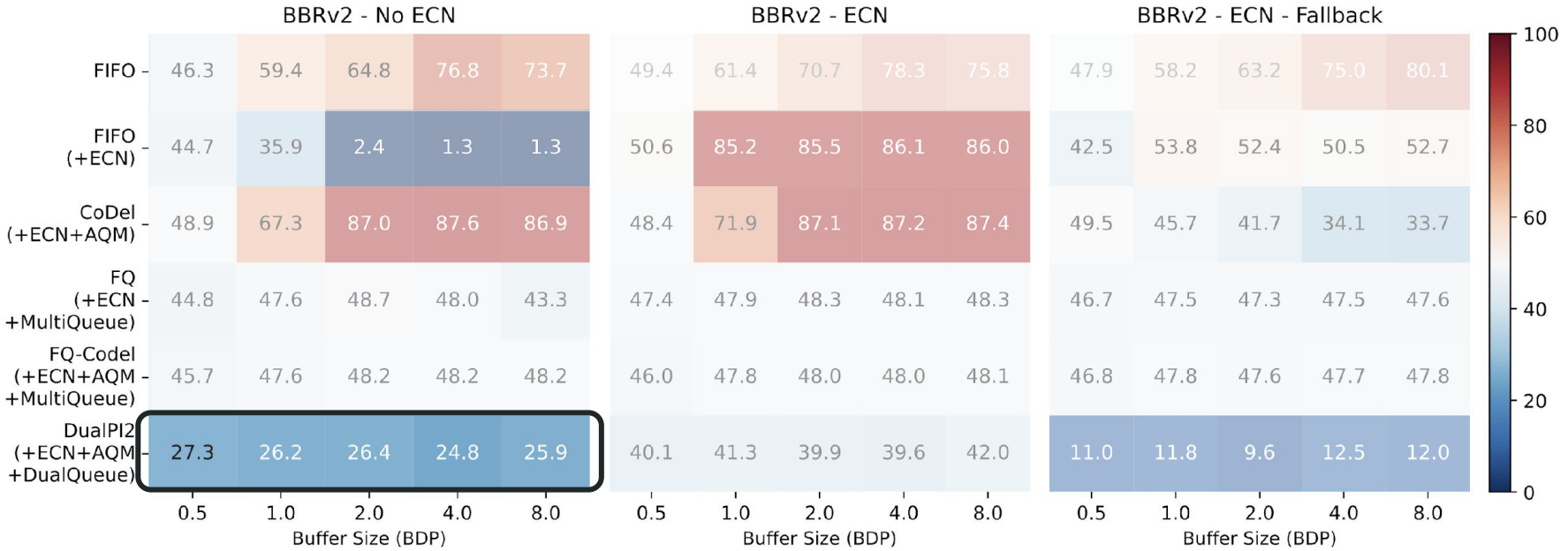
Prague Throughput is Degraded
No ECN BBRv2 Flow & Single Queue + ECN without AQM Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with BBRv2 flow.



Prague Throughput is Degraded
No ECN BBRv2 Flow & DualPI2 Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with BBRv2 flow.



Under what circumstances is TCP Prague performance

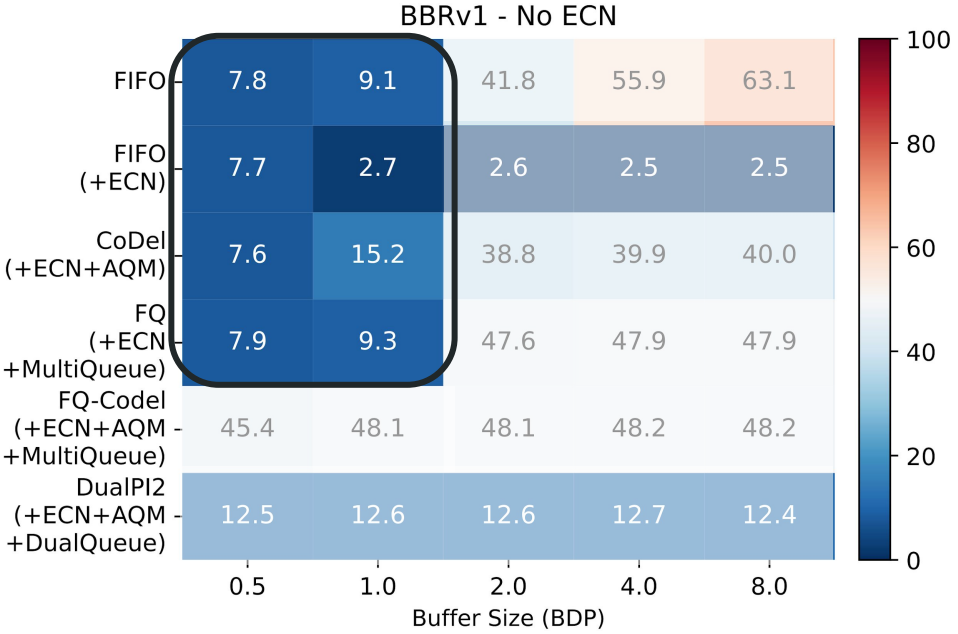
- **not as favorable as**
- but still **not harmful**

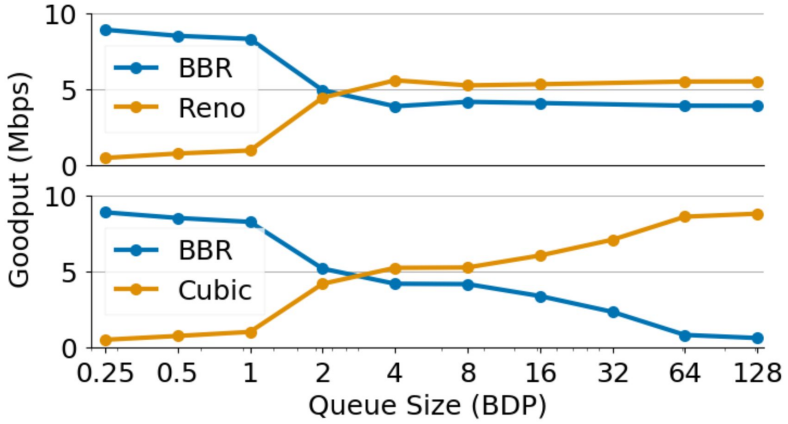
to classic flows?

➔ Note: BBRv1 dominates in a shallow buffer, but Prague is similar to Cubic/Reno in this setting.

Prague Throughput is Degraded BBRv1 Flow & Shallow Buffers & Single Queue + ECN without AQM Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with BBRv1 flow.





Ware et al., IMC 2019

Under what circumstances is TCP Prague performance

- **harmful**

to classic flows?

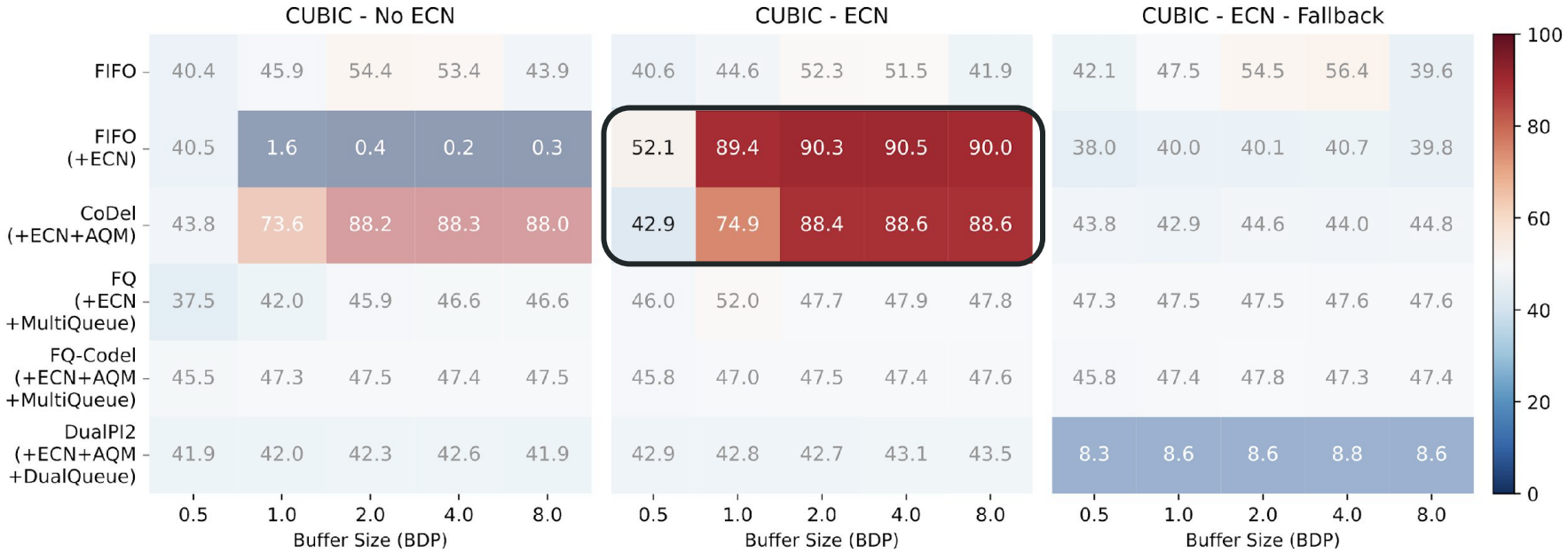
➔ When sharing a single ECN queue with a classic flow that responds to ECN.

➔ When sharing a single Codel queue with a classic flow that does not respond to ECN.

➔ When sharing a FIFO (non-ECN) queue with BBRv2.

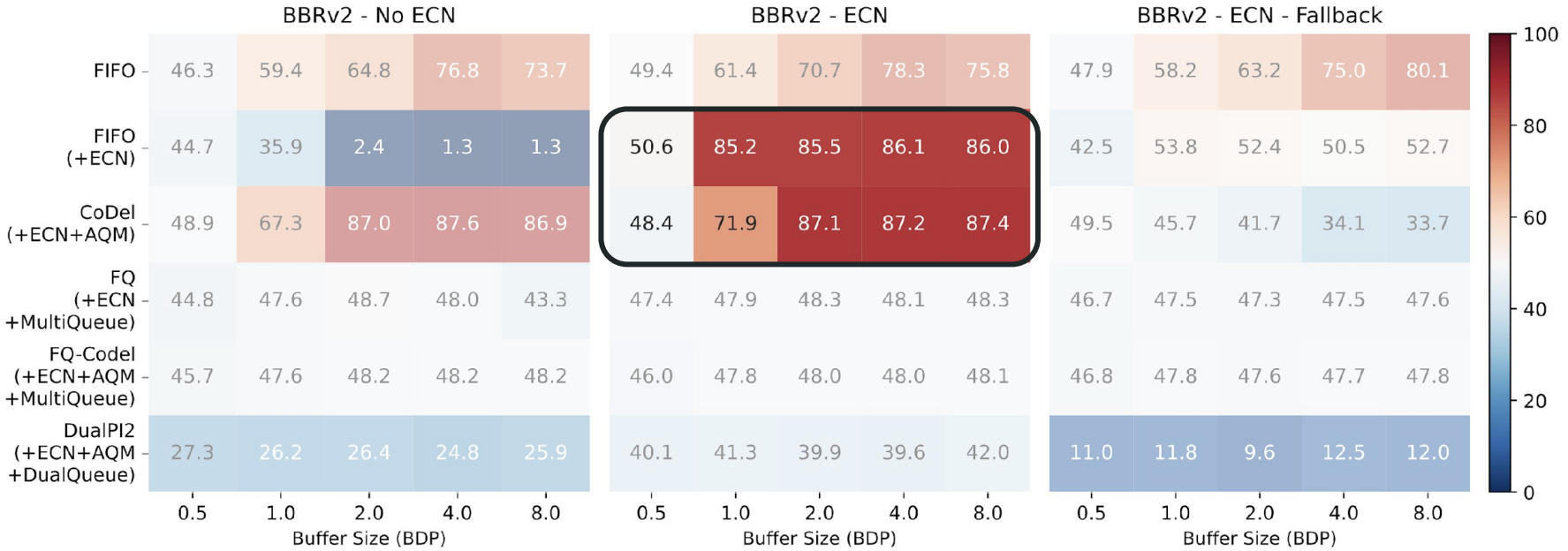
Prague Takes more than its Fair Share of Throughput
ECN Cubic Flow & Single Queue + ECN Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with Cubic flow.



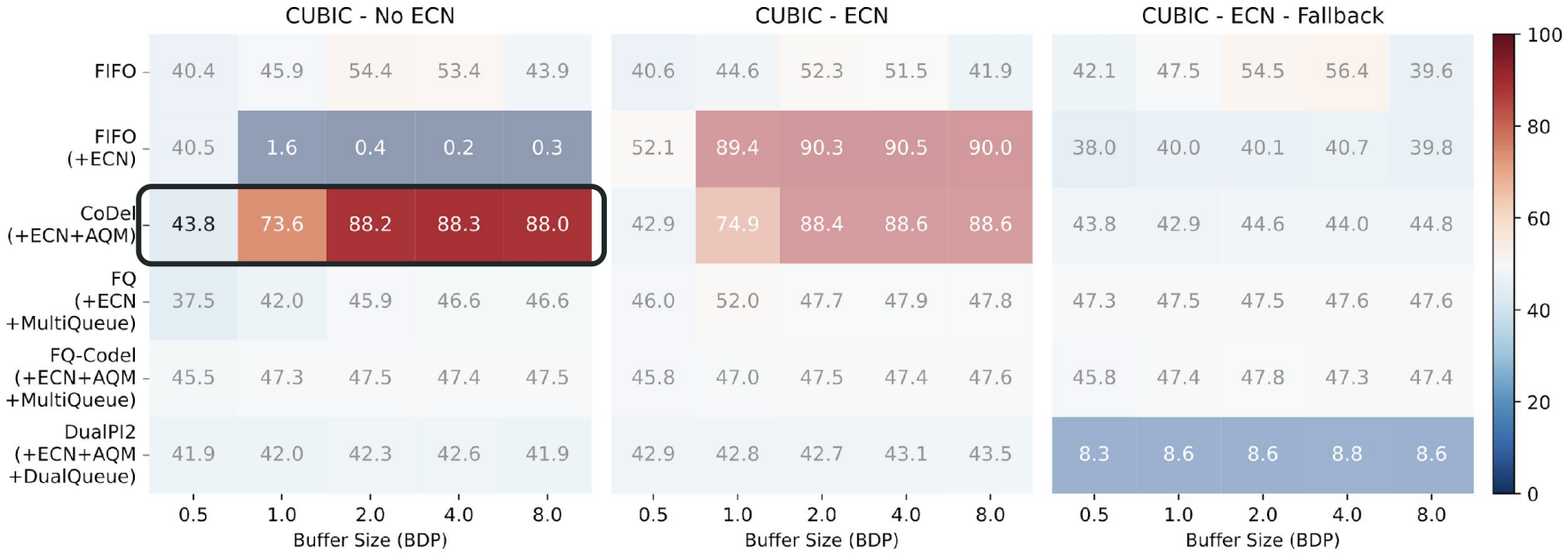
Prague Takes more than its Fair Share of Throughput
ECN BBRv2 Flow & Single Queue + ECN Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with BBRv2 flow.



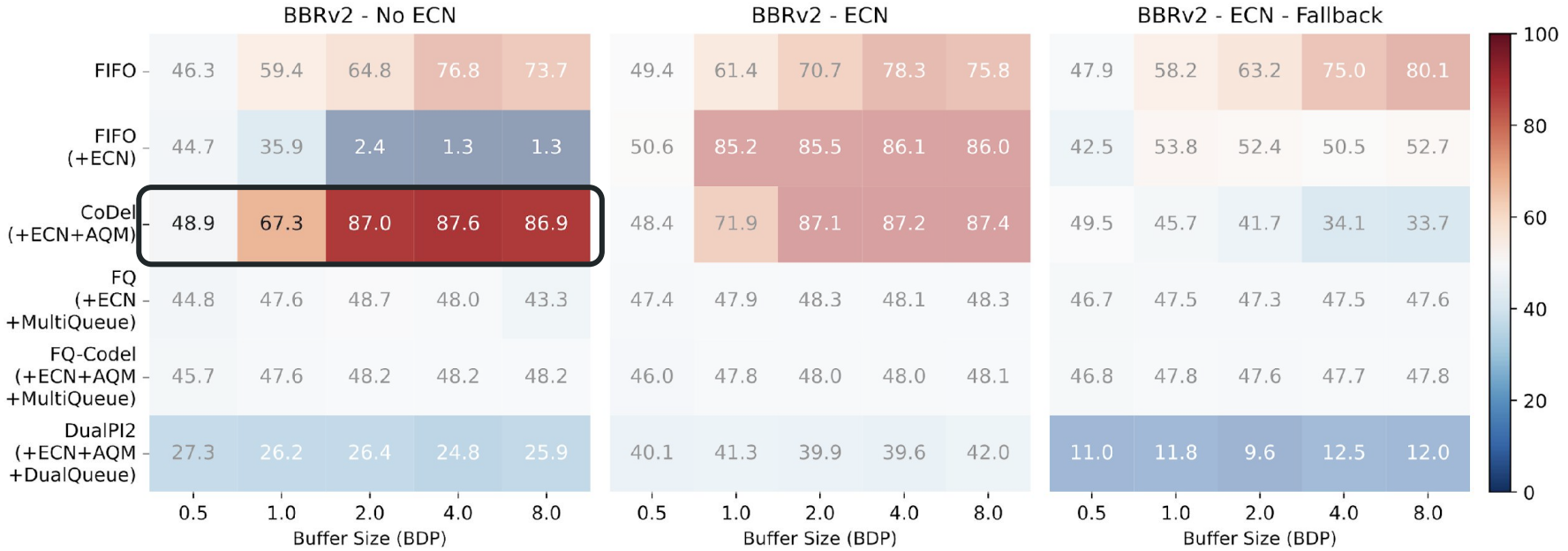
Prague Takes more than its Fair Share of Throughput
No ECN Cubic Flow & Single Queue + ECN with AQM Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with Cubic flow.



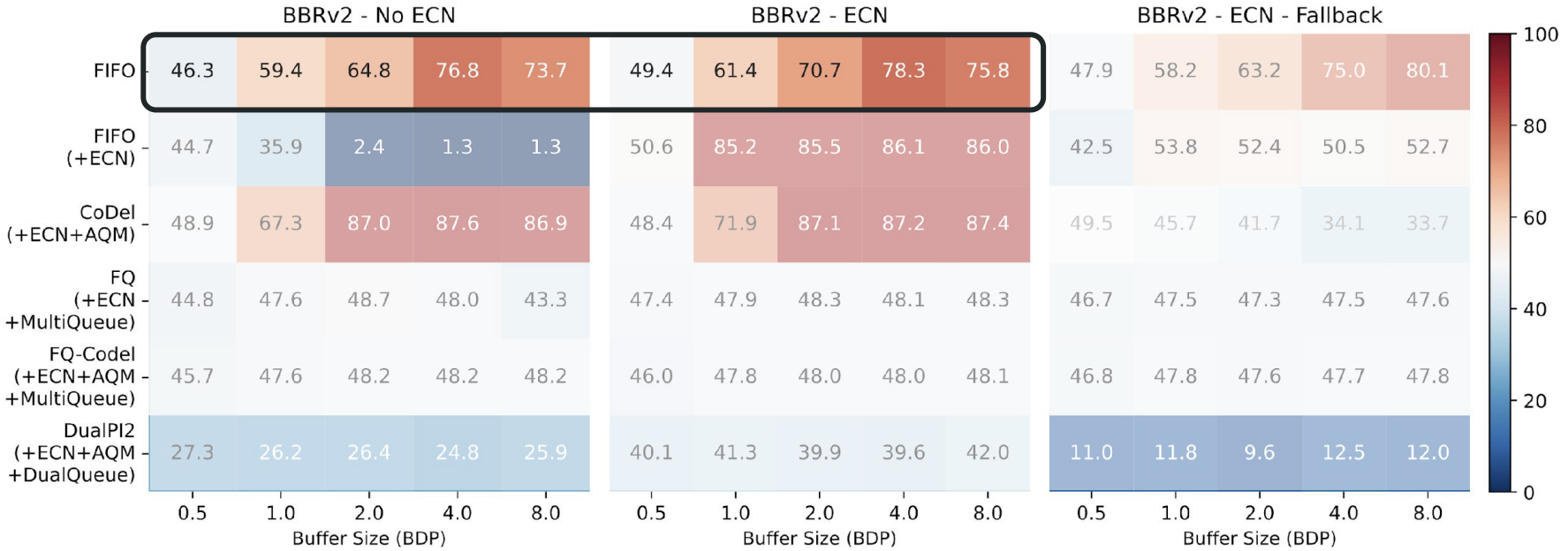
Prague Takes more than its Fair Share of Throughput
No ECN BBRv2 Flow & Single Queue + ECN with AQM Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with BBRv2 flow.



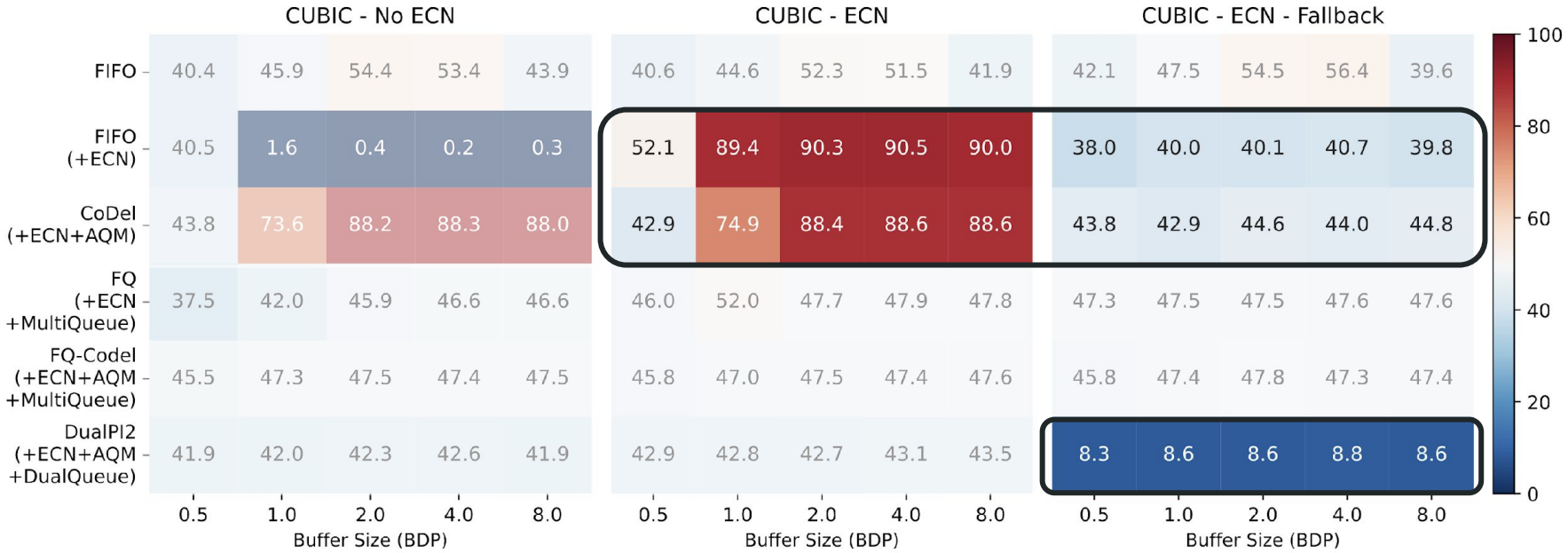
Prague Takes more than its Fair Share of Throughput
BBRv2 Flow & No ECN Bottleneck

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with BBRv2 flow.



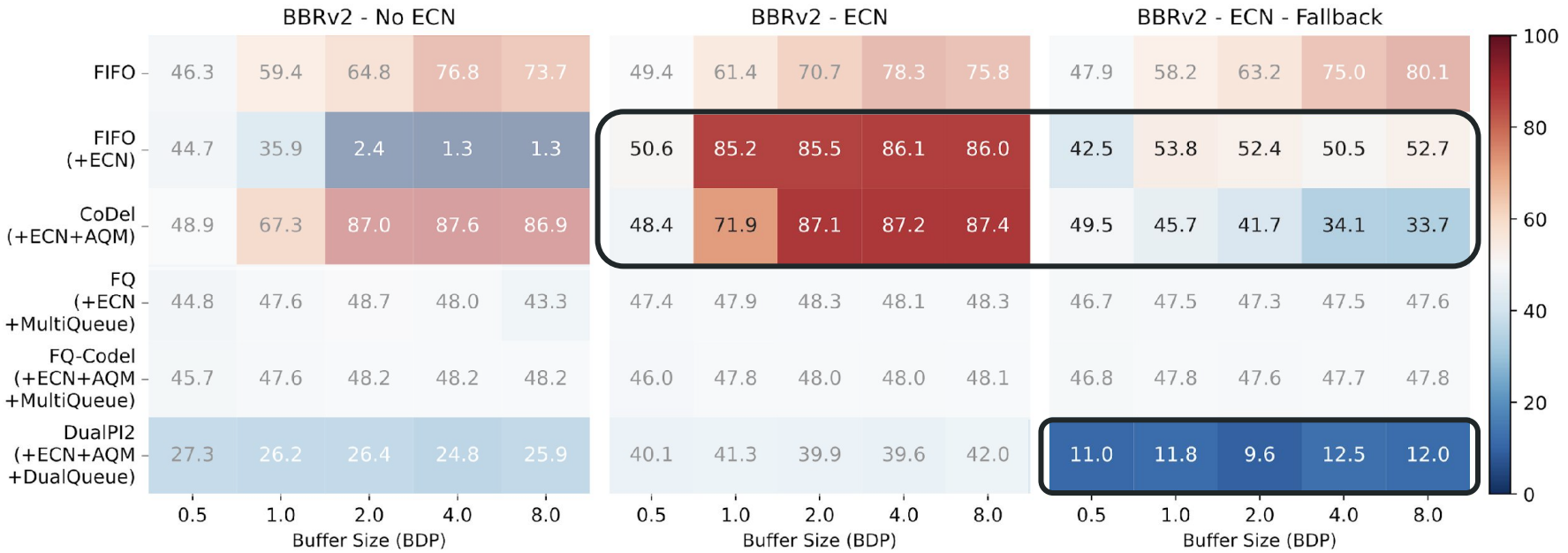
ECN Fallback Heuristic does not work well with DualPI2

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with Cubic flow.



ECN Fallback Heuristic does not work well with DualPI2

Prague throughput (Mbps) when sharing 100 Mbps bottleneck with BBRv2 flow.



Under what circumstances is TCP Prague performance

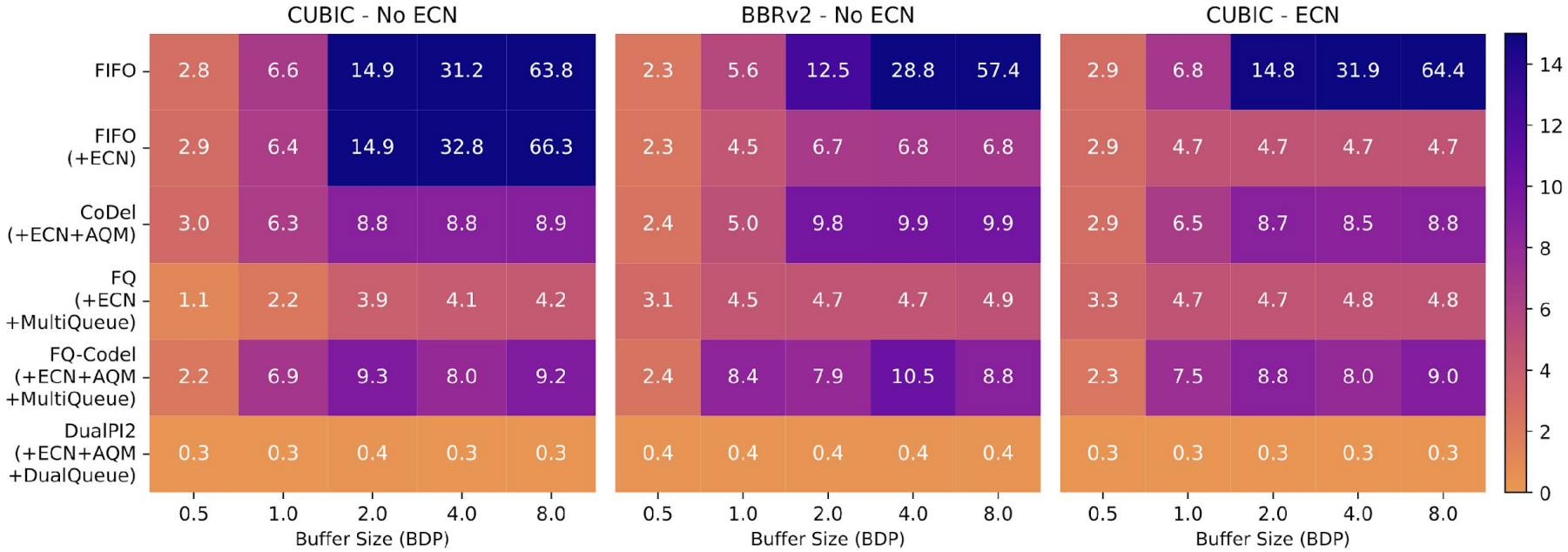
- **better** than

classic flows?

➔ Low-latency benefits are realized when the queue is DualPi2.

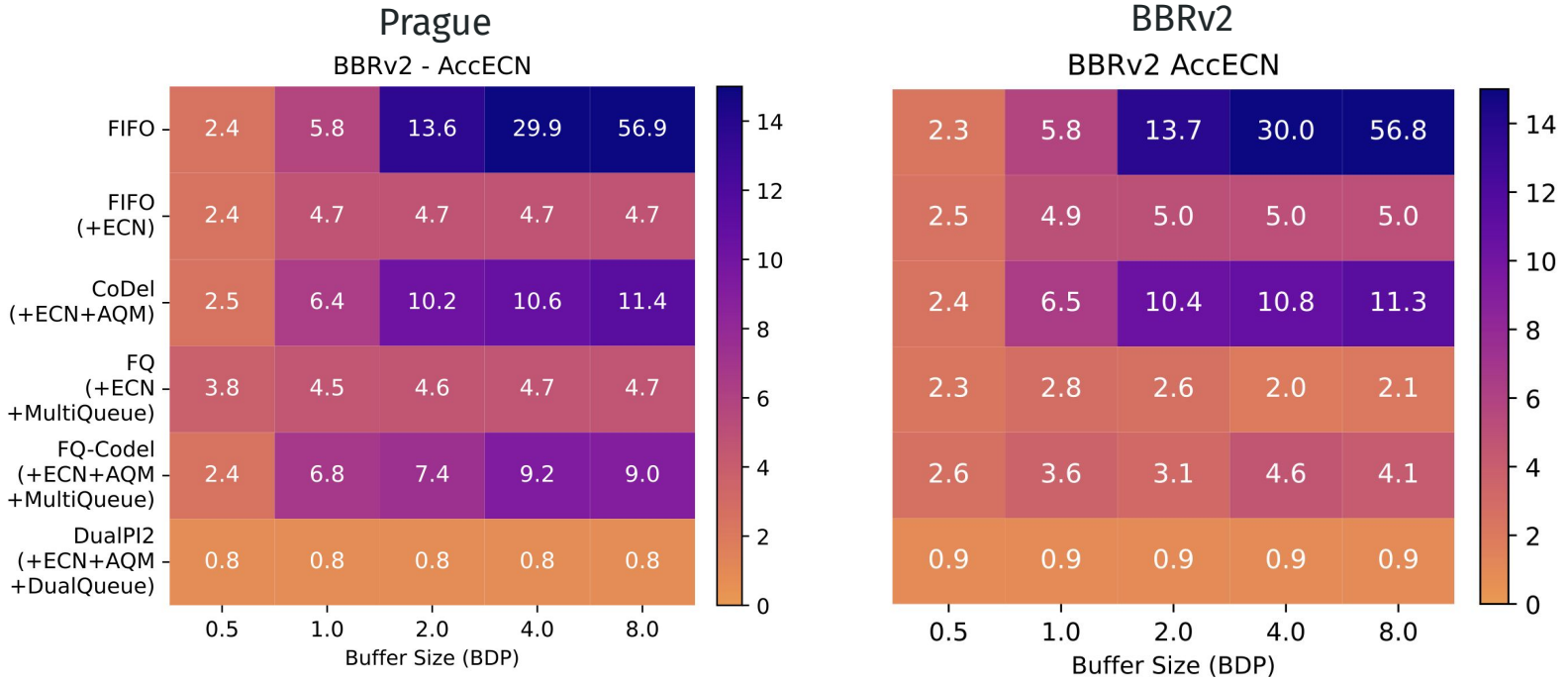
Ultra Low Latency is only Possible with DualPI2

Prague queuing delay (ms) when sharing bottleneck with legacy flow.
(ECN threshold is 5 ms, where applicable. For DualPI2, L4S queue has 1 ms threshold.)



BBRv2 AccECN also gets the low latency benefits of DualPI2

Queuing delay (ms) when sharing bottleneck with BBRv2 flow.
(ECN threshold is 5 ms, where applicable. For DualPI2, L4S queue has 1 ms threshold.)



Summary

Buffer Type	ECN Fallback OFF		ECN Fallback ON	
	CUBIC	BBRv2	CUBIC	BBRv2
SQ w/o ECN	✓	X	✓	X
SQ + ECN	X	X	✓	✓
FQ + ECN	✓	✓	✓	✓
DualPI2	✓	✓	X	X

Table 1: Is it okay to turn on TCP Prague or not? (SQ: single queue, FQ: fair queuing)

Thank You for Listening !

Experiment Artifacts:
https://github.com/fatihsarpkaya/L4S

Contact: fbs6417@nyu.edu