DualQ Coupled AQM

draft-briscoe-tsvwg-aqm-dualq-coupled-00

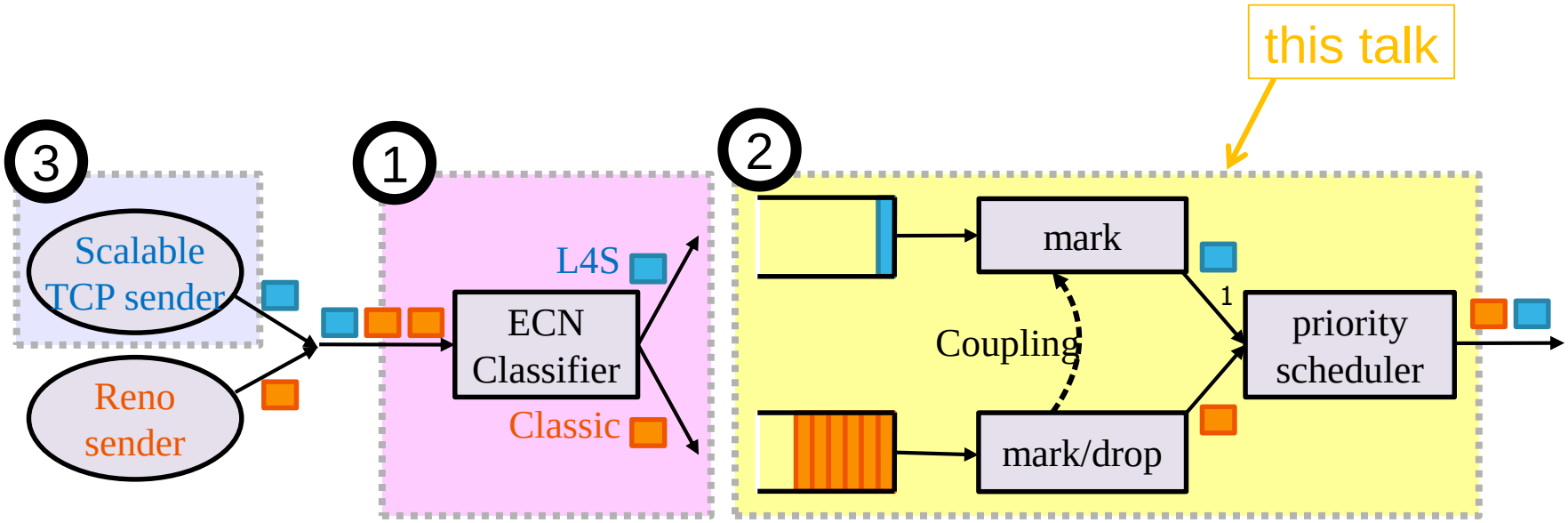
IETF-97 Nov 2016

Koen De Schepper
Bob Briscoe
Olga Bondarenko
Inton Tsang

The authors were part-funded by the European Community under its Seventh Framework Programme through the Reducing Internet Transport Latency (RITE) project (ICT-317700). The views expressed here are solely those of the authors.

L4S: low latency, low loss, scalable throughput

3 parts to standardise



1)	The identifier	draft-briscoe-tsvwg-ecn-l4s-id	tsvwg
2)	The DualQ AQM	draft-briscoe-tsvwg-aqm-dualq-coupled	tsvwg?
3)	Scalable transports	many	?

Updated version available

Name change: aqm-…-02 → tsvwg-aqm-…-00

Added Dual-PI2 as alternative to CurvyRED

- Reference to PI2 paper
- Dual-PI2 pseudo-code

Improved overload for both PI2 and CurvyRED:

- Time-shifted FIFO pseudo code
- Tail-drop on overload

ECN – Drop fairness problem
(not only for DualQ!!)

Needs special overload considerations because:

goodput for “100% drop” <> “100% mark”

Window at least 2MTU → ECN becomes unresponsive

Equal Window up to ~25% drop | ~25% Classic-mark
| ~100% DCTCP*-mark

Above ~25% not-ect traffic starves

→ reasonable overload threshold

* Different when L4S/TCP-Prague supports Window < 2MTU

Overload strategies

AQM is no flow policer !

- Optional separate function
- Standalone AQM still needs to handle overload

2 possible strategies for overload protection

a) Limit AQM drop / mark → rely on tail-drop
   - Sacrifices latency
   - Avoids drop of ECN traffic when Q not overflowing

b) Switch to Classic AQM drop for all
   - Preserves low latency

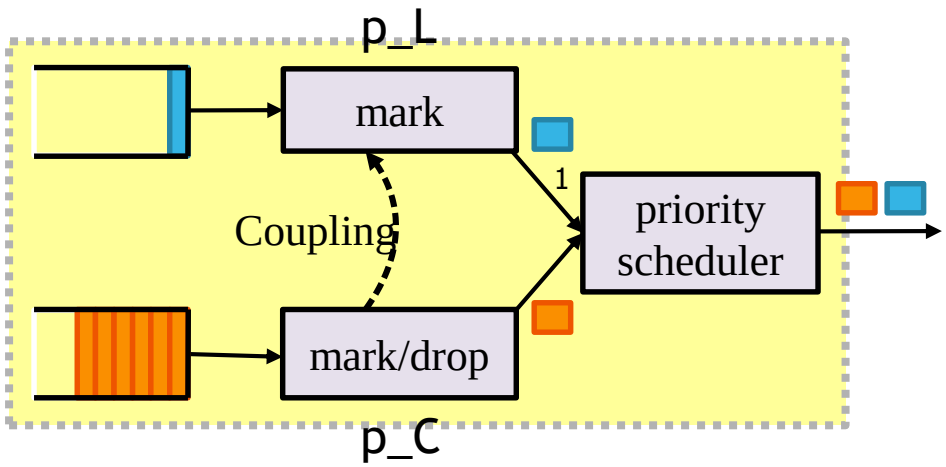
Following overload experiments show a) drop/mark limit → tail-drop

Coupling: $p_C = (p_L/k)^2$

k =2 → Limit p_L to 100%

→ p_C = 25%

(happy coincidence ☺)



Link: 100Mbps, 7ms base RTT

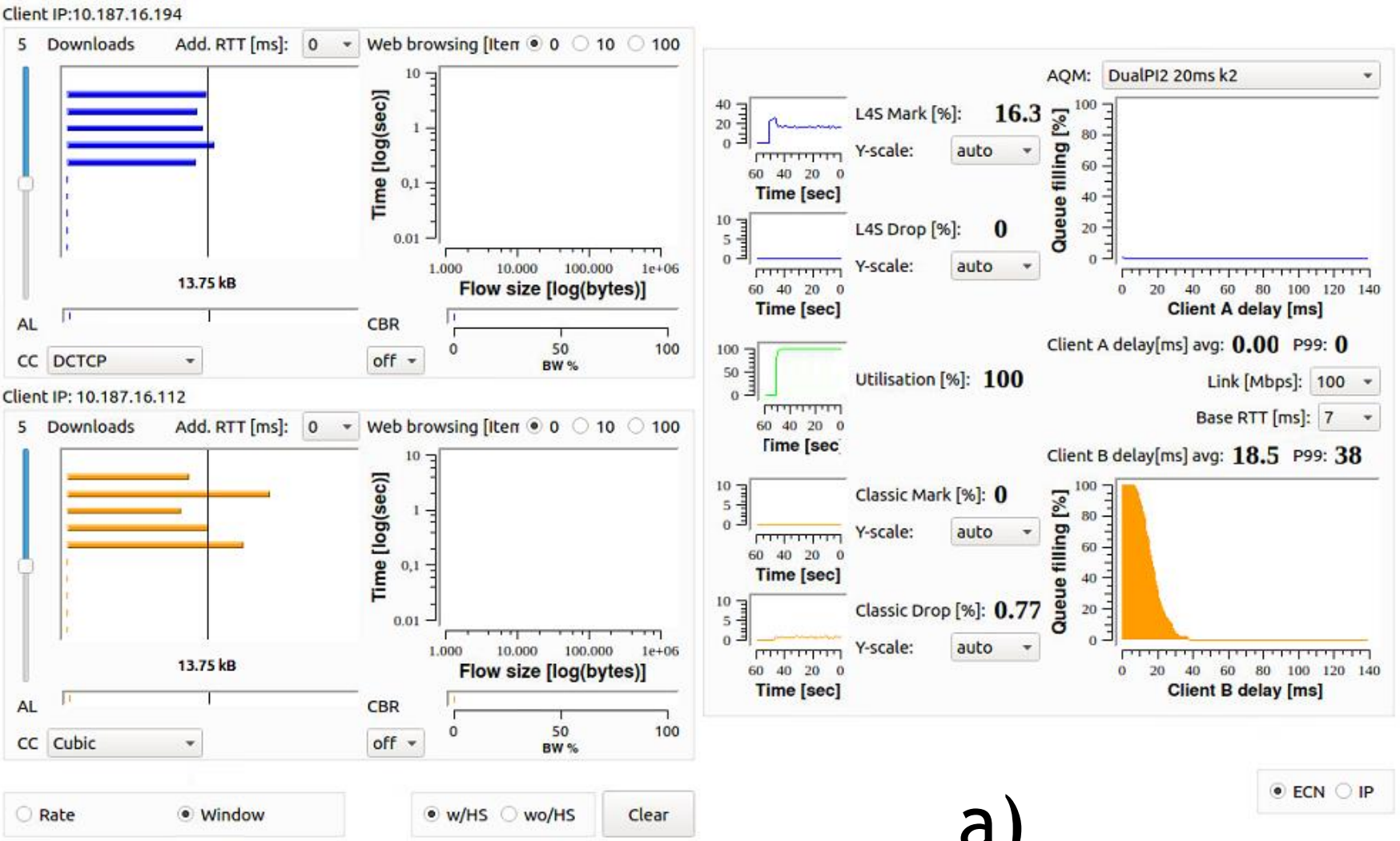
Classic Target: 20ms

5 TCP flows of each class

UDP traffic of 50, 100, 200Mbps

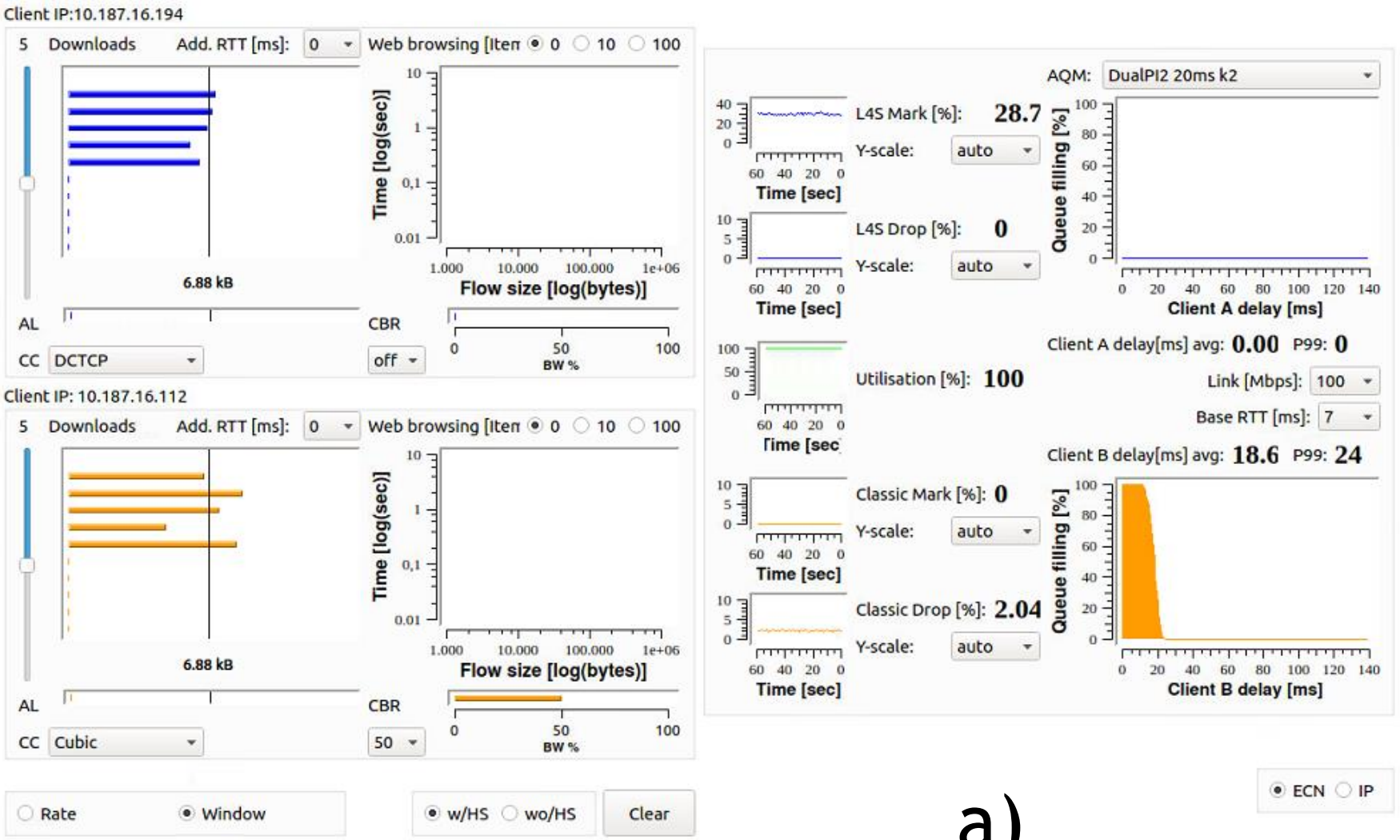
No unresponsive traffic

10 TCP on 100Mbps Baseline



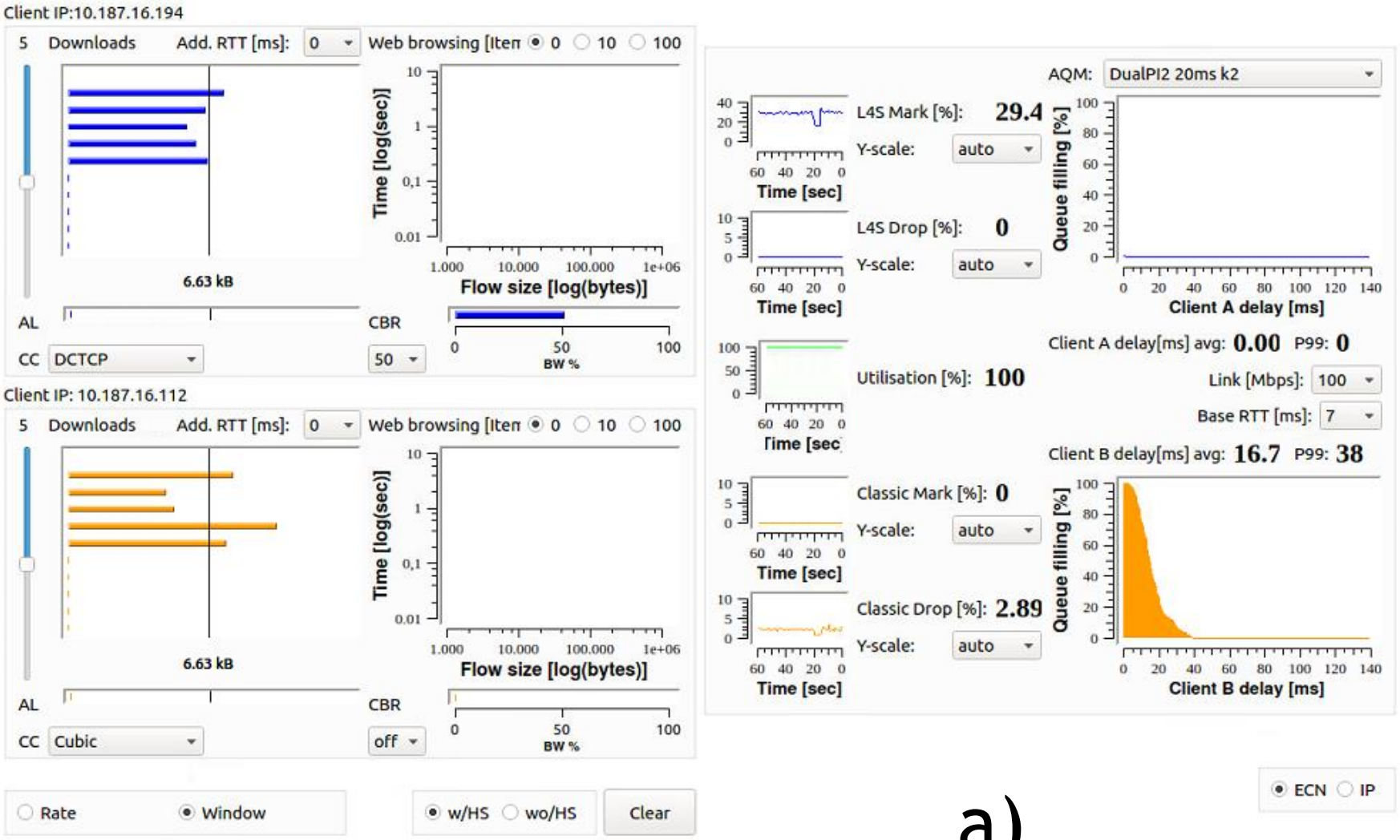
a)

50Mbps unresponsive Classic UDP traffic
Rest is shared fairly



a)

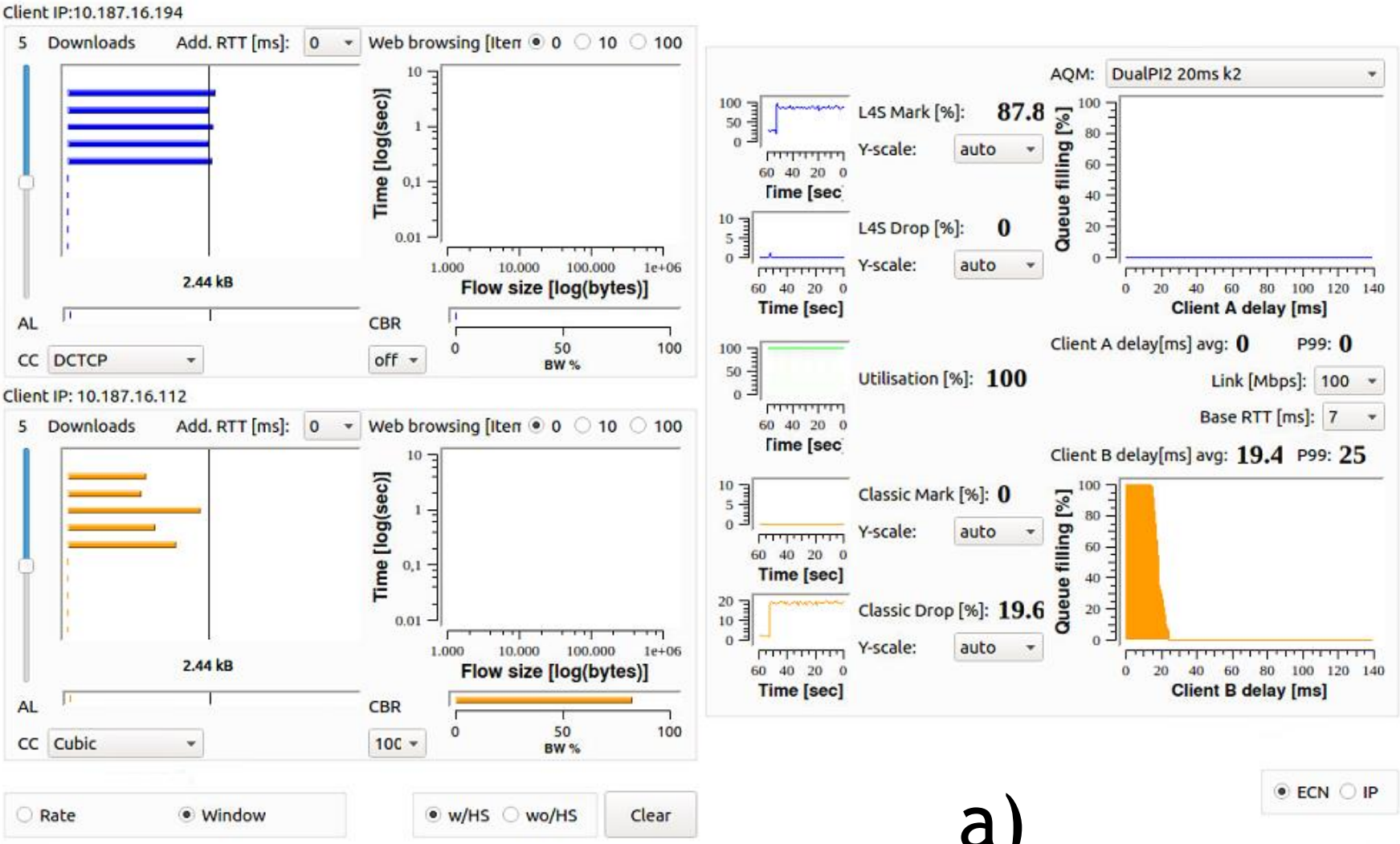
50Mbps unresponsive L4S UDP traffic
Rest is shared fairly



a)

100Mbps unresponsive Classic UDP traffic

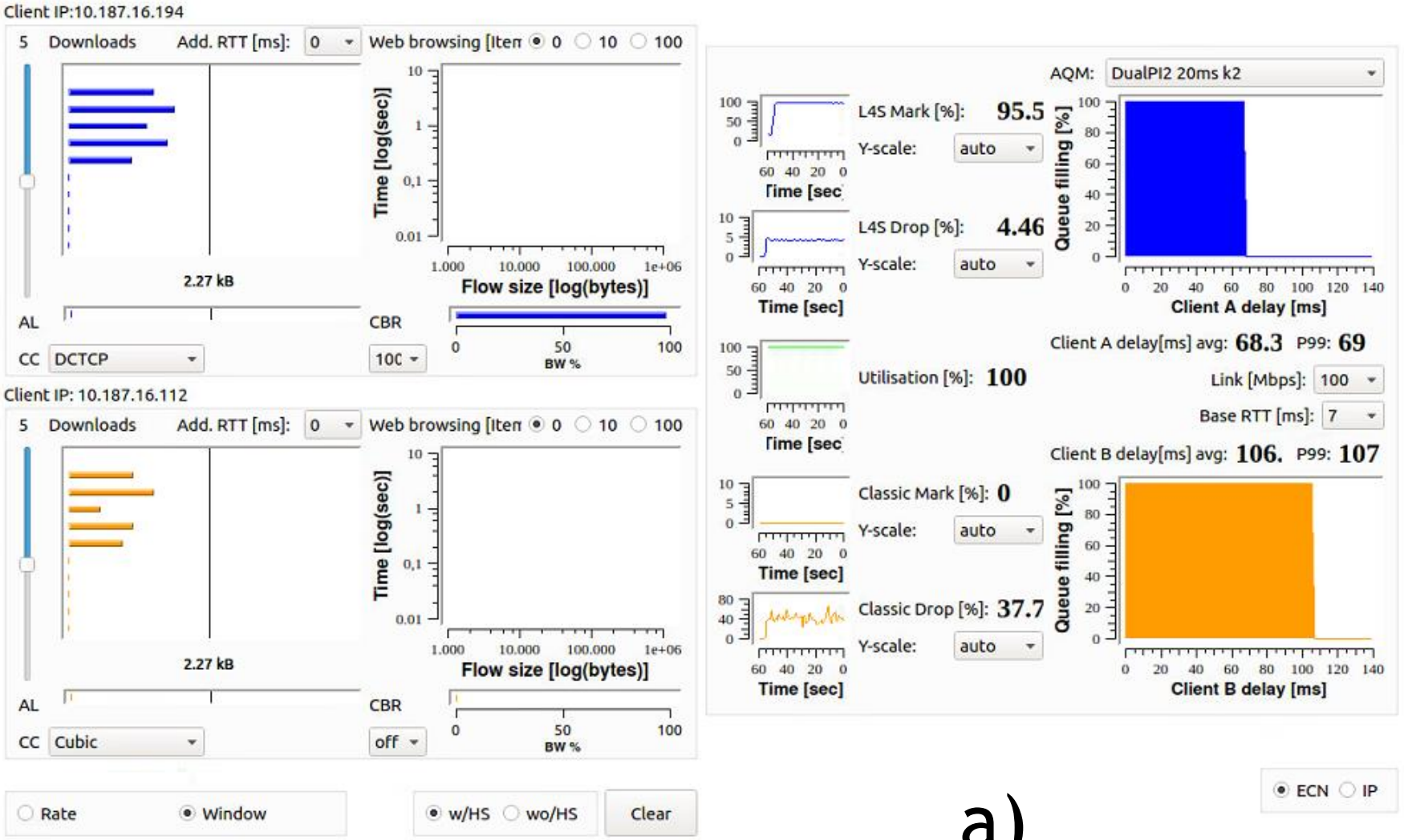
Drop below 25%, still fair



a)

100Mbps unresponsive L4S UDP traffic

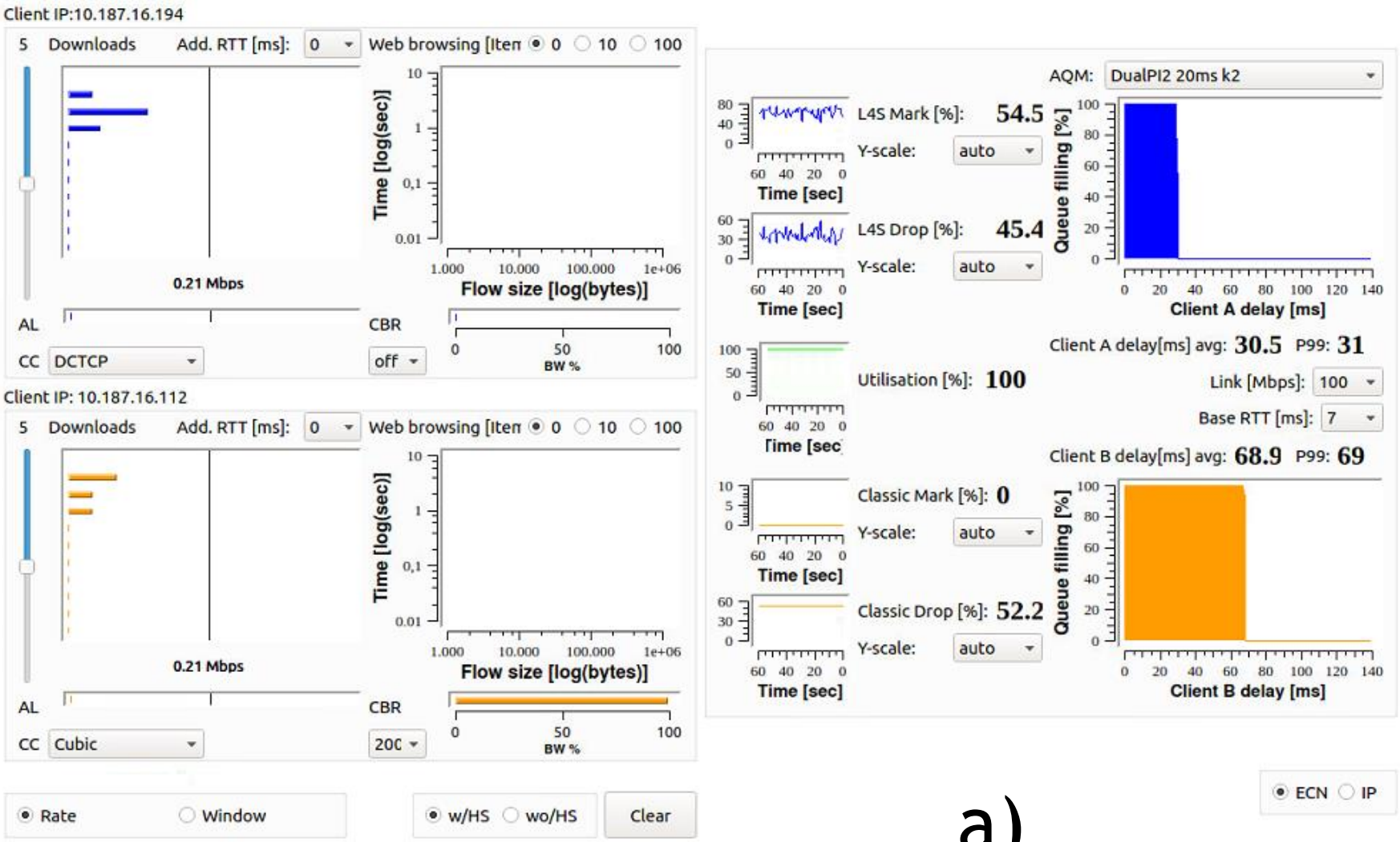
Controlled drop < 25% → tail drop



a)

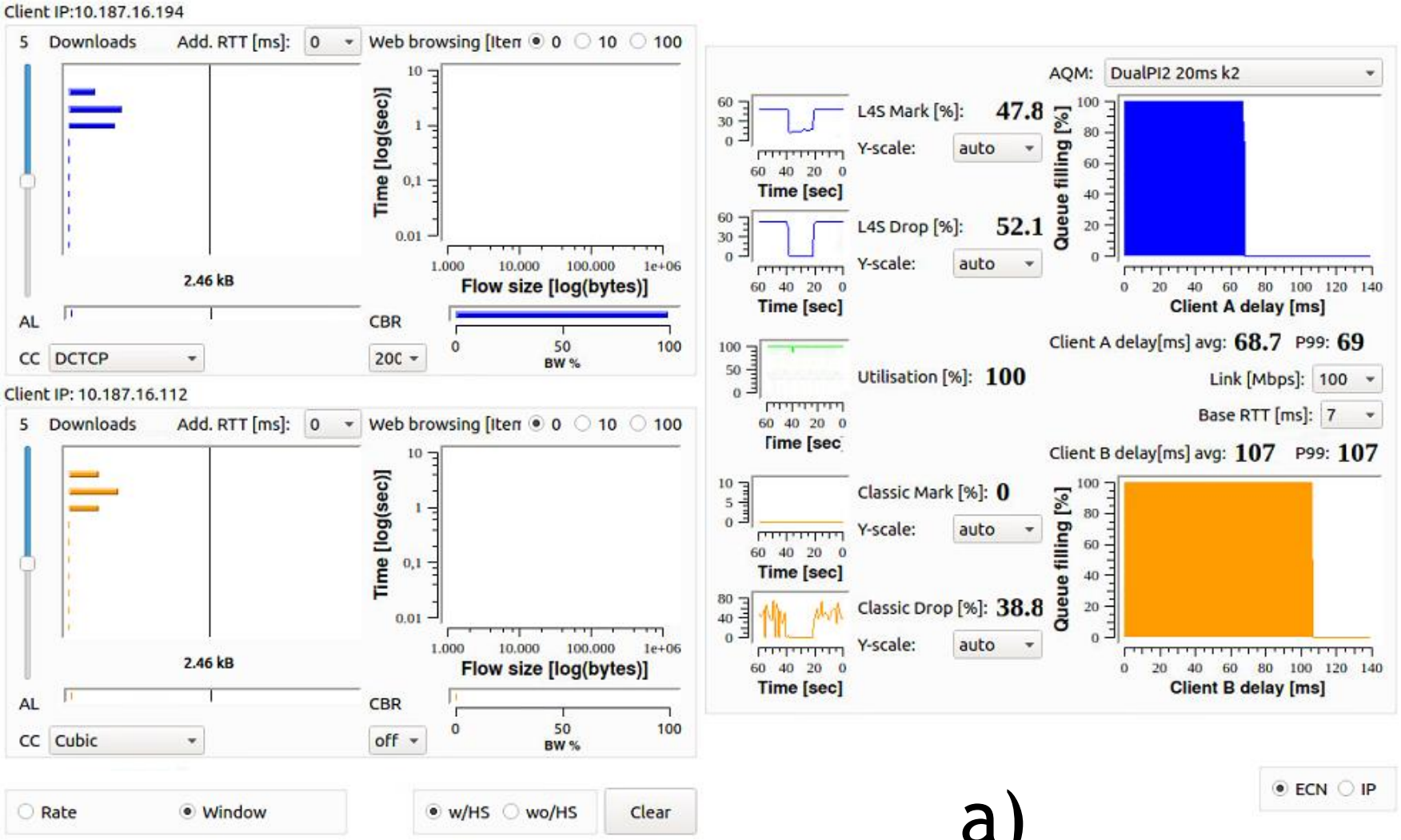
200Mbps unresponsive Classic UDP traffic

52% drop 69ms delay



a)

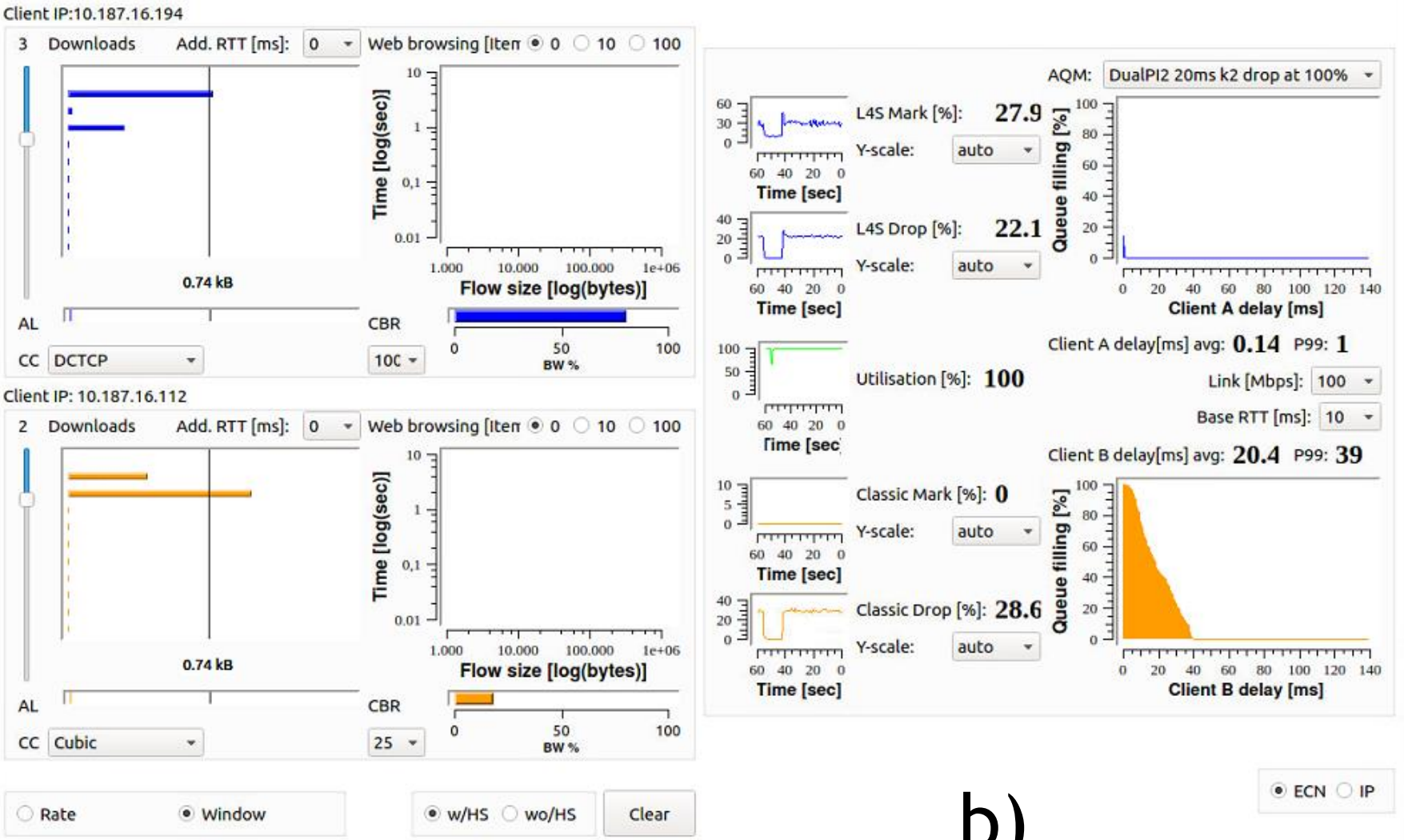
200Mbps unresponsive L4S UDP traffic
also 52% drop 69ms delay



a)

Switch to Classic drop for all

Preserves low latency Q



b)

Adoption of draft?

- Please review, comment, implement and discuss further on tsvwg@ietf.org cc: tcpprague@ietf.org

- Ready for adoption with only DCTCP experience?

- Is it OK to evolve DualQ for TCP-Prague after adoption?