

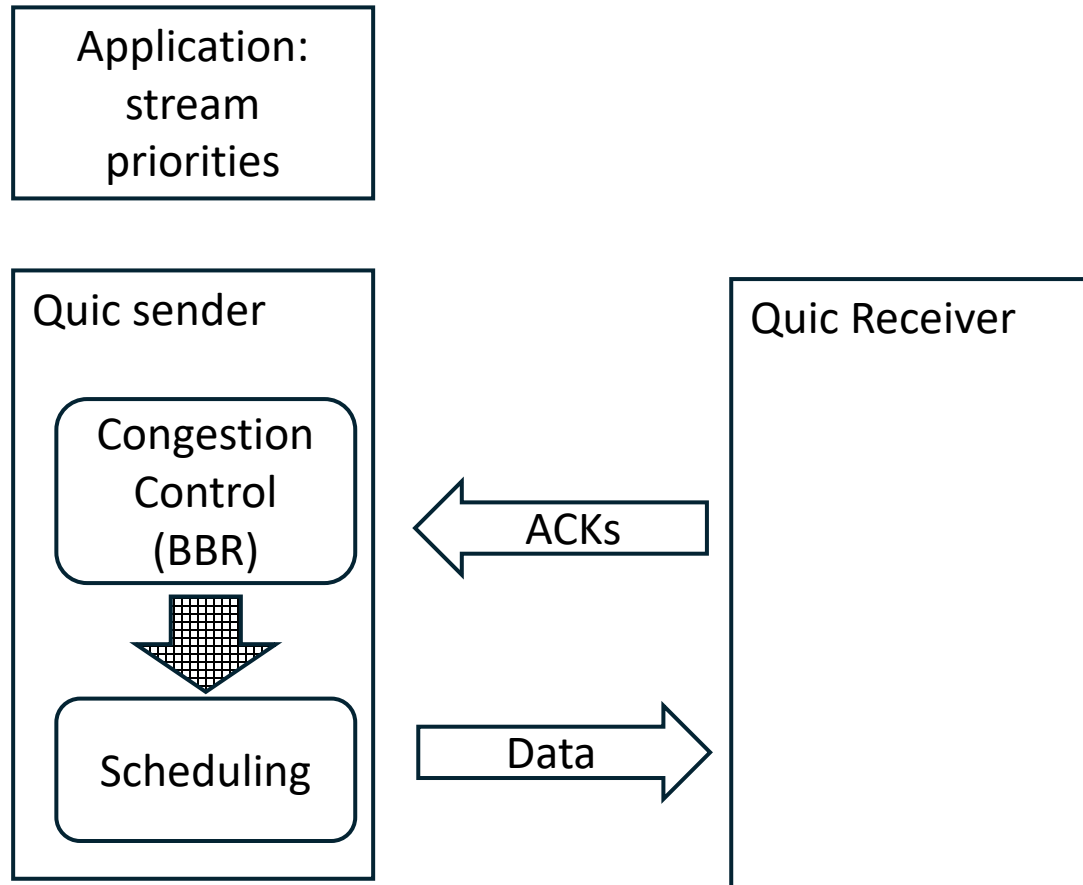
BBR Improvements for Real-Time connections

Christian Huitema, Suhas Nandakumar, Cullen Jennings

draft-huitema-ccwg-bbr-realtime-latest

IETF 120, Vancouver, July 2024

What we want to do: real time scheduling

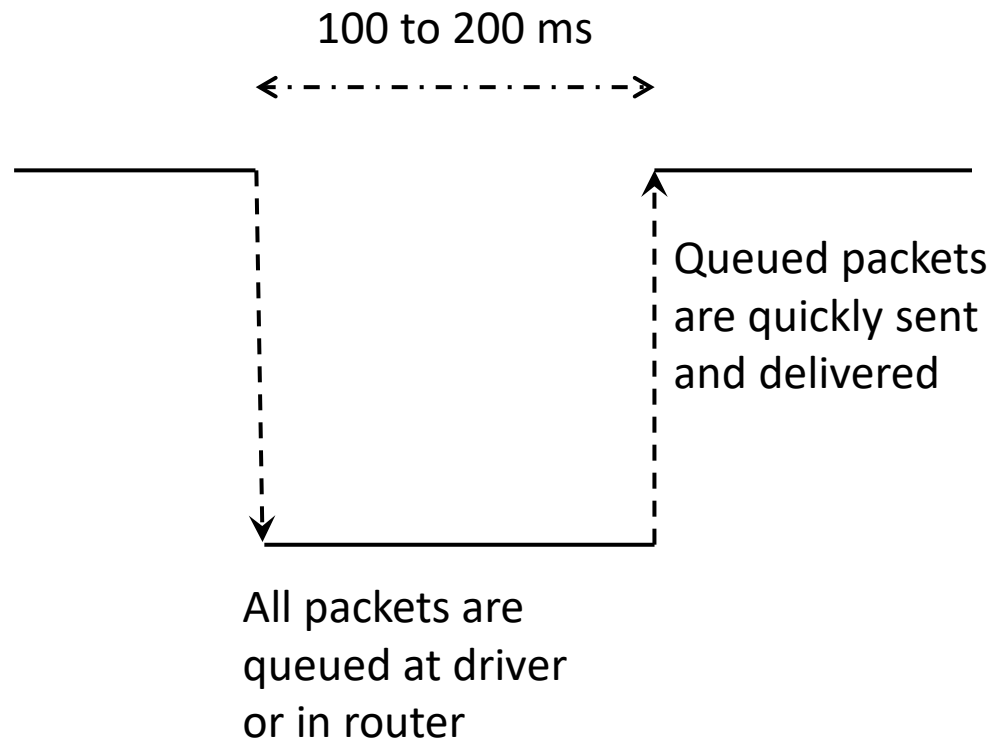


- Don't build queues!
 - Use BBR rather than Cubic, Reno
 - Target latency 100-200ms
- Schedule “most important” first:
 - Per stream priorities
 - Per network capacity
- Network capacity from BBR
 - In theory, if capacity changes, adapt quickly
- In practice...

A few issues

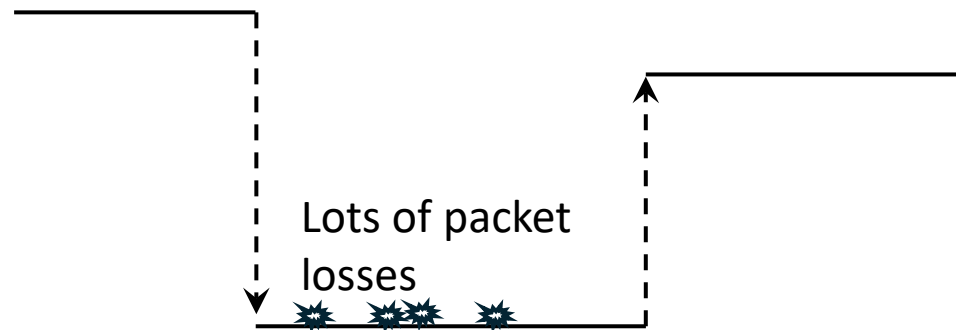
- BBR startup creates $2 * RTT$ queues (wait 3 rounds to exit)
- Poor early estimate impact connection for a long time
 - E.g., when application transitions from “app limited” to “burst”
- Wifi Suspensions are hard to handle (see next slide)
- Wifi can go bad quickly (see next slide)
- Downward data rate drift for app-limited applications
- App limited application may linger in Probe BW state

Wi-Fi suspension, because driver says so



- Suspension is undetected for 1 PTO
 - queues are building up with a mix of data of various priority levels
- Additional RTO events happen
 - BBR driven to “low bandwidth” state
- Priority inversion at end of suspension
 - Low priority packets delivered by Wi-Fi driver, router
- ACKs received quickly at end of suspension
 - Bandwidth ramps up progressively
- Retrieve normal rate “slowly”

Wi-Fi goes bad, because something moved



Bandwidth drops
from $N \cdot 100$
Mbps to a few
Mbps

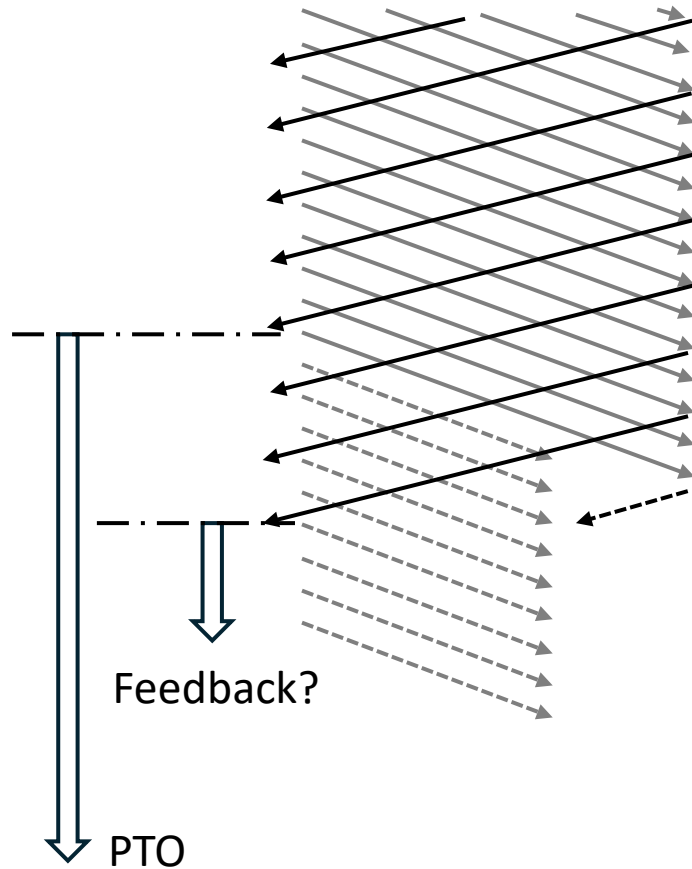
Bandwidth
"restored" after
several seconds,
Maybe to new
value.

- Before PTO, lots of queue buildup (e.g., in front of driver)
 - BBR sending at "normal" rate
 - Applying priorities to retransmissions helps!
- BBR adapts to lower bit rate
 - Maybe a bit slowly
- Very slow ramp up
 - BBR3 specifies long wait until ProbeBW-UP

Lots of proposed fixes (see draft)

- Implement an "early exit" from startup.
 - Make BBR startup more like HyStart++
 - Add an option for rapid start of ProbeBW-Up,
- Exiting Probe BW UP on delay increase
 - Avoid building queues!
- Add explicit handling of "suspension" to BBR,
 - Do something like "safe resume" on end of suspension
- Add detection of feedback loss (see slide)
- Entering Probe BW UP after new streams are started

Detecting the loss of feedback



- In normal condition, steady stream of ACKs
 - ACK interval predictable with “ACK Frequency” option in QUIC
- No ACK in predicted delay?
 - Early warning, before PTO
- On feedback loss event, become conservative!
 - Do not build queues!
- Back to normal if ACK arrives

Issues and solutions

- BBR startup queues, poor initial estimate
 - "early exit" from startup, rapid transition to ProbeBW Up after Drain
- Wifi Suspensions
 - Feedback loss event, handling of suspension in BBR
- Wifi can go bad
 - Feedback loss event, rapid transition to ProbeBW Up
- Downward data rate drift
 - Enter ProbeBW Up rapidly if new streams opened
- App limited linger in Probe BW
 - Exit ProbeBW up on delay increase

Next steps

- Socialize the issue, get attention from CCA developers (Done?)
 - Watch demo video:
<https://www.youtube.com/watch?v=MM2wdurKRrc>
 - Discuss applicability of Feedback Loss event
 - Discuss handling of Wi-Fi suspension
 - Prepare BBRv4 with fixes?
-
- Questions?