

Congestion control for lower latency and lower loss media transport

draft-ohanlon-rmcat-dflow-01

Piers O'Hanlon
Ken Carlberg

Updates(1): 00 01

- New sections
 - Delay Composition
 - transmission (or serialisation), propagation, processing, and queuing delays
 - Delay Measurement
 - How we measure it
 - Slow-start
 - Mentioned that delay-based approach is used there too
 - Loss-mode
 - A statement of intent

Updates(2): 00 01

- Reworked a lot of the document
 - Added refs for previous work
 - Cleared up notation
 - Various clarifications
 - Corrected typos

Background

- Most existing congestion control **is** loss-based
 - Results in full queues => **high delay and loss**
 - E.g. TCP{new/Reno, Cubic}, TFRC, SCTP, TFWC
- Some delay-based schemes – But **most** don't aim to *minimise* delay
 - E.g. CARD, Tri-S, Vegas, CTCP (partial), CxTCP, LEDBAT
- A few do now (but mostly unpublished)
 - Ghanbari fuzzy logic, Google RRTCC

DFlow: Objectives

- **Lower Delay:** Needed in today's bufferbloomed Net
 - Should stay below 150ms [ITU.G114] (not exceed 400ms)
- **Lower Loss:** Loss is bad for media (retransmit tricky)
 - Low delay usually implies low loss as queues not full
- **Smoothness:** Codec output generally smooth
 - Within constraints of: media, codec, and network path.
- **Fairness:** Should aim to be reasonably fair
 - Initially we aim for self fairness and we aim to tackle TCP fairness in later rev.

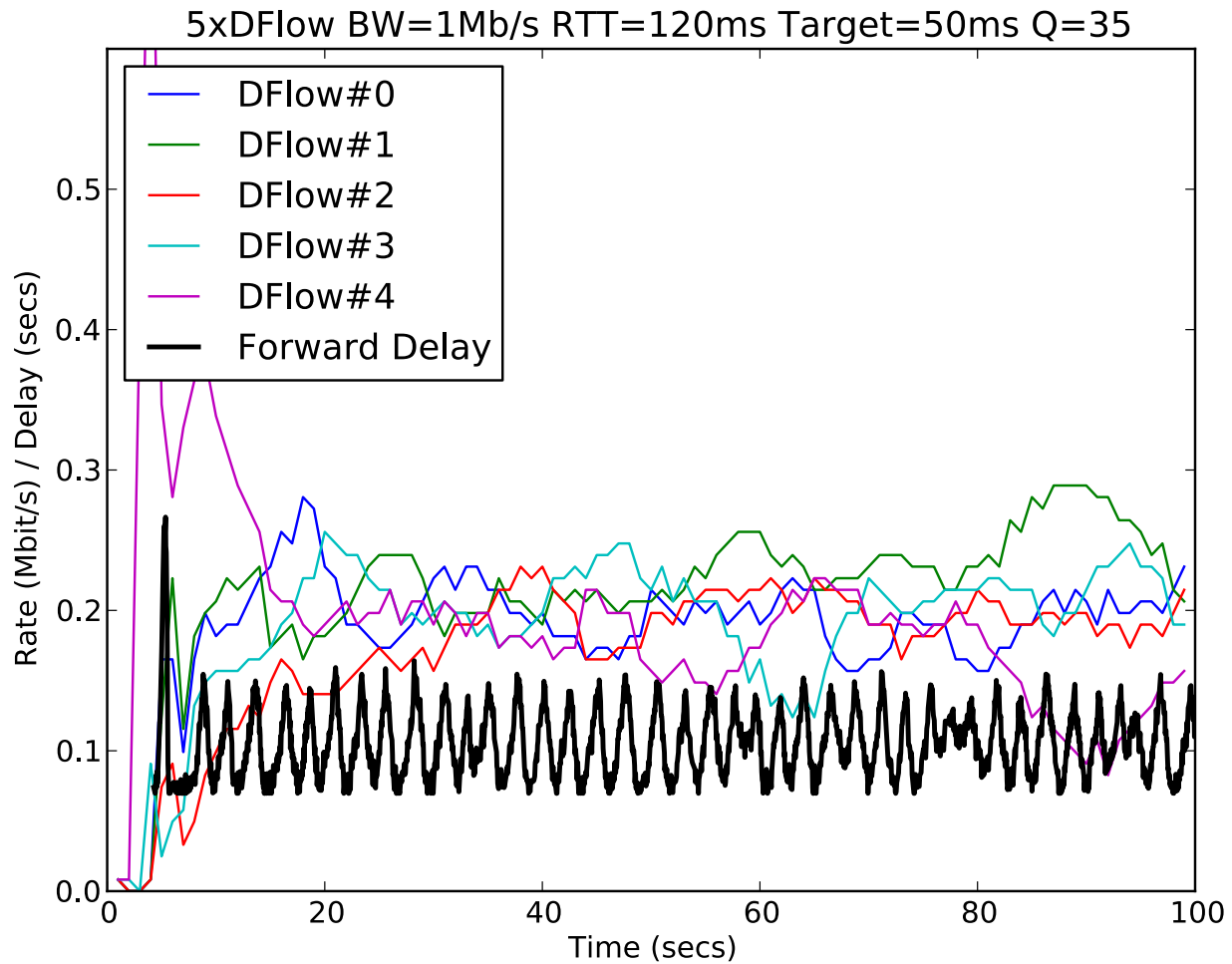
Objectives (Planned)

- [**Burst Management**]: Mechanisms to handle the bursty nature of media
 - E.g. Allow bursts when conditions permit
 - Providing for smoother quality
- [**Loss-based mode**]: Mechanisms to allow for 'fair' throughput against loss-based CC flows
 - Without additional network support (e.g. Codel, PIE) delay & loss would be largely beyond control.

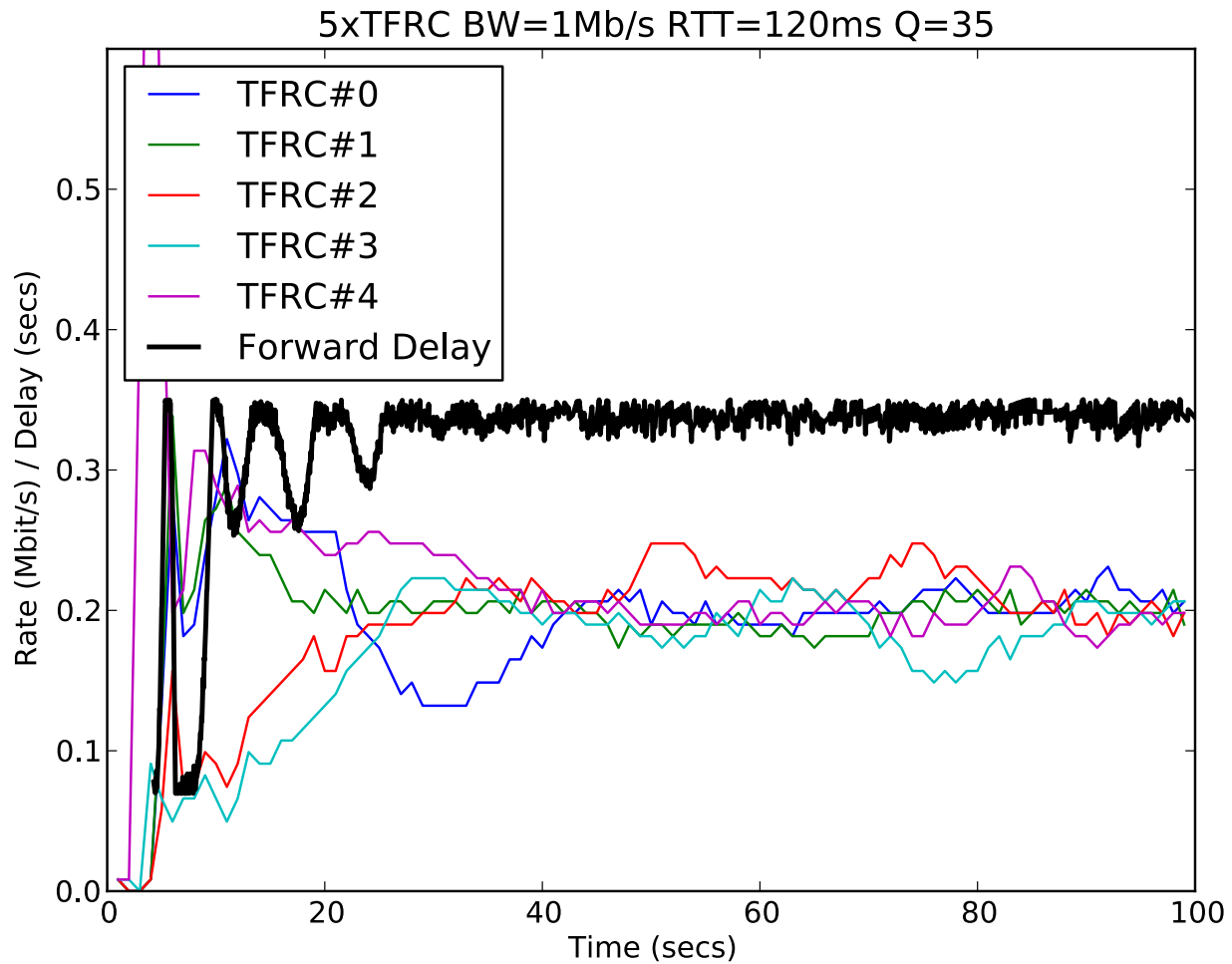
Design Outline

- Loosely based on TFRC design
 - Rate-based, TCP equation, RTT smoothing, and ‘loss’ event rate smoothing
- Uses TCP equation to derive an operating rate
- Utilises ‘delay losses’
 - Based on relative delay and its derivative
- Employs ‘congestion event history’
 - Based on TFRC ‘loss event history’ mechanism

Simulation: DFlow



Simulation: TFRC



Discussion

- This is work in progress and we're seeking feedback/comments
- Work on planned objectives
 - Loss-mode for competition with TFRC/TCP
 - Burst management
- More simulations and testing