# Some Congestion Experienced

**draft-morton-tsvwg-sce-01**
**draft-morton-tsvwg-codel-approx-fair-00**
**draft-morton-tsvwg-lightweight-fair-queueing-00**
**draft-morton-tsvwg-cheap-nasty-queuing-01**

Jonathan Morton
Pete Heist
Rodney W Grimes

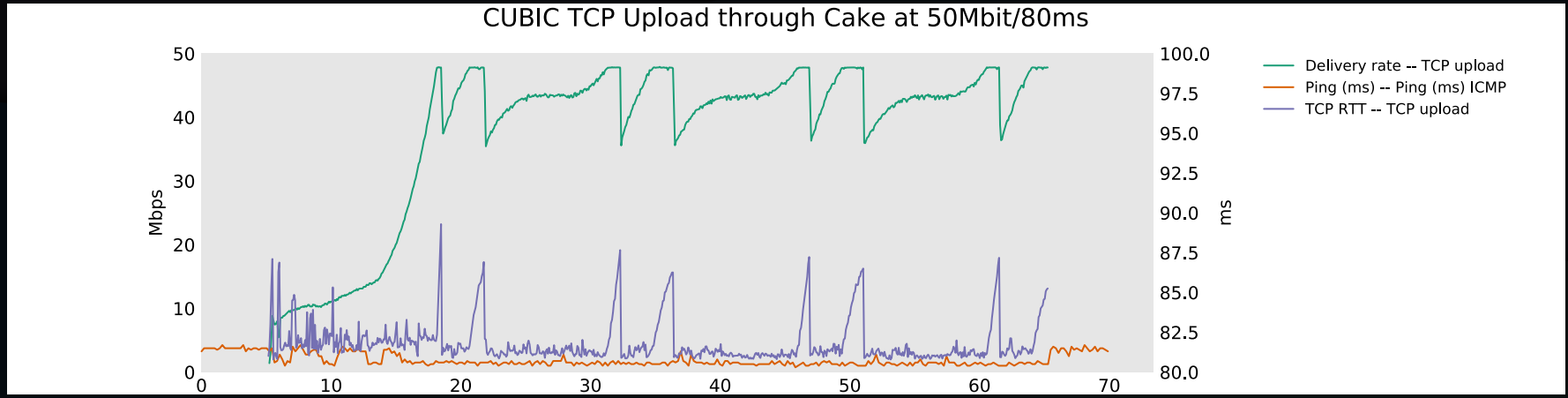
IETF Interim Meeting 2020.02.20

# Overview

- Problem Statement and Goals
- SCE Signaling
- Co-existence with non-SCE Traffic
- Prague Requirements
- Difficult Environments

# SCE Problem Statement

*Existing congestion control signals and responses are safe, but can lead to under-utilization and spikes in queue depth*



# SCE Goals

- Define a high fidelity congestion signal, which can be used to:
  - Decrease latency and jitter
  - Increase utilization

- Safety
  - Existing signal compatibility
  - No signaling ambiguity
  - Seamless bottleneck shifts
  - Safety enables innovation
- Simplicity
  - Ease of implementation
  - Robust failure modes

# The SCE Signal

SCE adds a third signal:

| Signal | Status | Response | Notes |
|---|---|---|---|
| Drop | Existing, Jacobson88 | 50% mult. decrease | Reliable, imprecise |
| CE | Existing, RFC 3168 | ABE mult. decrease | Reliable, imprecise, avoids drop |
| SCE | Proposed use of ECT(1) | Small backoff | Unreliable, precise, avoids CE |

## Why a separate signal?

- Compatibility with millions of existing RFC 3168 ECN AQM instances
  - CE's MD without drop still useful for sudden capacity reductions

# The SCE Signal Compared

SCE mark is essentially similar to
L4S CE mark, but non-ambiguous

| Request | RFC-3168 | SCE | L4S |
|---|---|---|---|
| Large decrease | Single CE | Single CE | O(cwnd) CEs |
| Small decrease | | Single SCE | Single CE |
| Steady state | <1 CE per RTT | >1 SCE per RTT | >1 CE per RTT |
| Growth permit | ECT(0) | ECT(0) | ECT(1) |

# RFC 3168 ECN and RFC 8511 ABE

ECN: Reliable, imprecise, max 1 per RTT, avoids drop


## Sender:

Standard response: $\beta_{ECN} = 0.5$

ABE response: $\beta_{ECN} = 0.7\text{-}0.85$

ABE response useful in SCE context

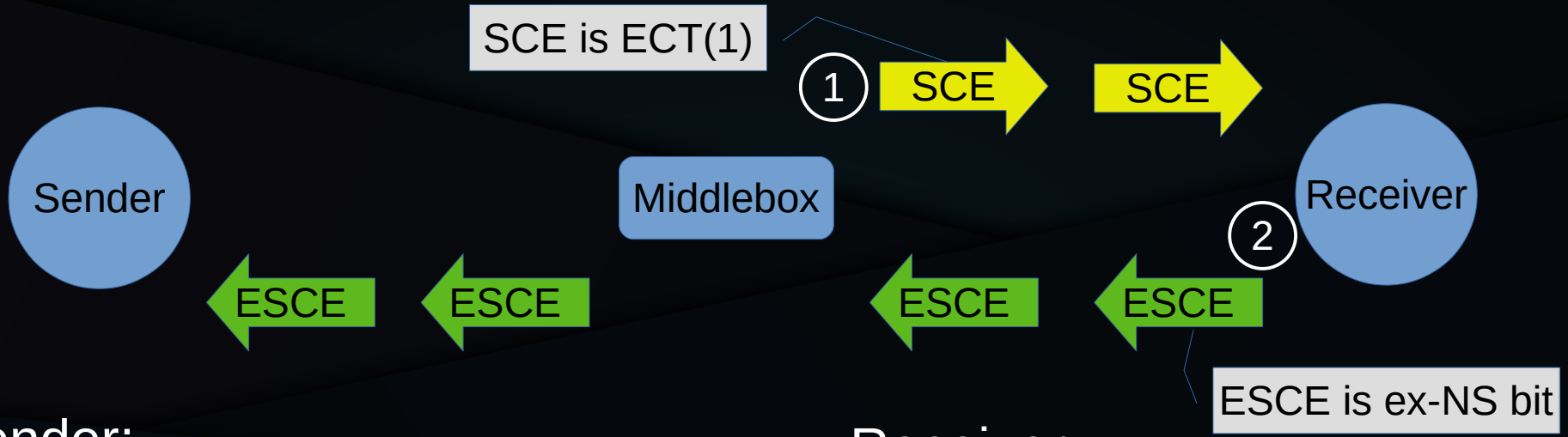
## Why retain CE semantics?

1) Safety, compatibility

2) For sudden capacity reductions

3) As a “backstop” signal for SCE

# SCE Signaling- Many Signals / RTT




## Sender:

- Respond to ESCE w/ small cwnd reduction:
  - DCTCP: ½ ESCE-flagged data
  - ELR: sqrt(cwnd segs) * ESCE-flagged data
- Non-SCE senders do nothing

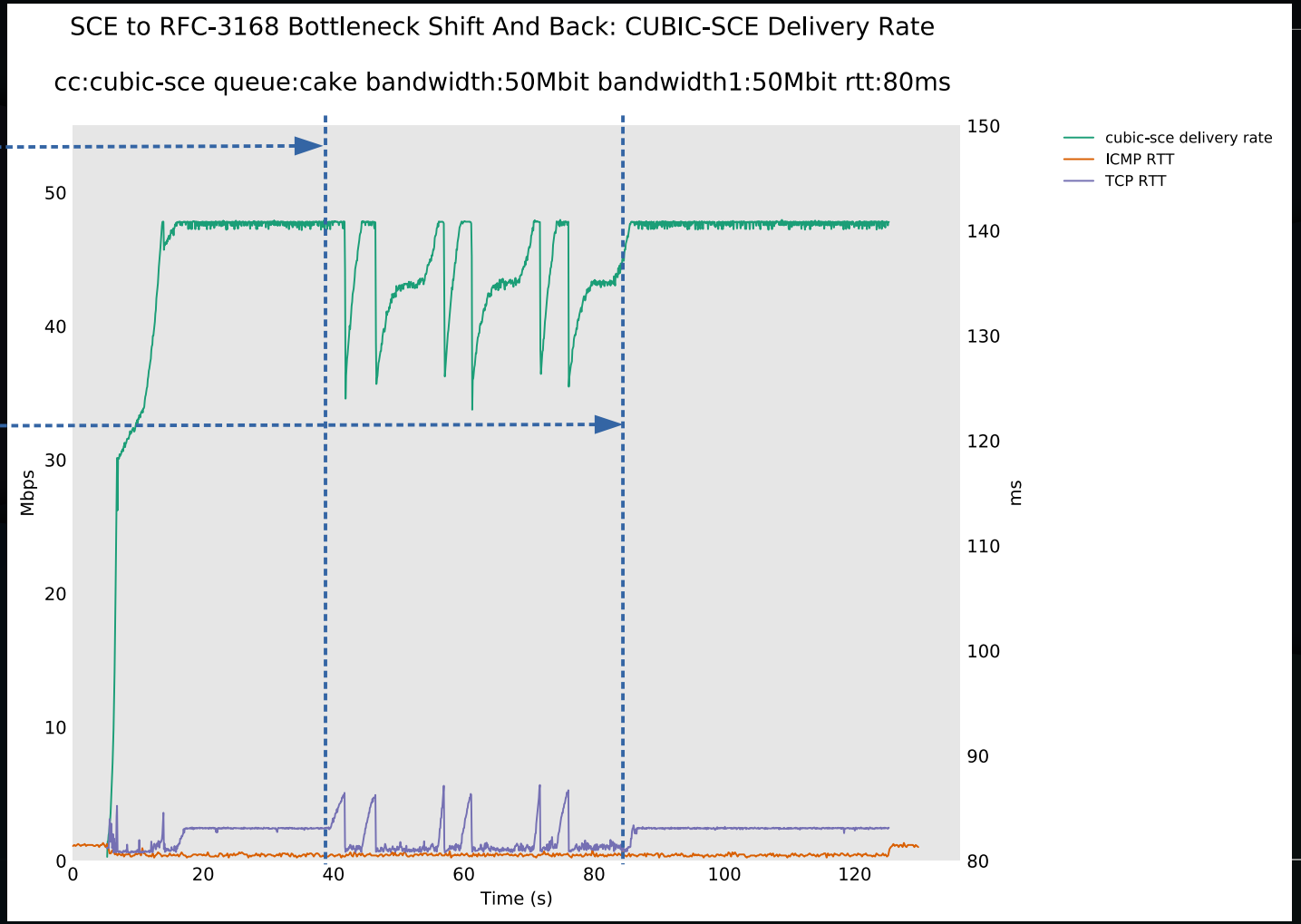
## Receiver:

- Echo SCE to ESCE
- Up to 100% relative error tolerated
- Non-SCE receivers do nothing

# Test Scenario: Bottleneck Shift

Shift to RFC 3168 bottleneck

Shift back to SCE bottleneck

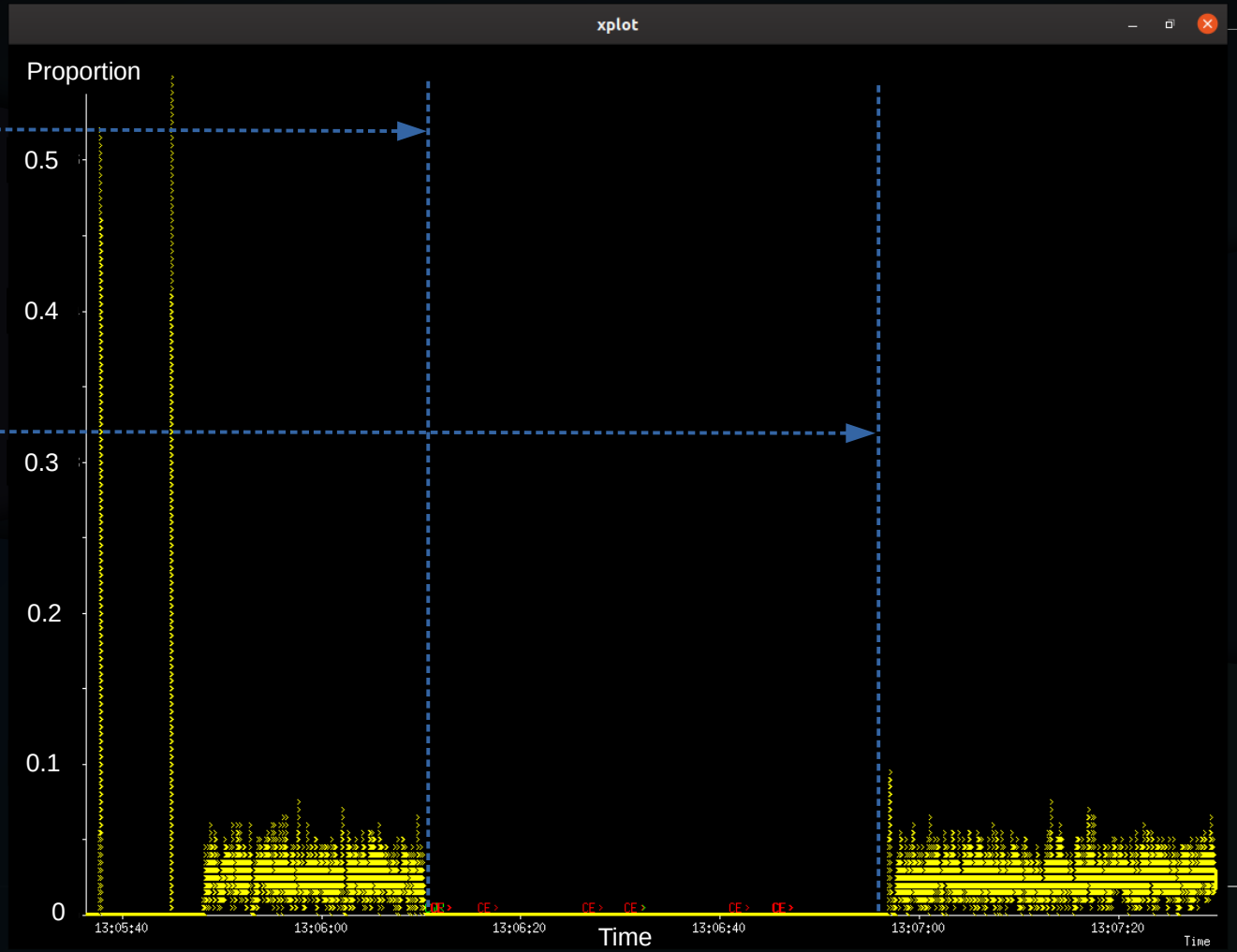


# Test Scenario: Bottleneck Shift

Shift to RFC 3168 bottleneck

Shift back to SCE bottleneck

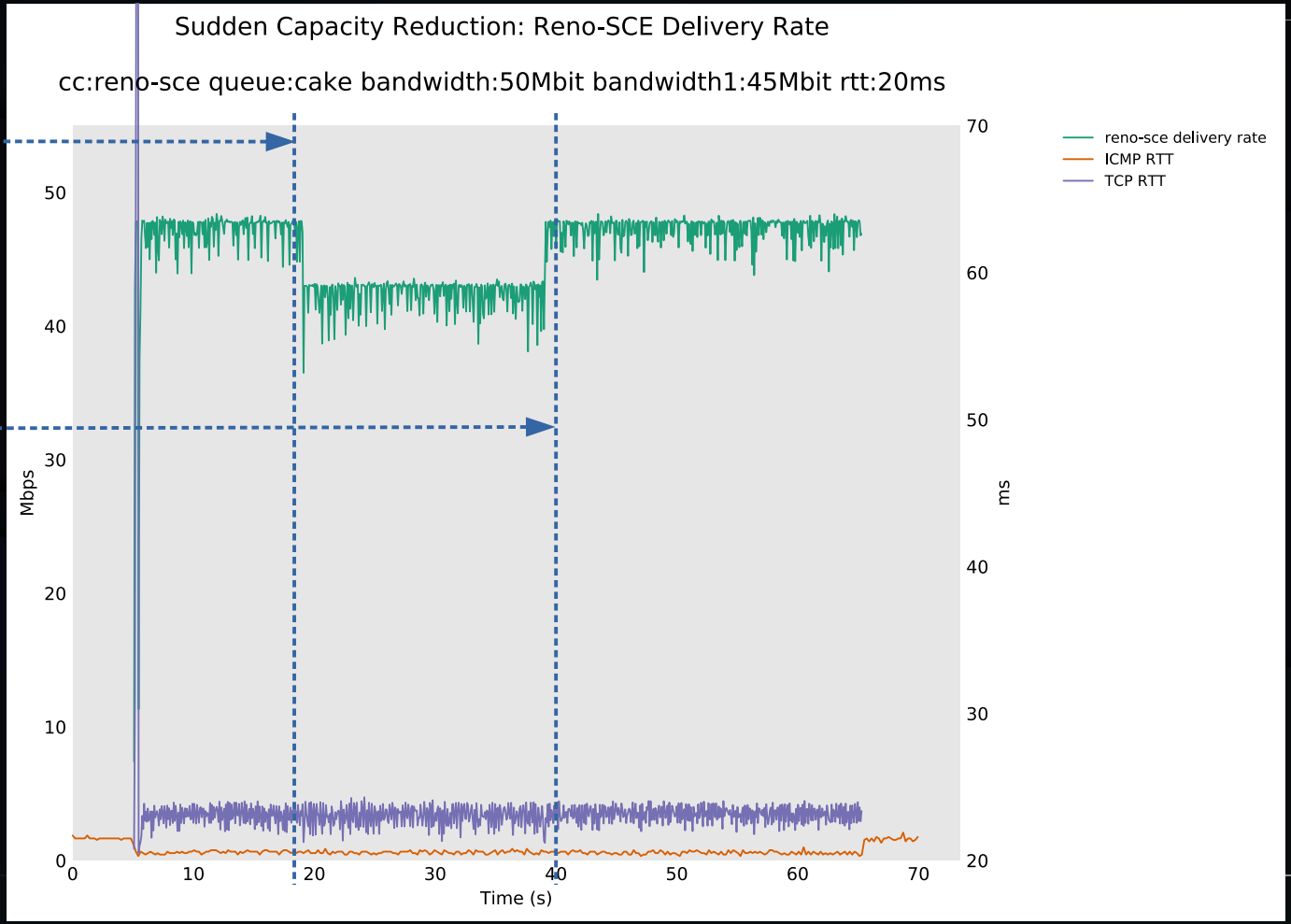
> SCE marks
> CE marks
> CWR



# Test Scenario: Capacity Reduction 1

Capacity reduction **50-45Mbit**

Capacity return to 50Mbit

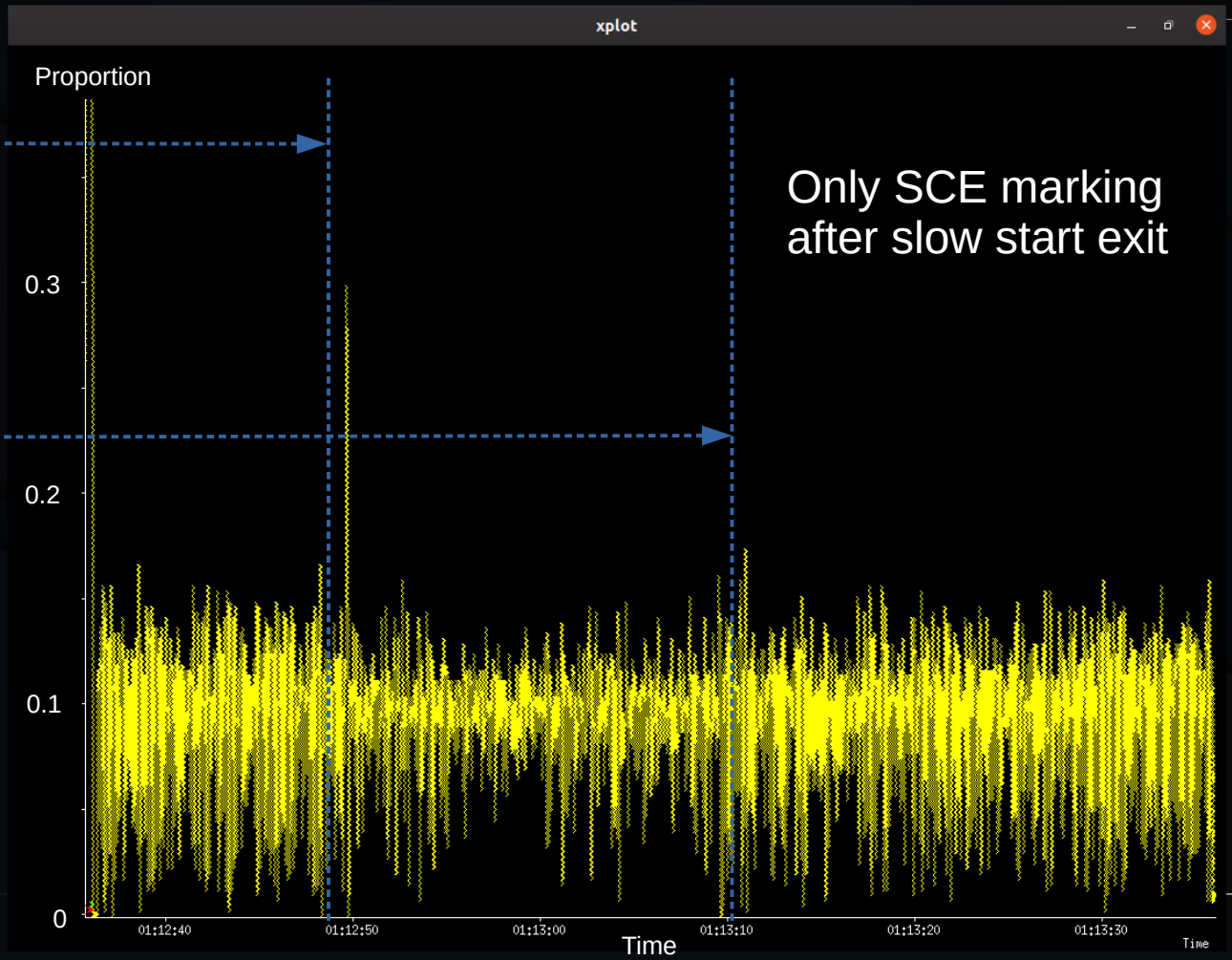


# Test Scenario: Capacity Reduction 1

Capacity reduction **50-45Mbit**

Capacity return to 50Mbit

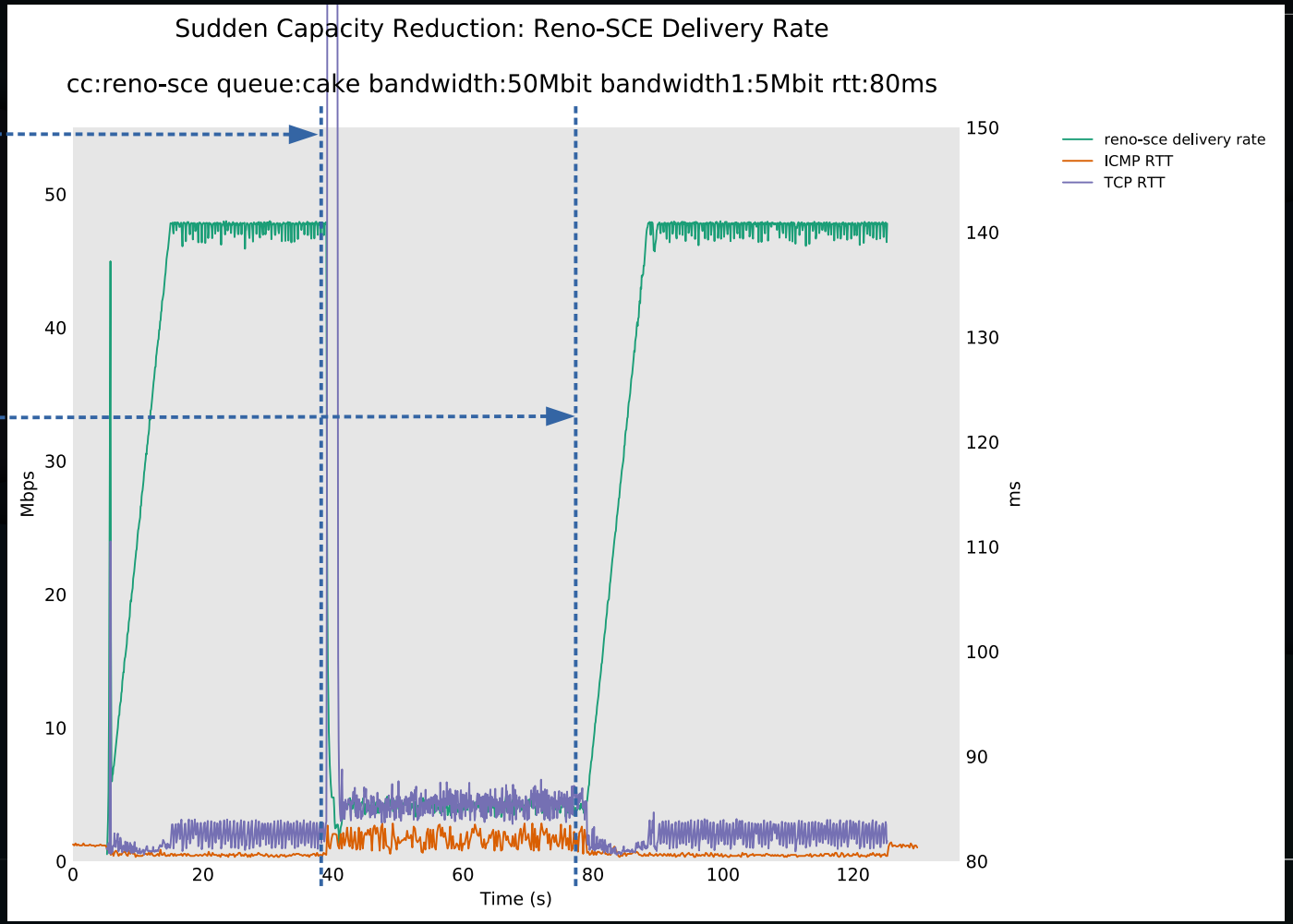
> SCE marks
> CE marks
> CWR



# Test Scenario: Capacity Reduction 2

Capacity reduction **50-5Mbit**

Capacity return to 50Mbit

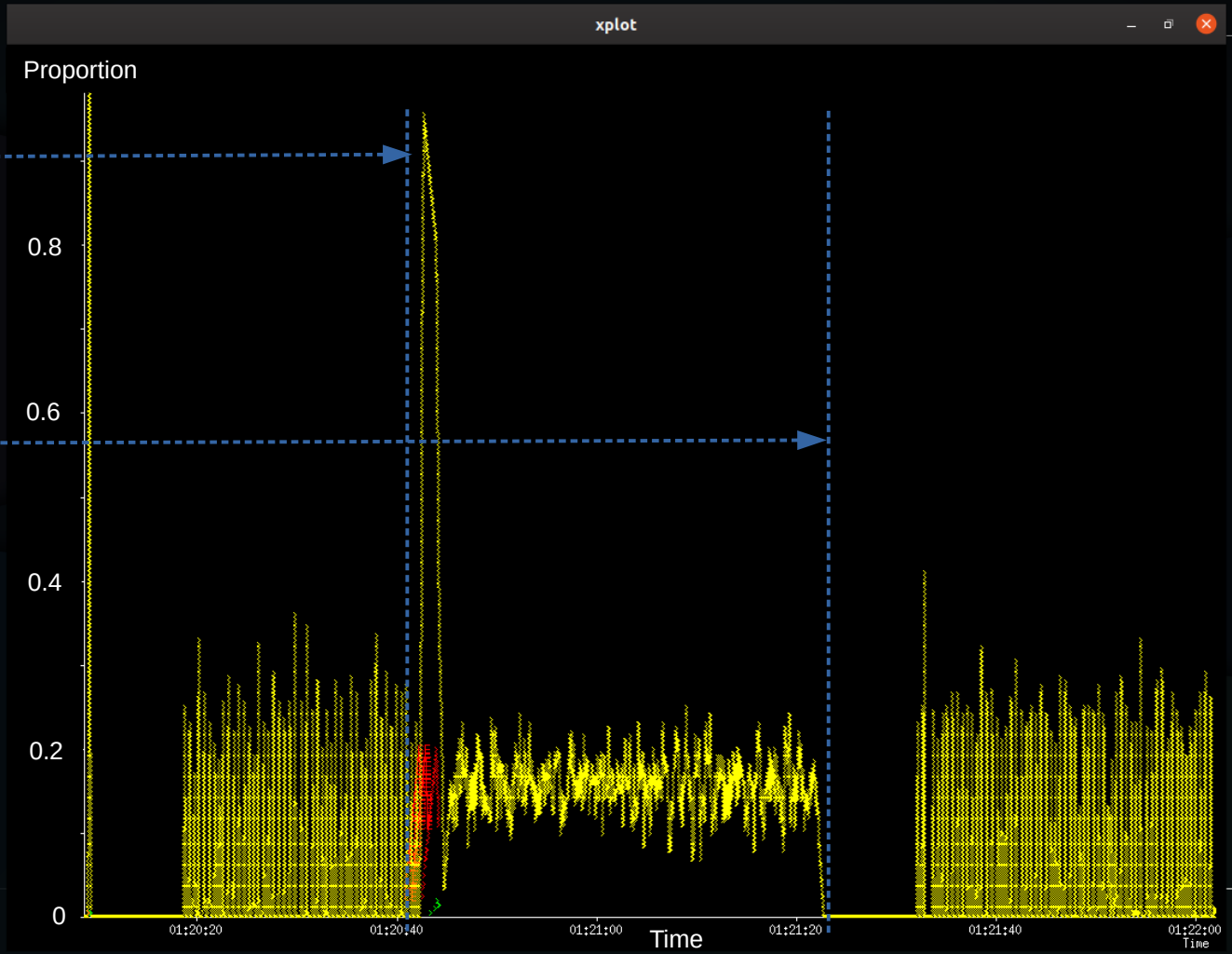


# Test Scenario: Capacity Reduction 2

Capacity reduction **50-5Mbit**

Capacity return to 50Mbit

> SCE marks
> CE marks
> CWR



# Co-existence with non-SCE Traffic

SCE needs help from the network in a single SCE AQM queue

Full FQ is good, but not required, options include:

Fair Queueing (FQ)

- Cake
- LFQ
- fq_codel

Approximate Fairness (AF)

- CNQ-CodelAF

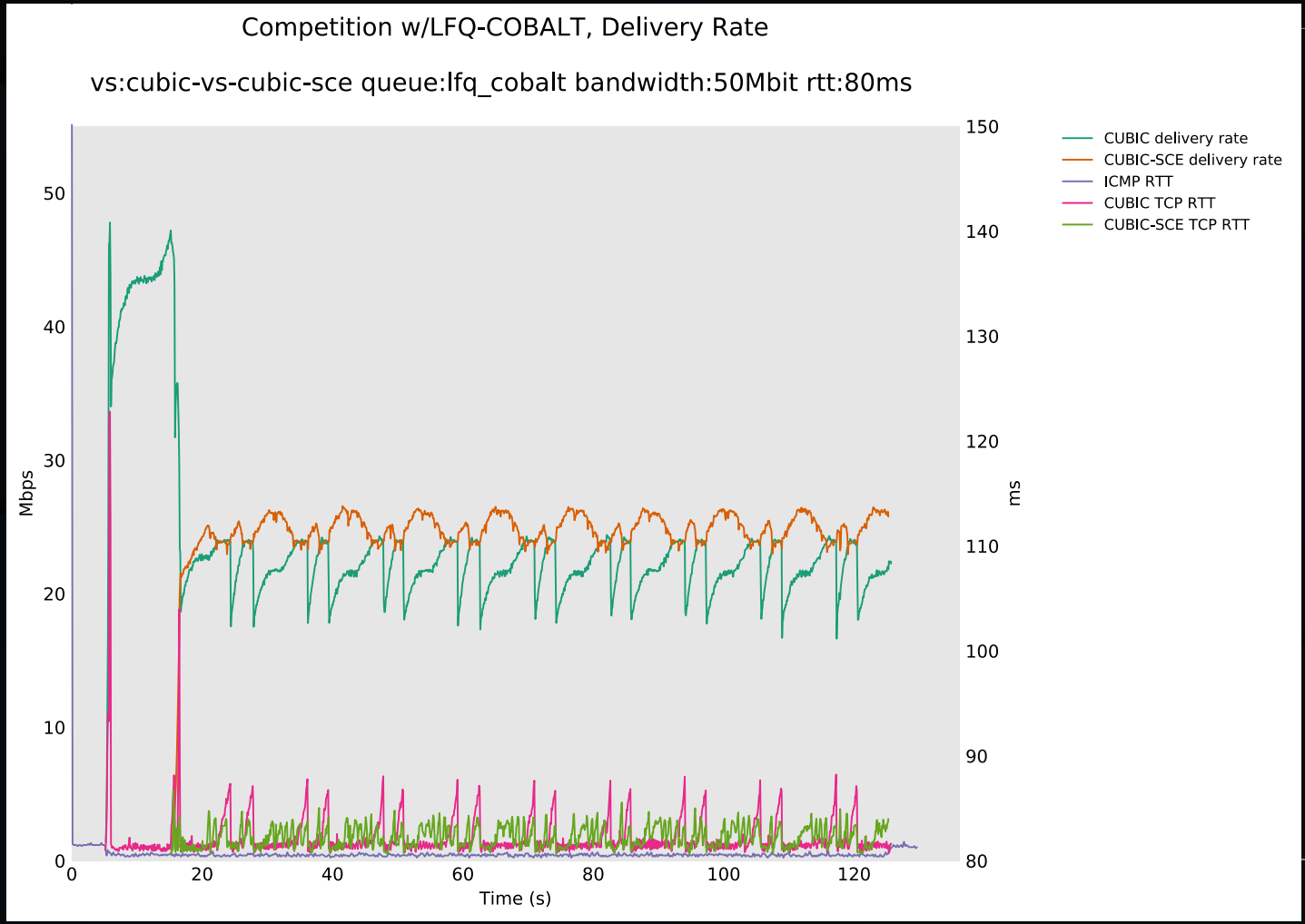
Rate-Based Congestion Control

- Ongoing research

# Lightweight Fair Queueing

- FQ for low cost or high speed hardware
- Flow fairness similar to DRR++
- Flow isolation and sparse flow optimization
- Per-flow AQM
- Only 3 FIFO queues in hardware

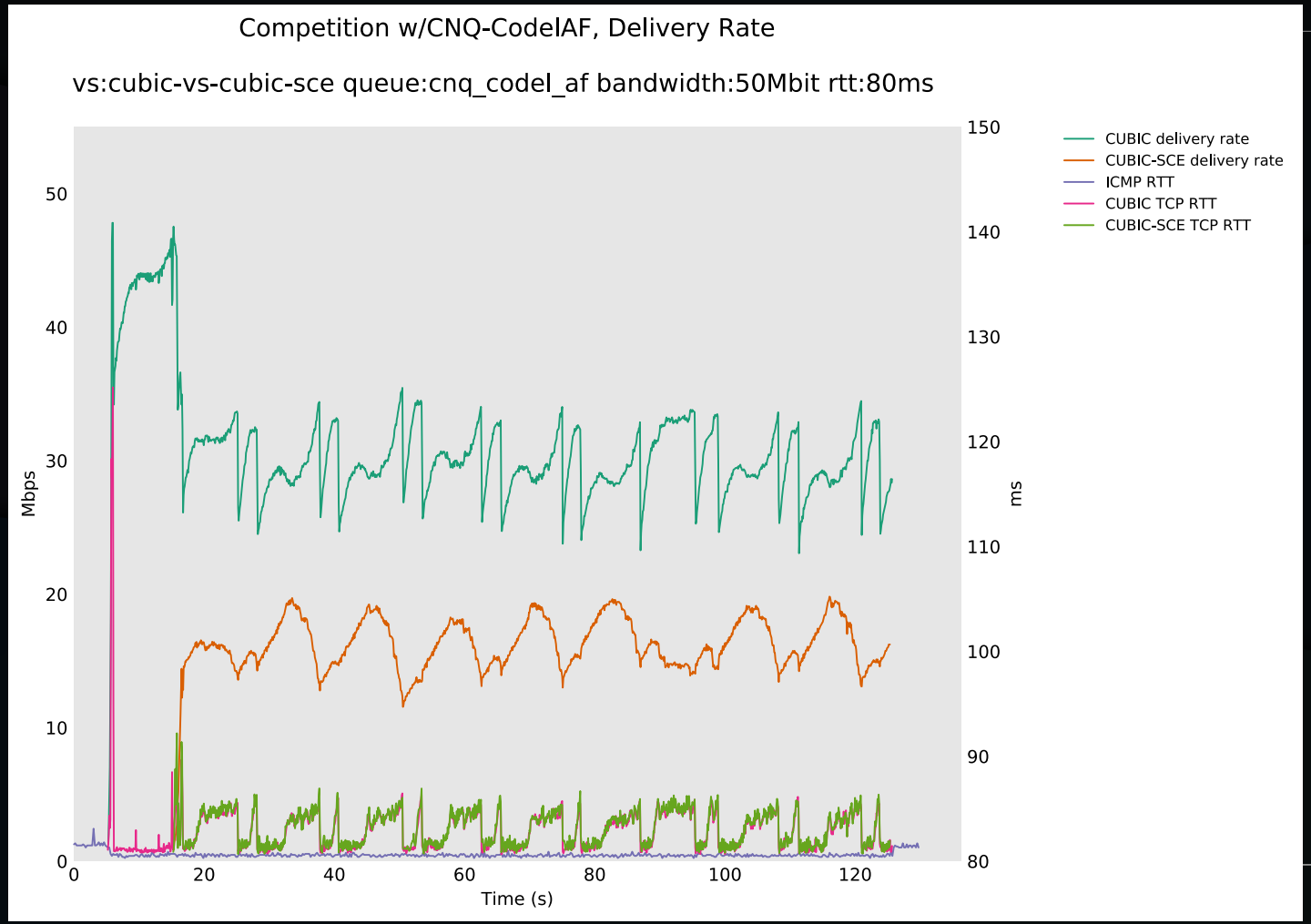
# Test Scenario: LFQ



# CNQ-CodelAF

- Approximate Fairness (AF) is an alternative to FQ
- Deployed hardware with AF exists
- Usable in a single queue (but no flow isolation)
- CNQ-CodelAF combines AF and SCE signaling:
  - Bulk queue with per-flow differential signaling
  - Sparse queue for arrival rates < bulk sojourn

# Test Scenario: CNQ-CodelAF



# SCE and the Prague Requirements

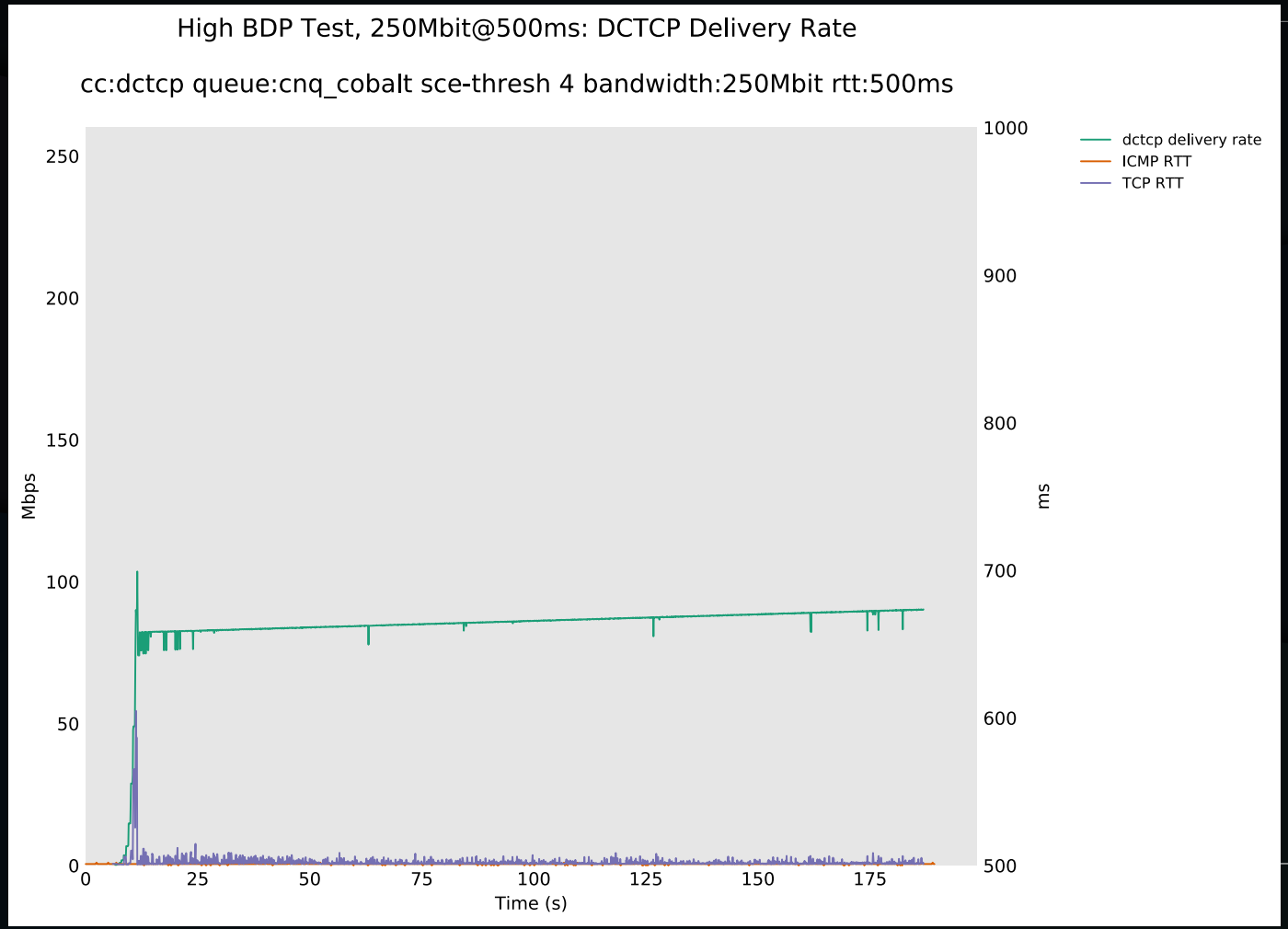
| Requirement | SCE Fulfillment |
|---|---|
| Packet identifier | Yes, ECT(0) |
| Accurate ECN feedback | Yes, ESCE feedback accurate, unreliable |
| Fallback to MD on loss | Yes |
| Fallback to MD on RFC-3168 mark | Yes, CE treated unambiguously |
| Reduce RTT dependence | Throughput inversely proportional to RTT |
| Scale down to fractional cwnd | Possible with pacing scale factors |
| Reordering tolerance on time basis | Yes, inherited from RACK |
| Scalable throughput | Yes, using CUBIC derivative |

# Difficult Environments: High BDP

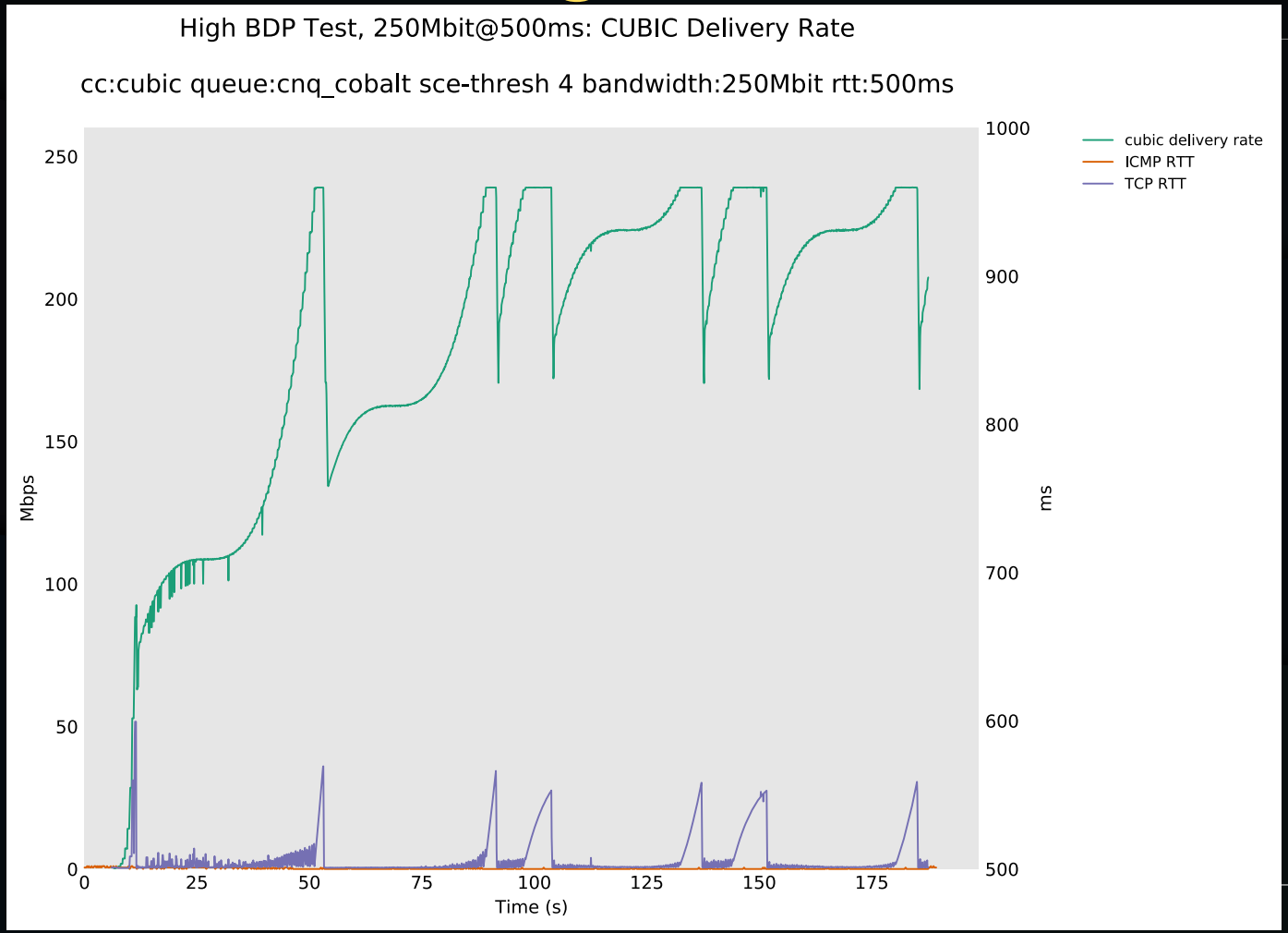
- CUBIC scales well to high BDPs
- CUBIC-SCE adds SCE response:
  - Reset polynomial growth curve
  - Apply SCE's ELR to cwnd and cubic curve:

```
reno_accum -= acked_bytes * sqrt(cwnd);
if(reno_accum <= -(cwnd * mss)) {
    reno_accum += cwnd * mss;
    cwnd--;
}
```

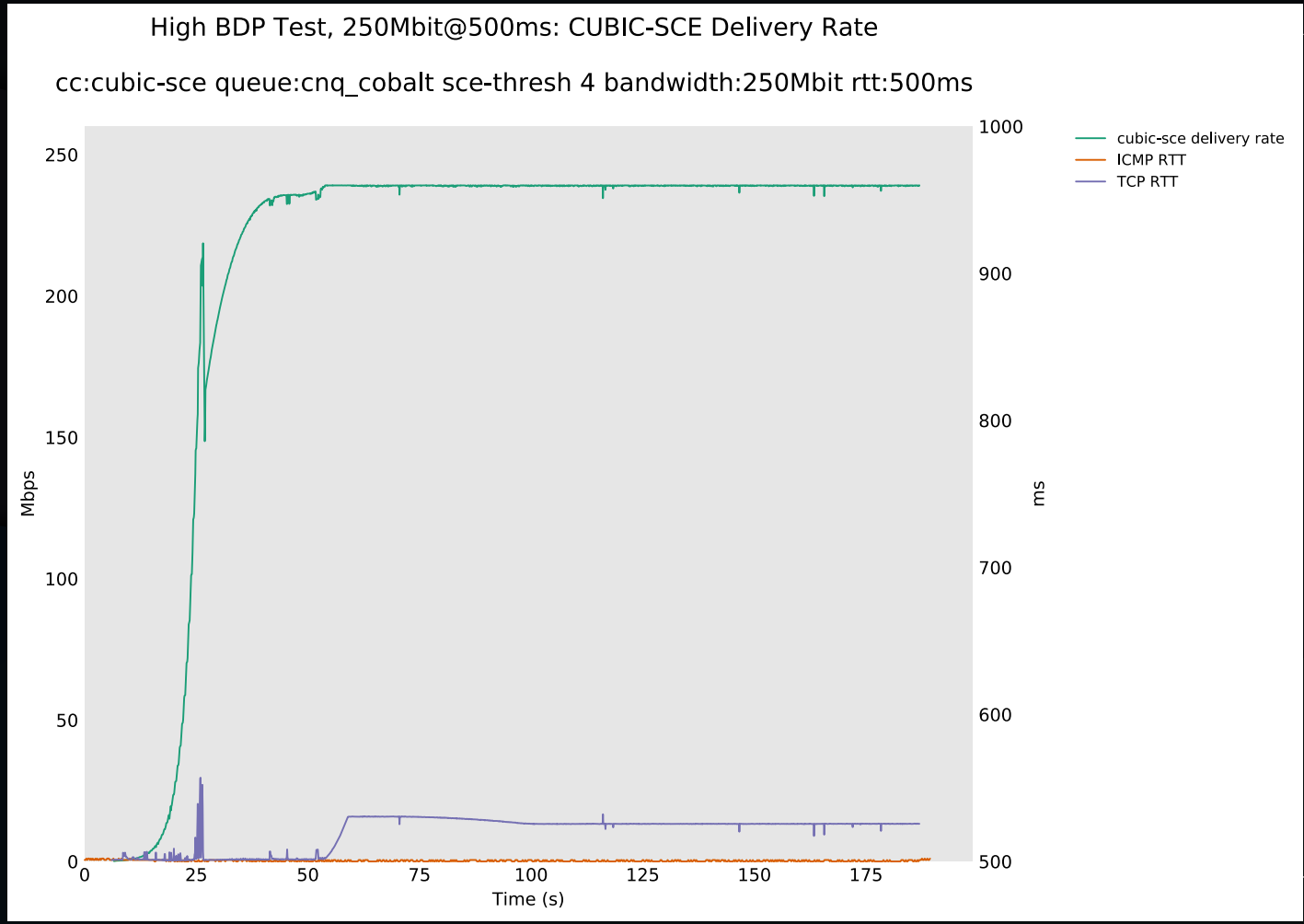
# Test Scenario: High BDP DCTCP



# Test Scenario: High BDP CUBIC



# Test Scenario: High BDP CUBIC-SCE



# Difficult Environments: Burstiness

- Burstiness a challenge for high fidelity congestion control
- SCE marking with a second CoDel instance can improve this, as shown at Singapore Hackathon
- Research ongoing

# SCE: Next Steps

- Continue towards WG adoption
- Continue RFC-5033 and other guided testing
- Research ways of meeting Prague requirement #5 (reduce RTT dependence)

# Some Congestion Experienced

Any questions?