BBR and LEDBAT++

Alberto García-Martínez and Marcelo Bagnulo

IETF 115 - London

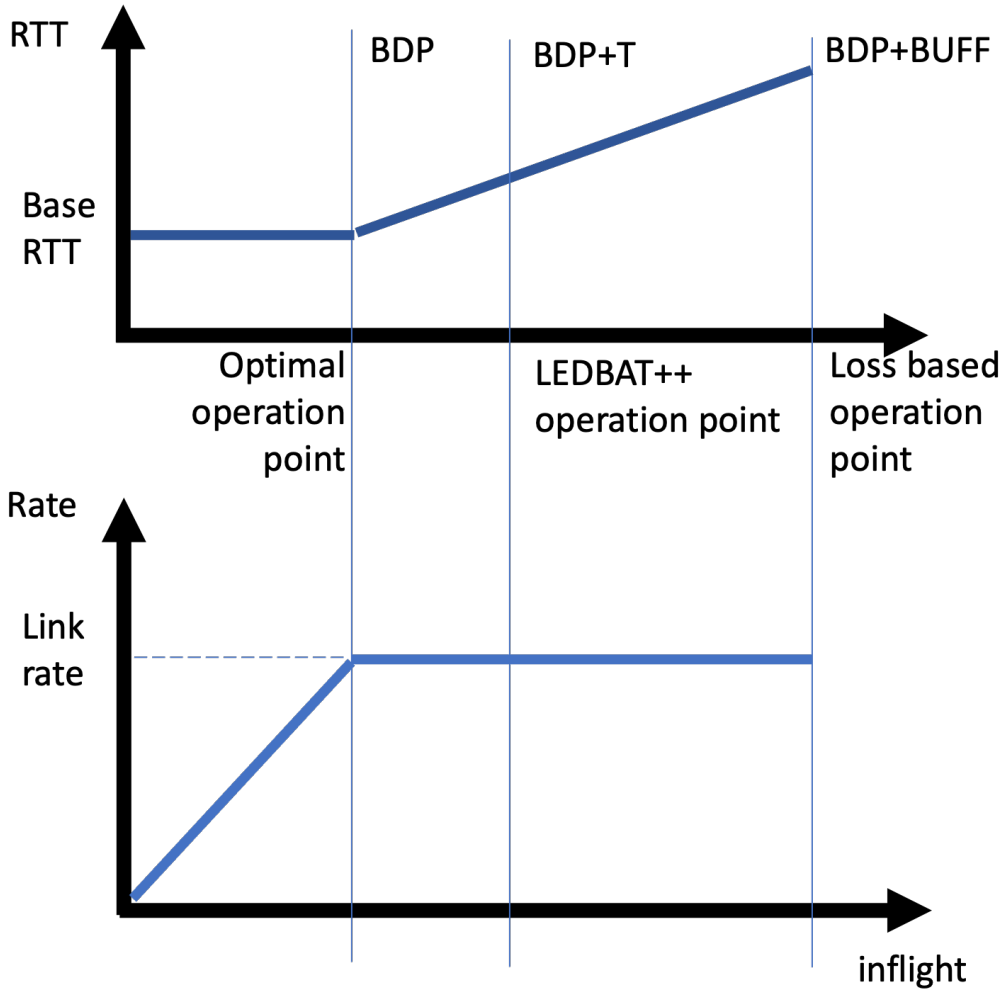
Operation points

LEDBAT++ is LBE transport targetting a queue of 60 ms

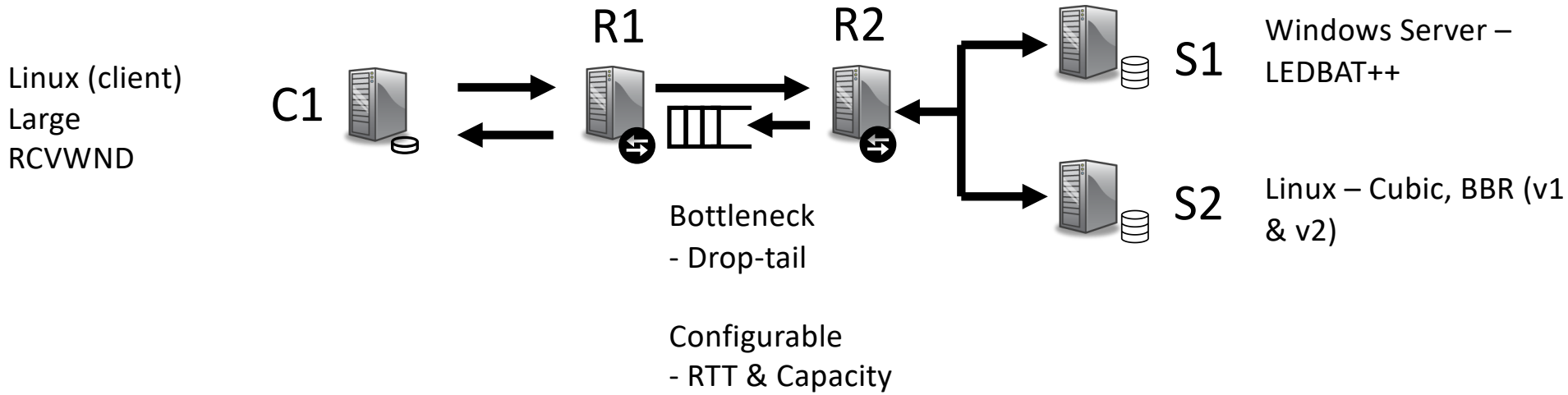
BBR is a BE transport aiming to the optimal operation point

Expressing buffer size in terms of the max queueing delay.

Assuming Buffer>T



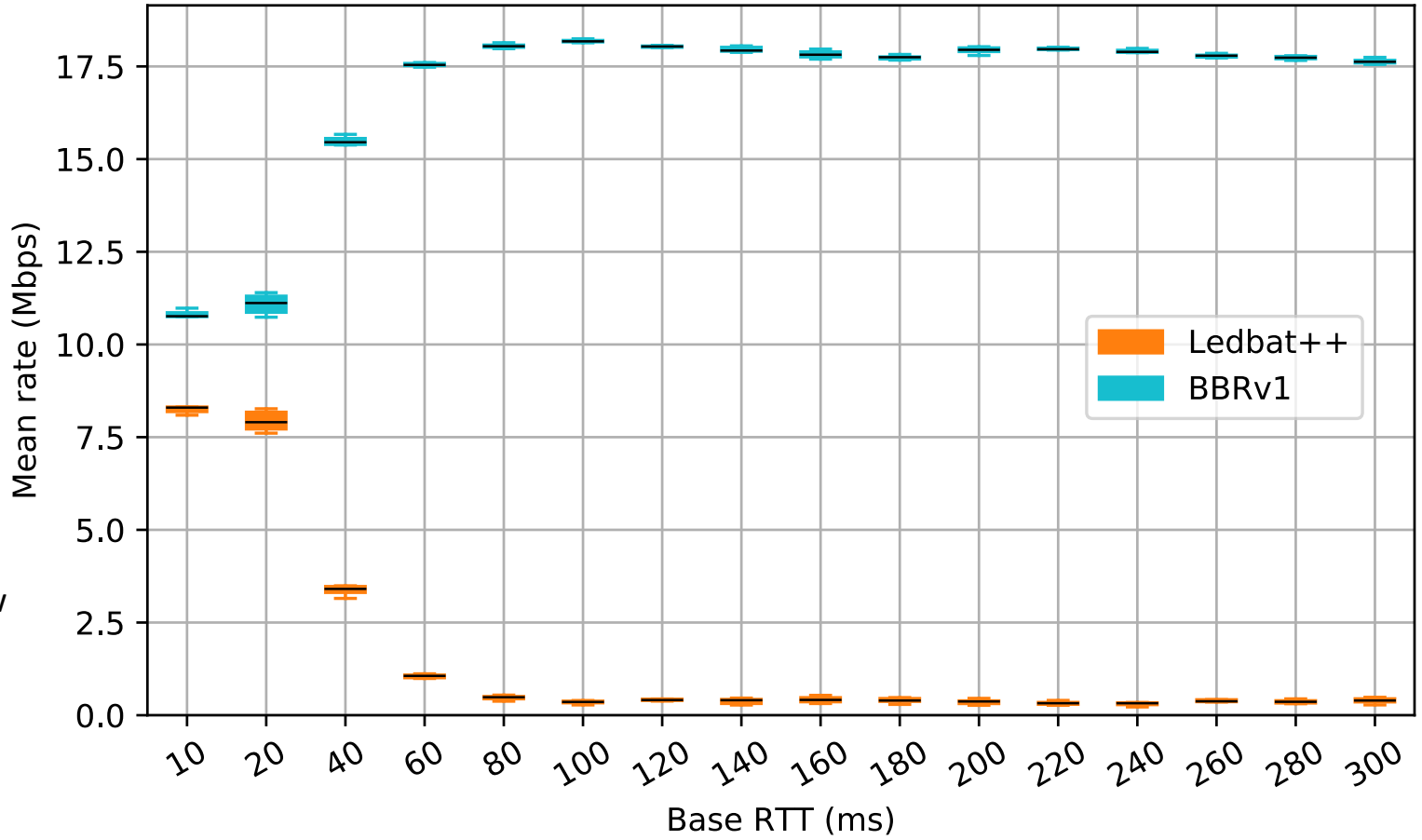
Experimental setup



- If Buffer>T, LEDBAT++ reacts to queuing delays
- If Buffer<T, LEDBAT++ reacts to losses

BBRv1 vs LEDBAT++ - delay based

BBRv1 Flow throughput limited by BBRv1 flightsize cap, set to 2 BDP



1 BBRv1 Flow vs 1 LEDBAT++ Flow

Buff = 500 ms

Bottleneck 20 Mbps

Variable base RTT

BBRv1 vs LEDBAT++ - delay based

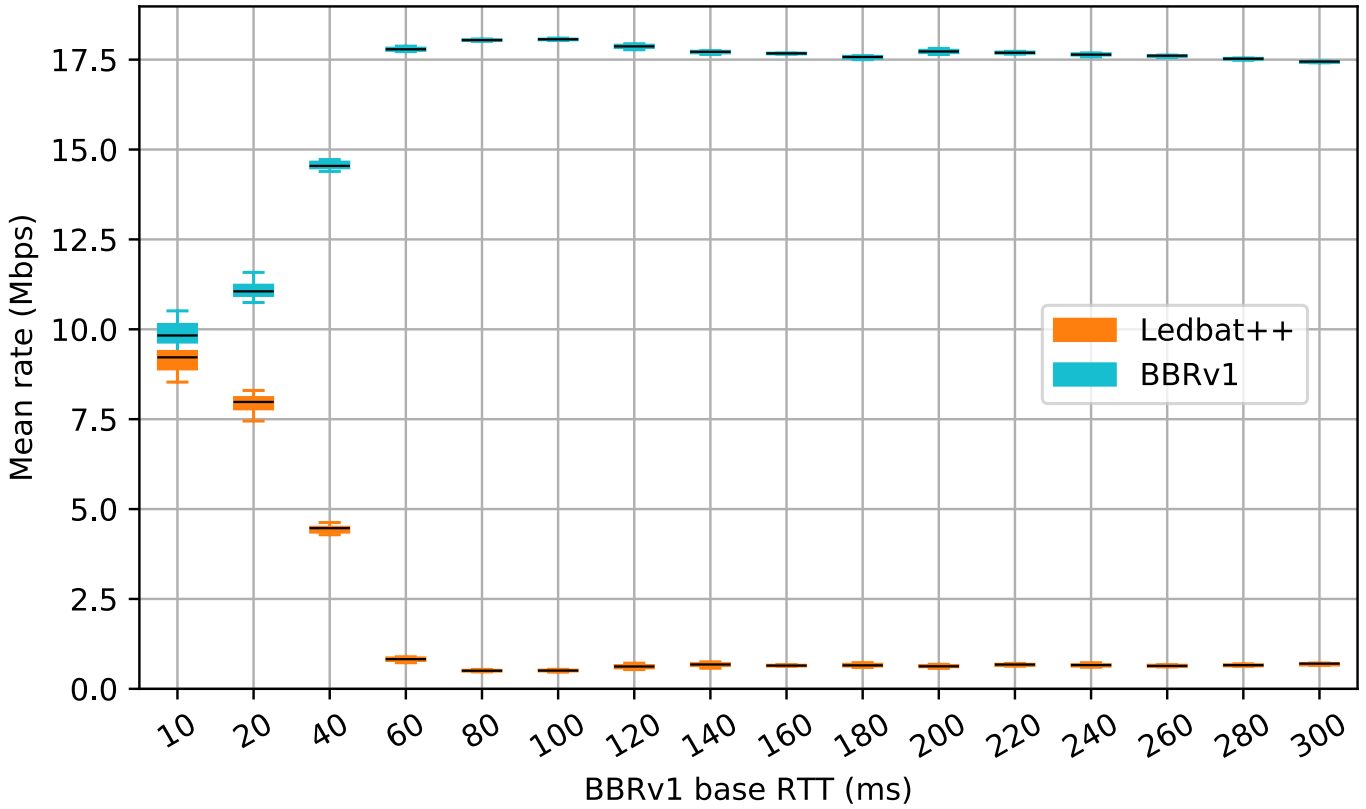
Similar behaviour if LEDBAT++ base RTT set to 20 ms and variable base RTT for BBRv1 flow

1 BBRv1 Flow vs 1 LEDBAT++ Flow

Buff = 500 ms

Bottleneck 20 Mbps

Variable base RTT for the BBRv1 flow, base RTT for LEDBAT++ Flow set to 20 ms



BBRv1 vs LEDBAT++ - delay based

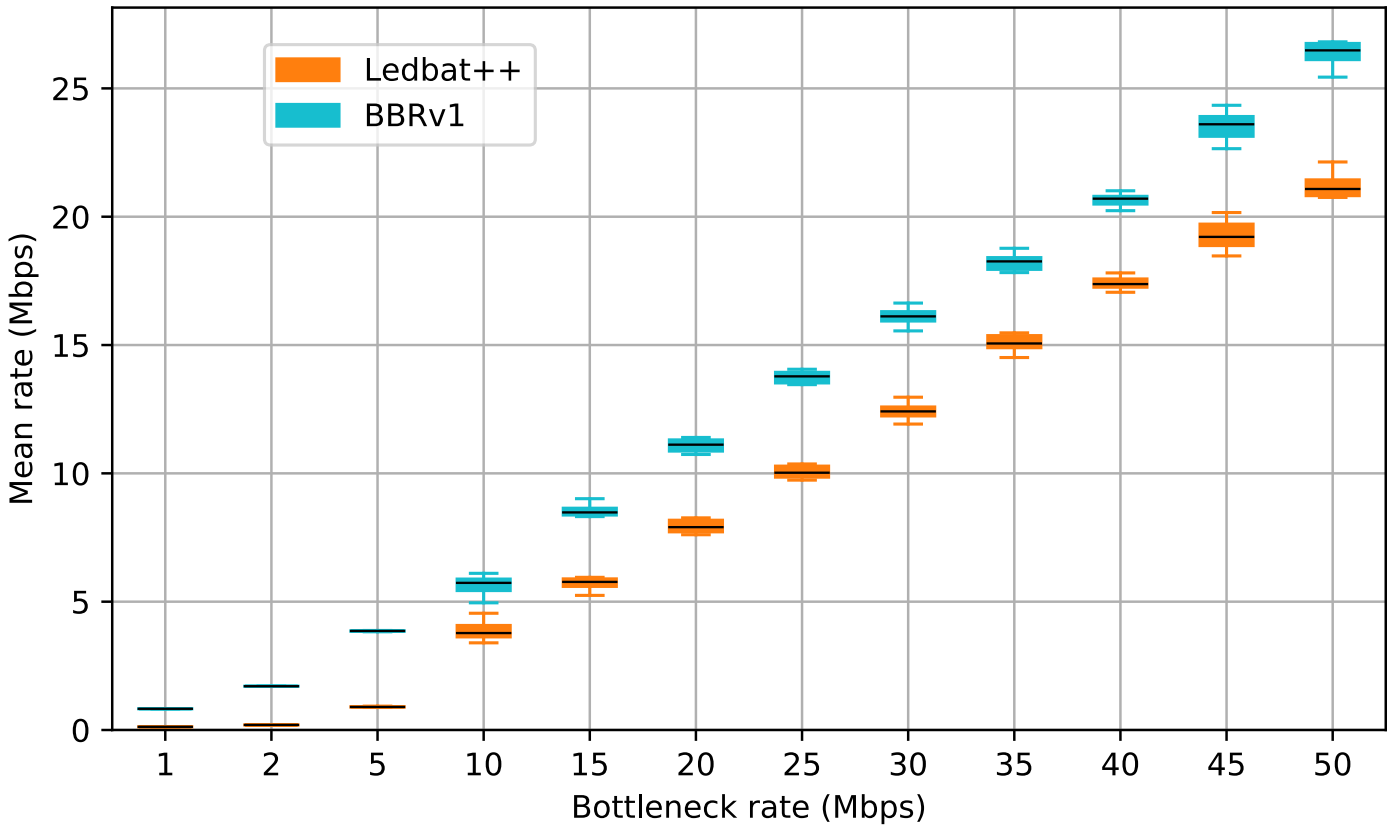
Similar Split irrespectively of the capacity

1 BBRv1 Flow vs 1 LEDBAT++ Flow

Buff = 500 ms

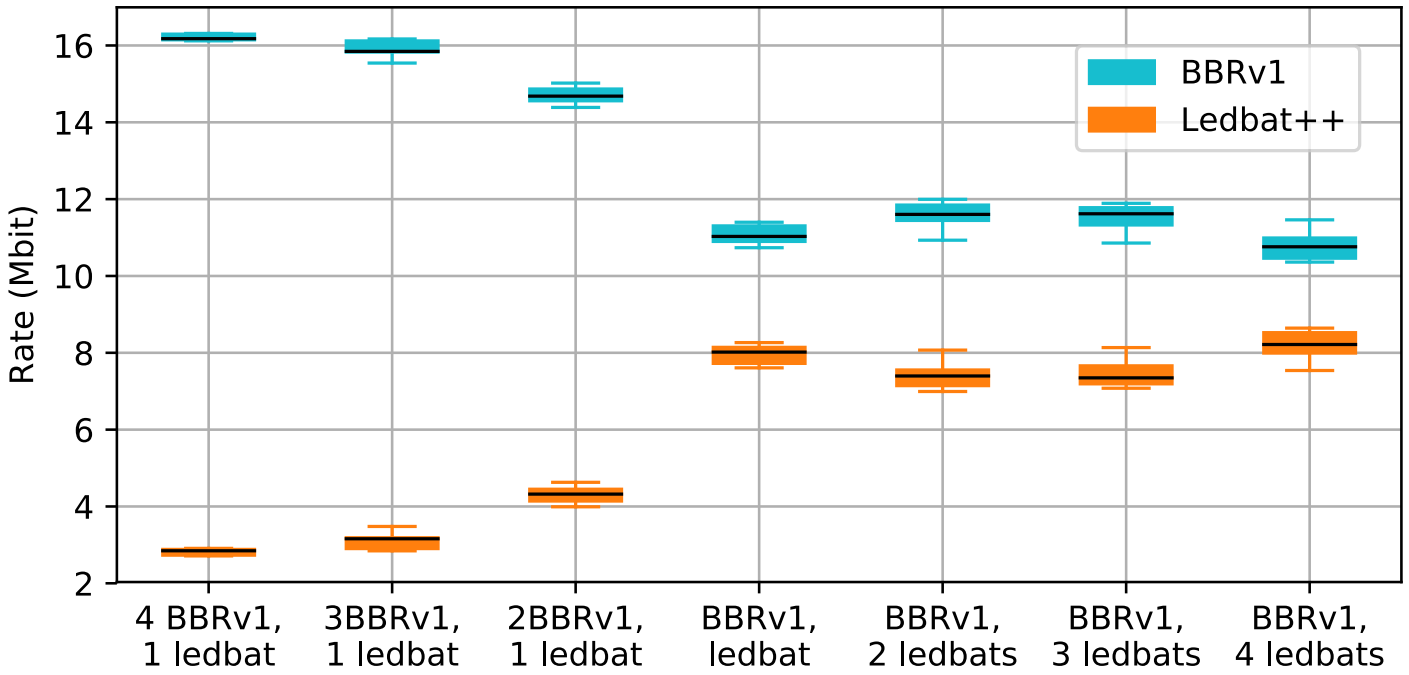
Variable capacity

base RTT 20 ms



BBRv1 vs LEDBAT++ - delay based

BBRv1/LEDBAT++ split determined by number of BBRv1 flows

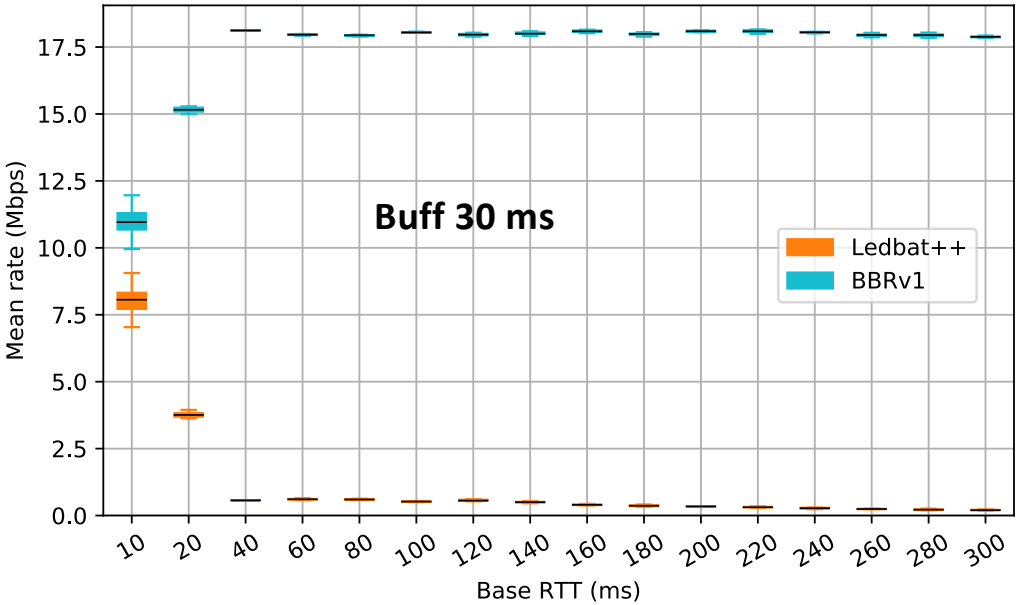


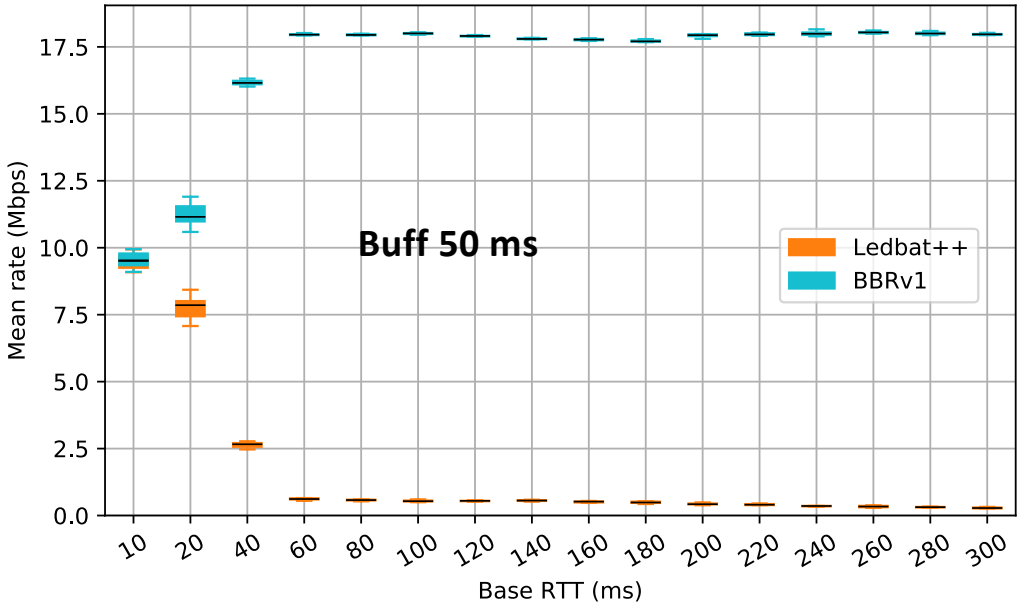
Buff = 500 ms

Bottleneck 20 Mbps

base RTT 20 ms

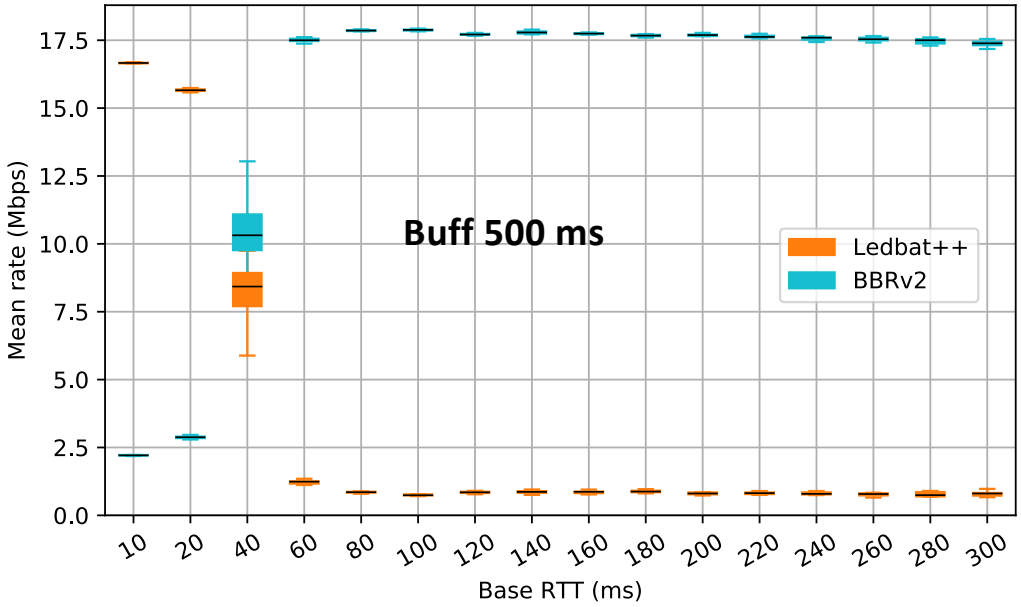
BBRv1 vs LEDBAT++ - Loss based

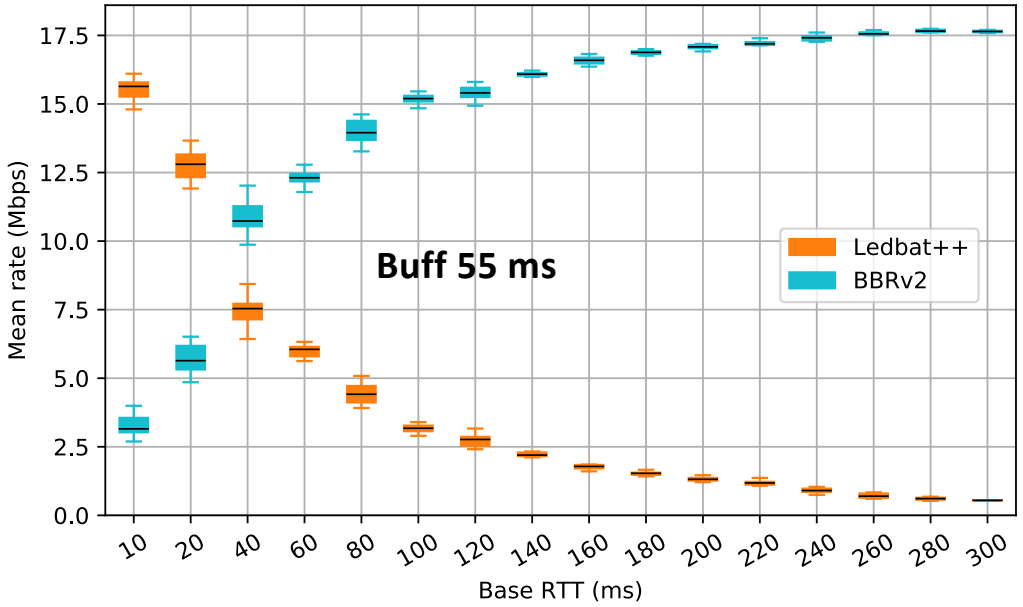




1 BBRv1 Flow vs 1 LEDBAT++ Flow
Bottleneck 20 Mbps
Variable base RTT

BBRv2 vs LEDBAT++

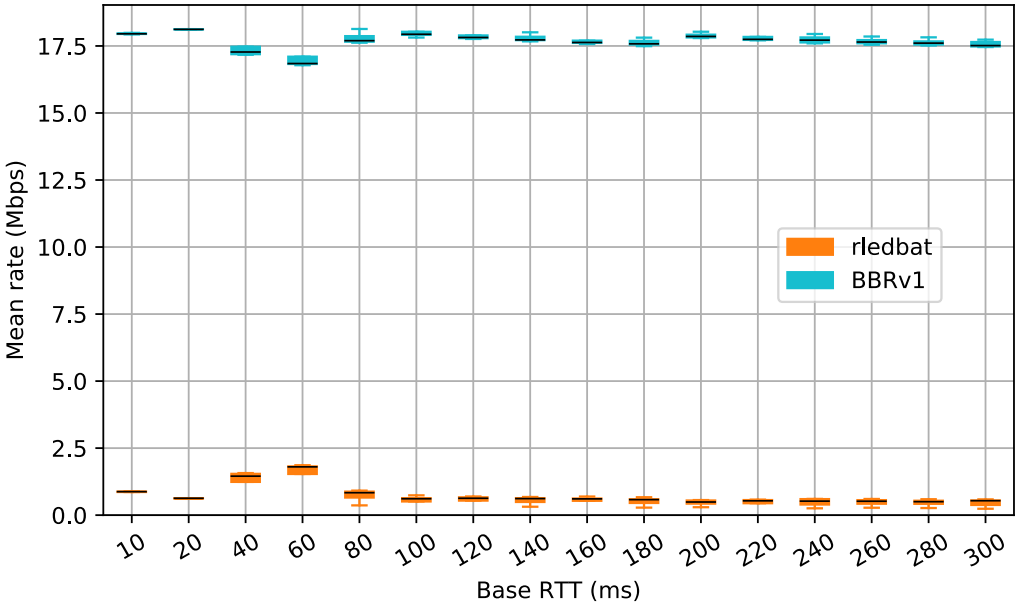


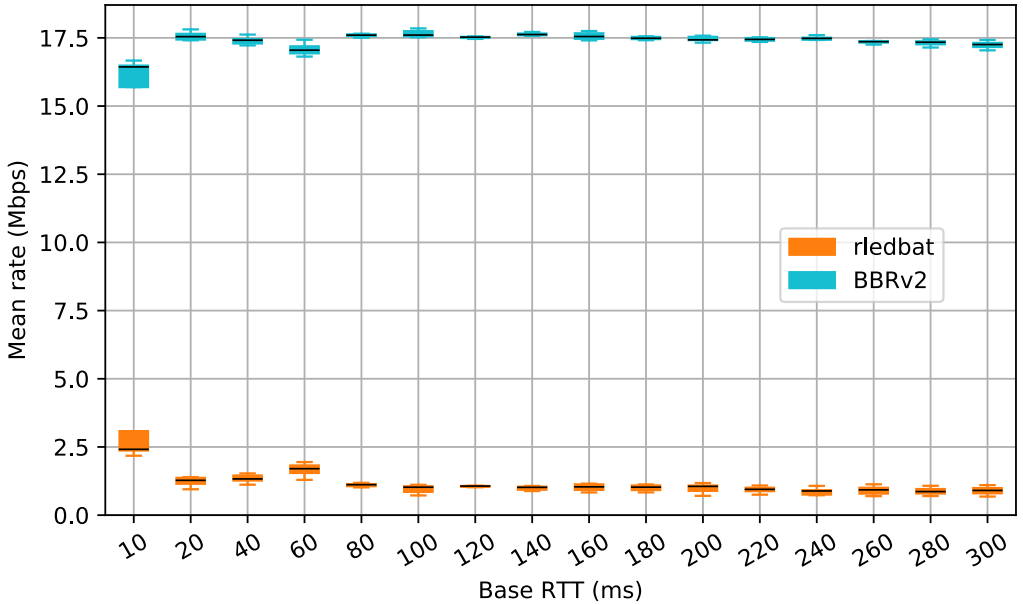


1 BBRv1 Flow vs 1 LEDBAT++ Flow
Bottleneck 20 Mbps
Variable base RTT

One possible solution modifying LEDBAT++

- Set LEDBAT++ target to min(60ms, base RTT)





1 BBR Flow vs 1 LEDBAT++ Flow
Bottleneck 20 Mbps
Variable base RTT

Measuring the base RTT: Periodic slow downs

- Both BBR and LEDBAT++ perform periodic slow downs to empty buffers and accurately measure the base RTT
- LEDBAT++ periodic slow down period:
  - 9 times the time that it takes for the slow start to ramp back up (i.e depends on the CWND size)
  - Lasts 2 RTTs
- BBR periodic slow down period:
  - 10 seconds (since a min base RTT value was observed)
  - Lasts 200 ms and at least 1 RTT.

Experiment

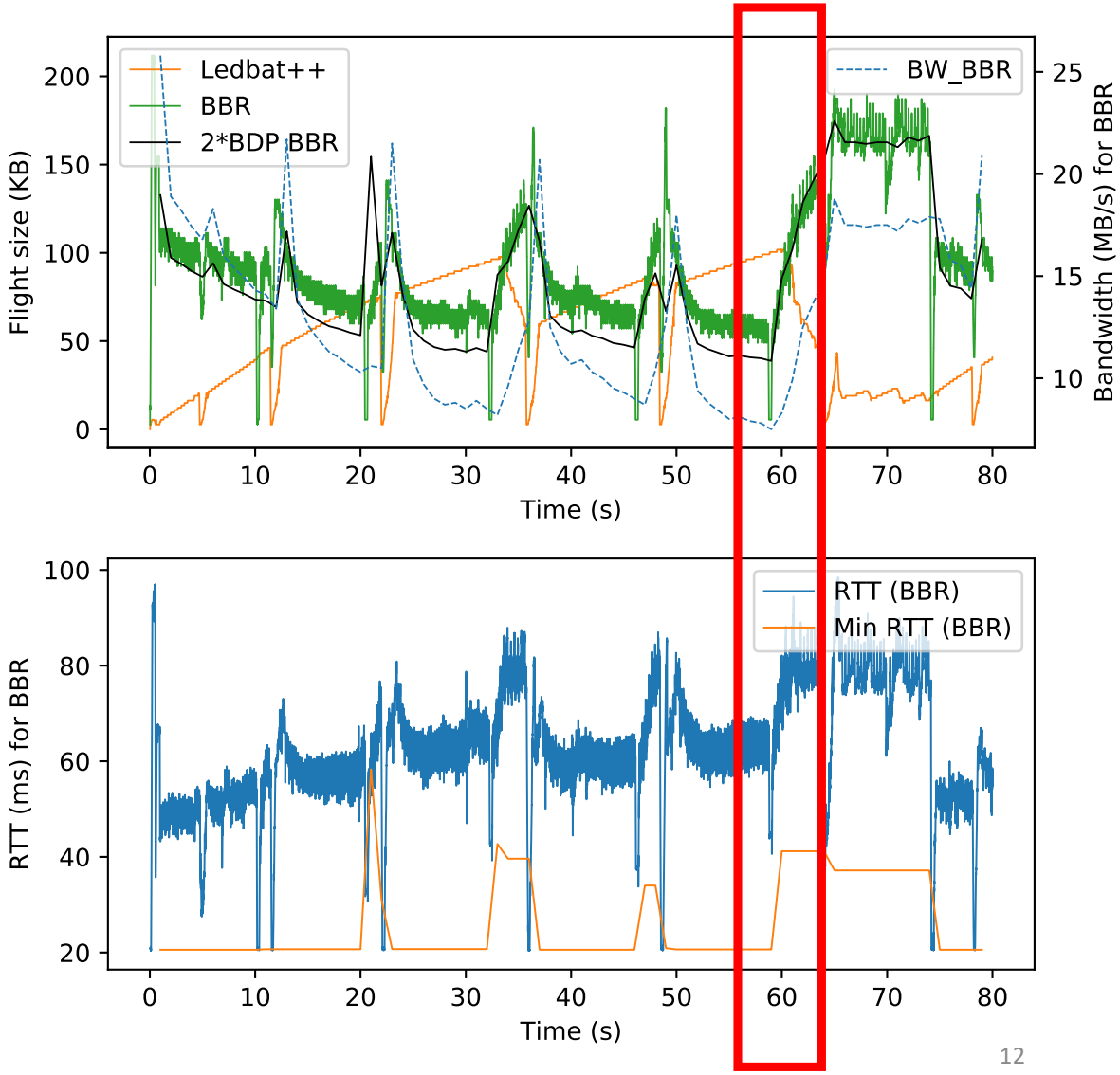
Un-synchronized slow downs prevent accurate measurements of the base RTT

1 BBRv1 Flow vs. 1 LEDBAT++ Flow

Buff = 500 ms

Bottleneck 20 Mbps

base RTT 20 ms



Possible solution

- Use the same slow down mechanism in both LEDBAT++ and BBR
- LEDBAT++ periodic slow downs exhibits other shortcomings:
  - Period depends on CWND
  - AIMD is RTT biased
    - LEDBAT++ flows with smaller RTT get more capacity
  - Multiple LEDBAT++ with DIFFERENT RTTs competing, obtain different capacity
    - Slow downs will not be synchronized, prevent emptying buffers and accurate measurements of the base RTT
    - Havent tested it experimentally (yet)
- Possible solution: update LEDBAT++ to adopt BBR's periodic slow down approach

Congestion Control Algorithm Invariants

- In order to be able to accuratelly measure the base RTT, competing flows using different congestion control algorithms should avoid interfere with each other
- Should we reccomend that all congestion control algorithms should implement the same periodic slow down mechanism?
  - Or at least implement the means to avoid interference?
- Are there other invariants that all congestion control algorithms should implement to avoid interfering with each other?

More information

M. Bagnulo & A. García-Martínez, When less is more: BBR versus LEDBAT++, submitted to Computer Networks journal.

Temporarilly available at:
https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4200007