Managing Real-Time Media Flows through a Flow State Exchange

Safiqul Islam, *Michael Welzl, David Hayes, Stein Gjessing*

Networks and Distributed Systems Group

Department of Informatics

University of Oslo

Outline

- Context
- Problem Statement
- Example Architecture
- Flow State Exchange (FSE)
- Results
- Summary

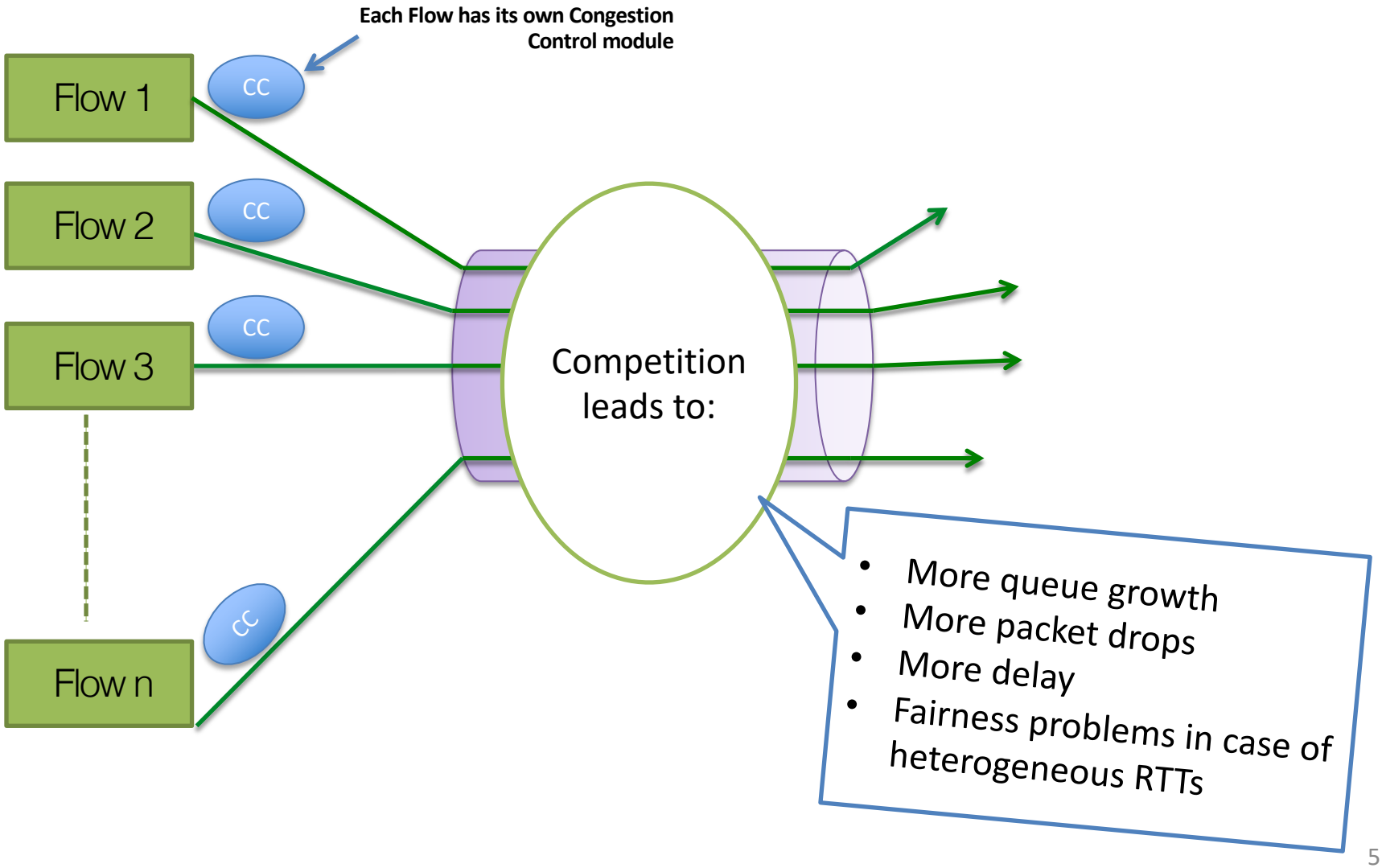
Context/Background

- WebRTC/RTCWeb enables real-time communication between web-browsers.

Context/Background

- Congestion control for real-time media is delay sensitive
  - However, Except TFRC, there are no standardized protocols until now
- IETF RMCAT WG is formed to develop standards for RTP based Media
- Three mechanisms are WG items
  - Cisco's Network Assisted (NADA)
  - Google's Google Congestion Control
  - Ericsson's ScreAm

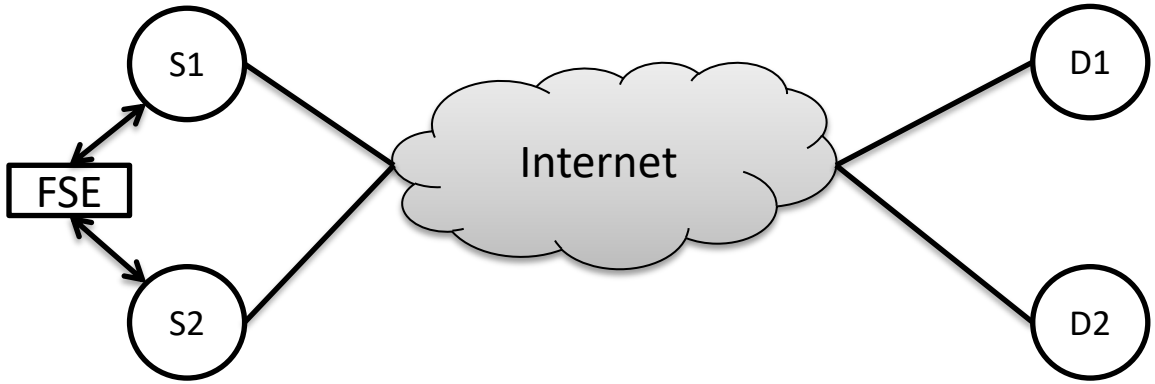
Problem Statement



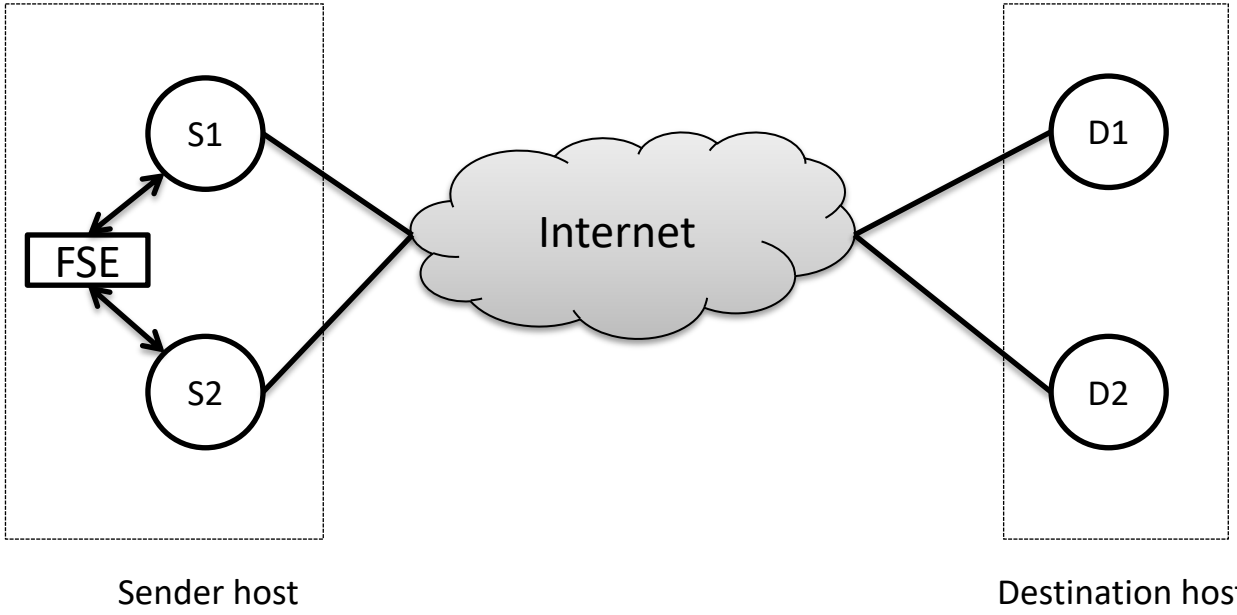
Problem Statement

- This can be solved with a single congestion controller, but it's too complicated
  - Congestion Manager
- Ensemble-TCP, EFCM
    - Only for TCP, and requires Extensive changes

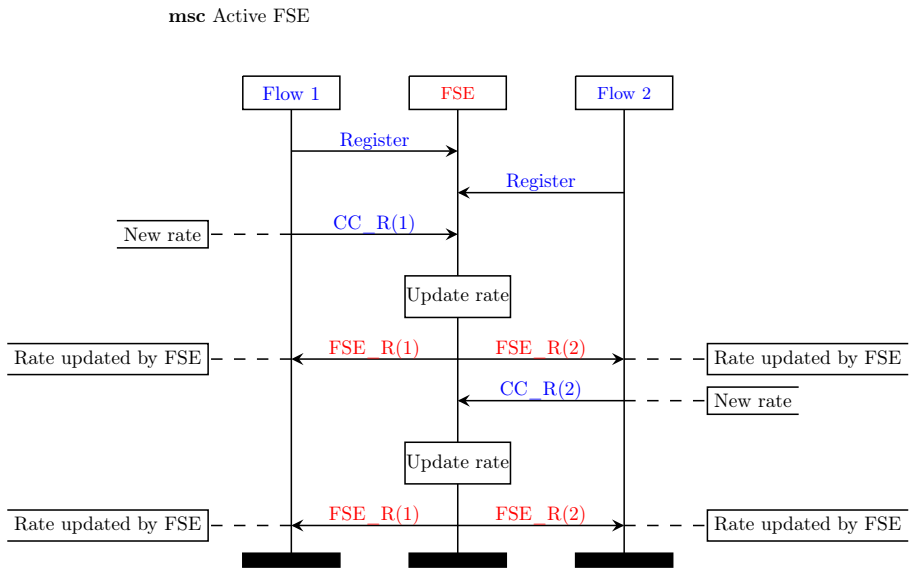
Example Architecture

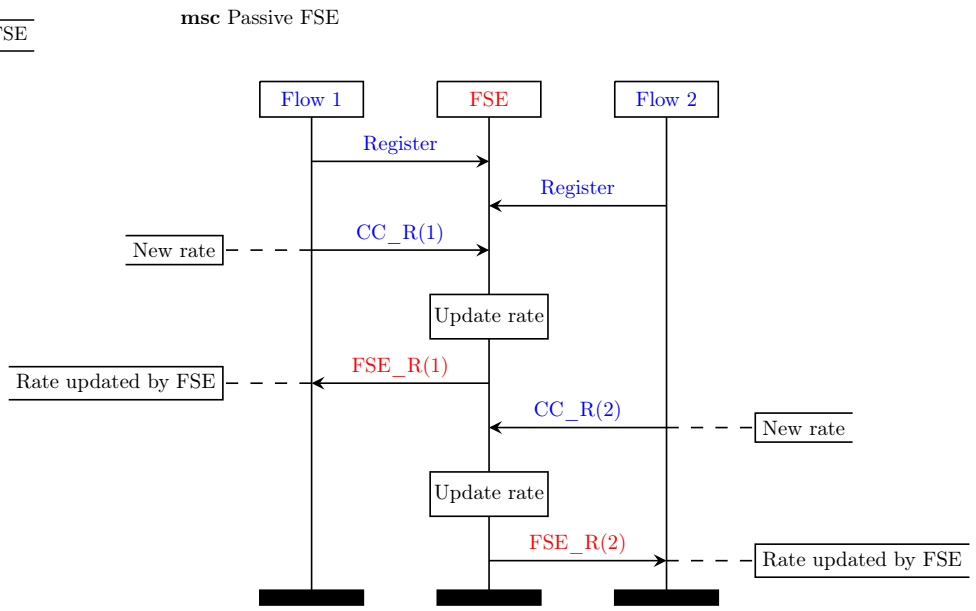


Example Architecture



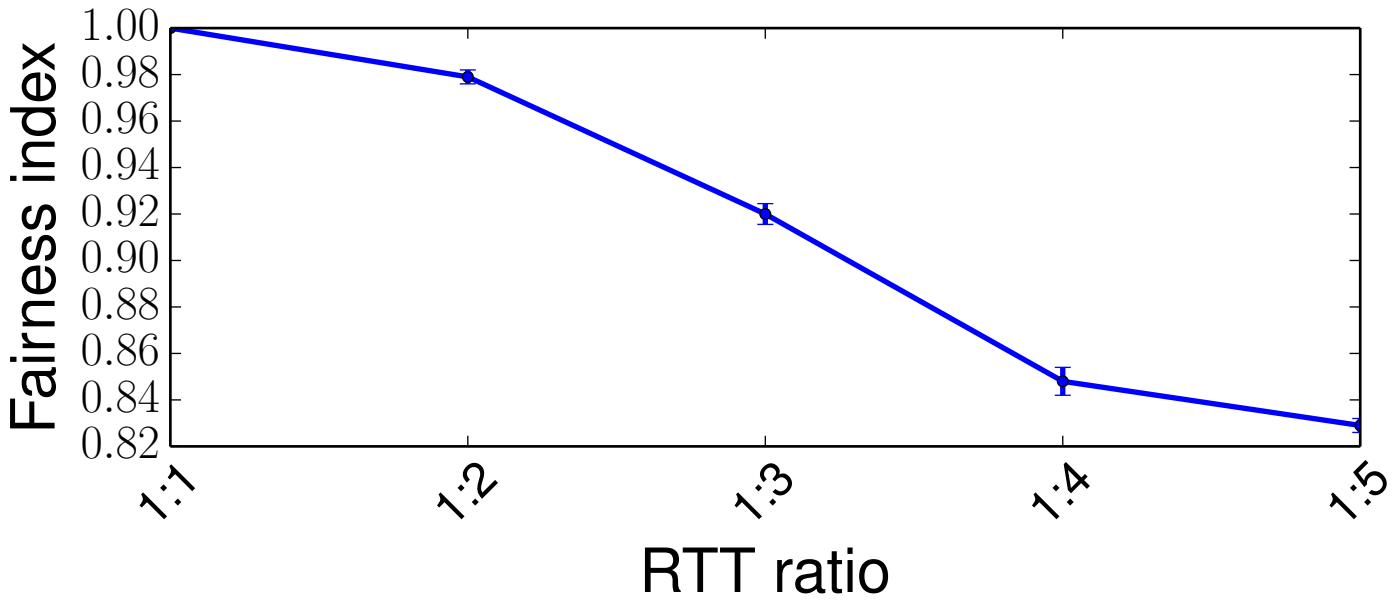
Versions?


msc Active FSE
Flow 1
FSE
Flow 2
Register
Register
New rate
CC_R(1)
Update rate
Rate updated by FSE
FSE_R(1)
FSE_R(2)
Rate updated by FSE
CC_R(2)
New rate
Update rate
Rate updated by FSE
FSE_R(1)
FSE_R(2)
Rate updated by FSE


msc Passive FSE
Flow 1
FSE
Flow 2
Register
Register
New rate
CC_R(1)
Update rate
Rate updated by FSE
FSE_R(1)
CC_R(2)
New rate
Update rate
FSE_R(2)
Rate updated by FSE

Why Passive version doesn't always work?

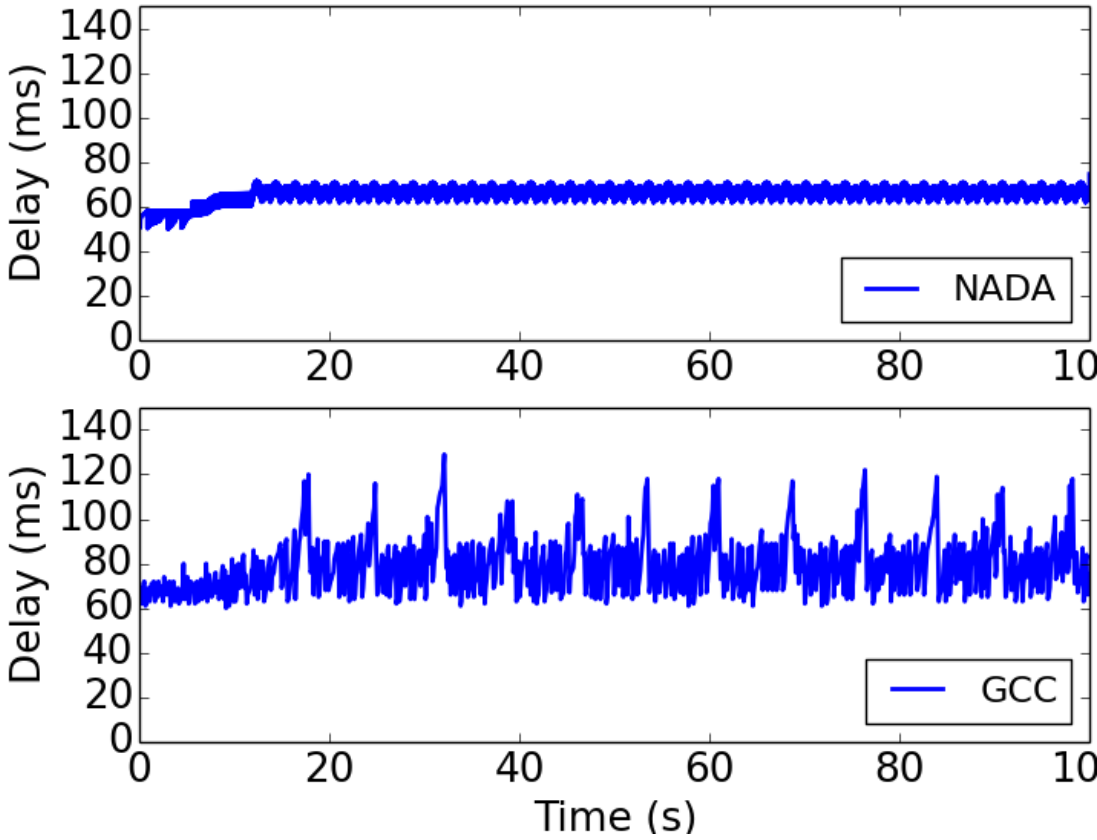
- …



Adaptation of the algorithm

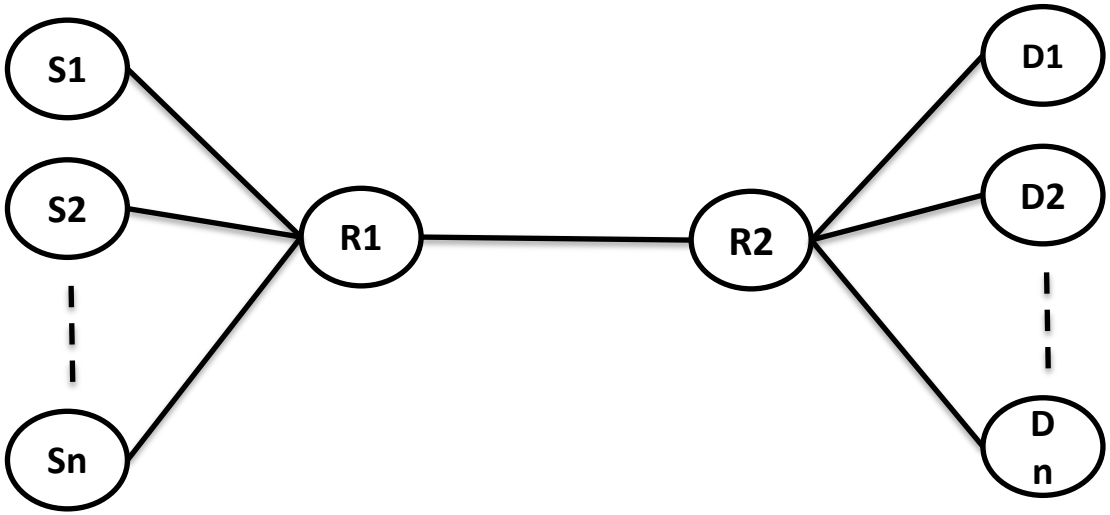
- …

Adaptation of the algorithm



Simulation Testbed

- Implemented the mechanism in both Chromium Browser and ns-2
- Actual tests and media characteristics are hard to predict
  - IETF RMCAT group prepared a number of test-cases
    - Dumbbell topology
    - With a bottleneck queue of 300ms
    - We show two pertinent test cases: Round-trip time fairness and media pause and resume
  - Prioritization test
  - Delayed feedback test



Prioritization Test

- Controllable fairness with prioritization is one of the key requirements of WebRTC

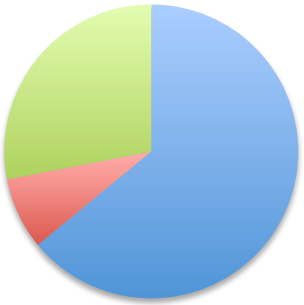
Have you personally experienced that the network traffic of applications on your computer have influenced each other?

Yes, and I found it annoying

Yes, but I didn't care

No: this never happened - or if it did, I didn't notice

If there was an easy-to-use tool that would let me prioritize how my applications access the network, I'd use it



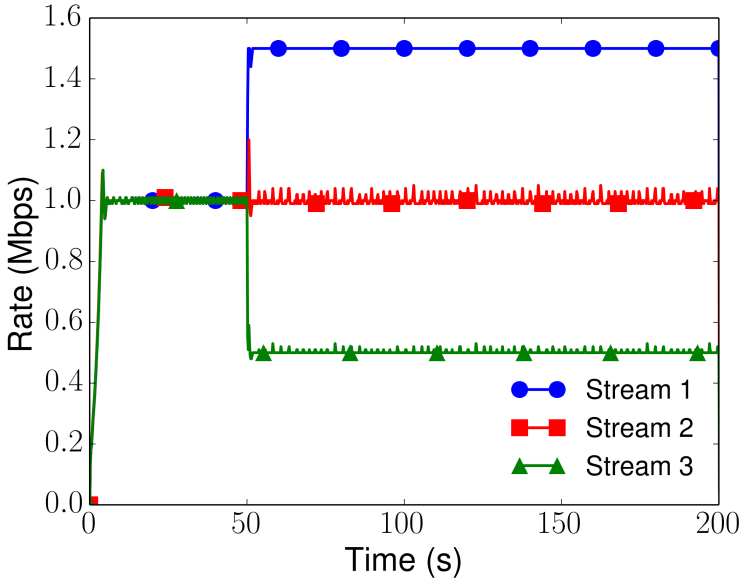
Yes

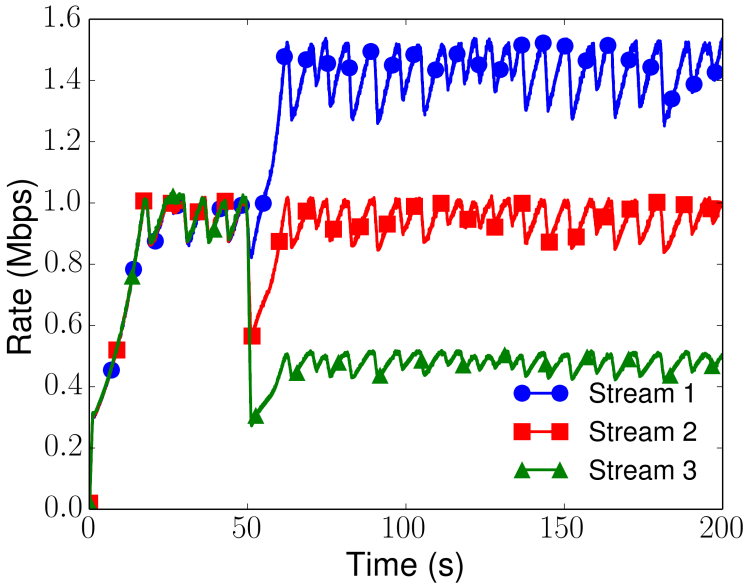
No

Something else (other or "I use QoS" etc)

Prioritization Test

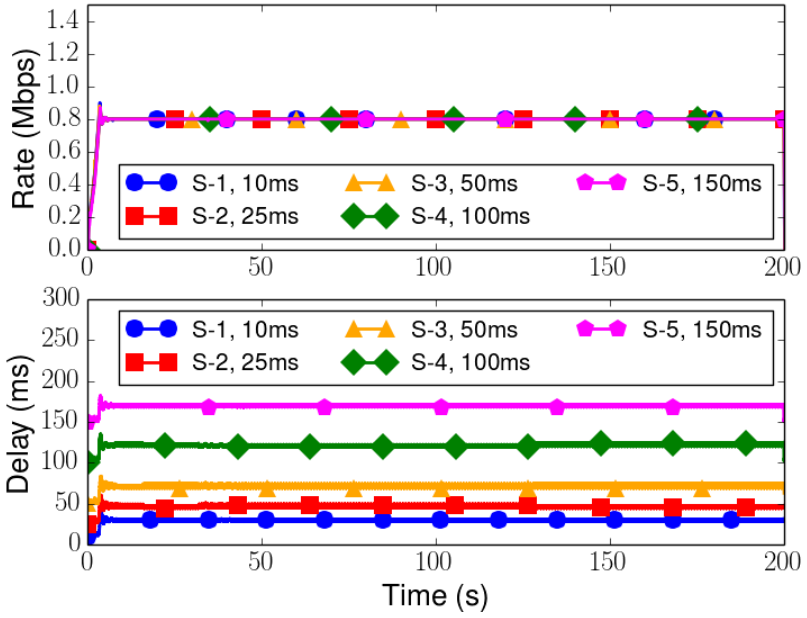
- The FSE can assign rates based on the priorities assigned from the uses
- *One way base delay, 50ms*

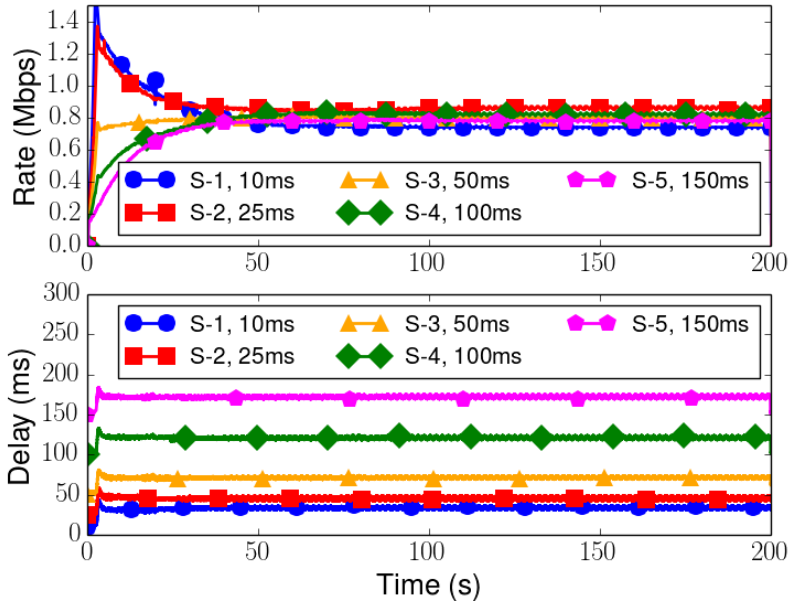




Round-trip time fairness, NADA flows

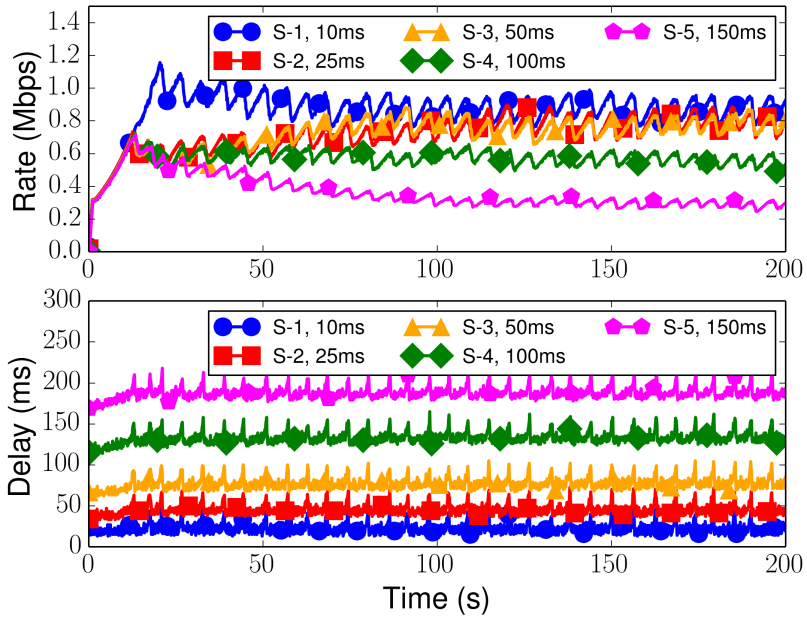
- One-way delays of s1, s2, s3, s4 and s5 are 10ms, 25ms, 50ms, 100ms, and 150ms respectively, and bottleneck capacity, 4Mbps

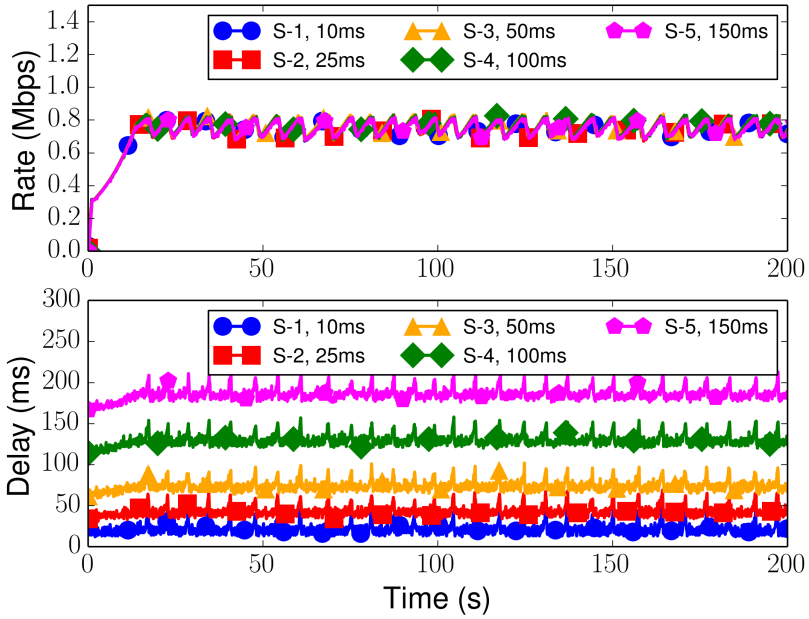




Round-trip time fairness – GCC flows

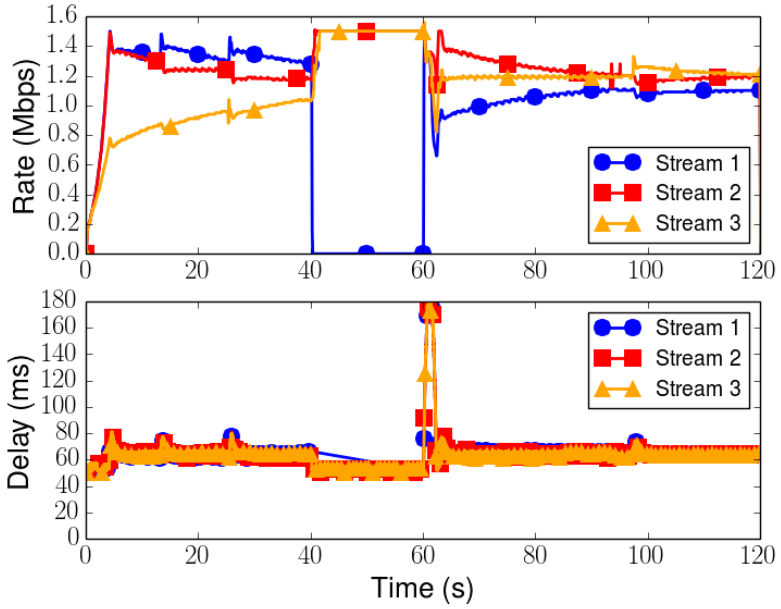
- One-way delays of s1, s2, s3, s4 and s5 are 10ms, 25ms, 50ms, 100ms, and 150ms respectively, and bottleneck capacity, 4Mbps

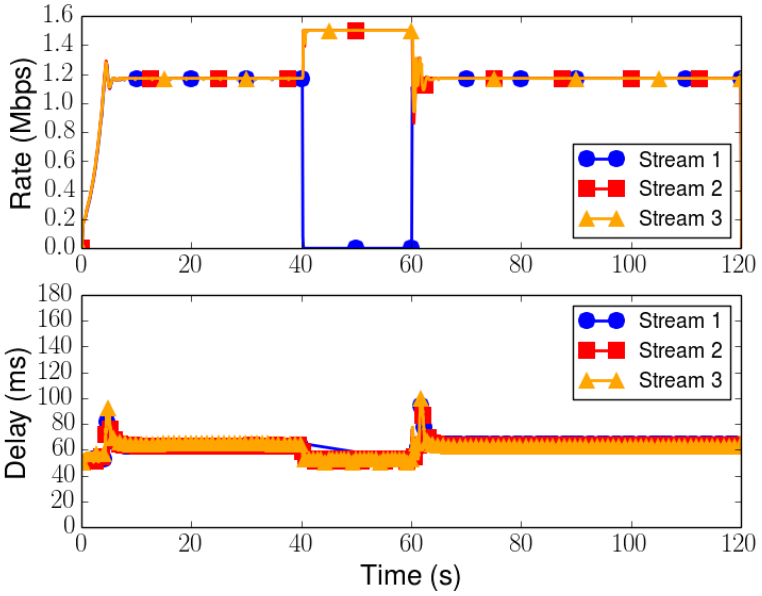




Media Pause and Resume (NADA)

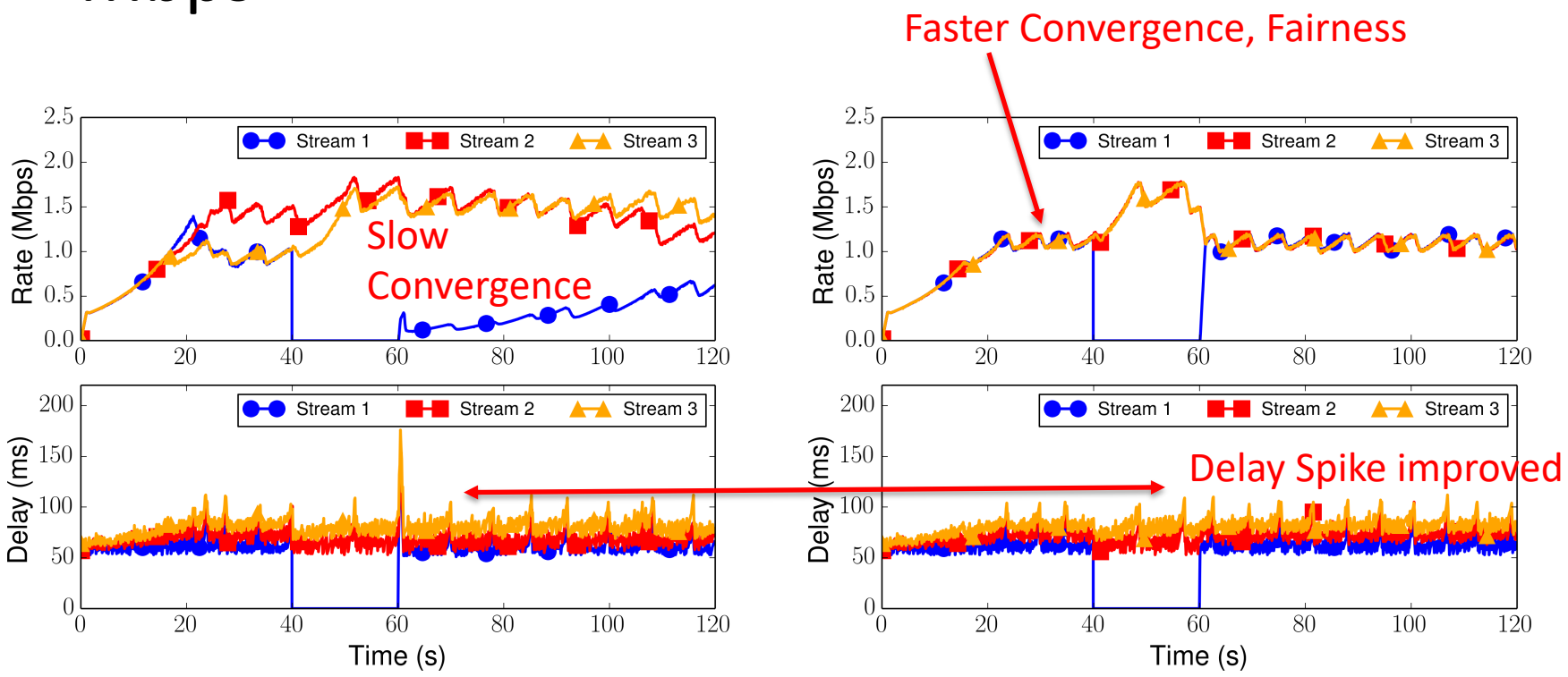
- One way delay 50ms and Path capacity 3.5 Mbps





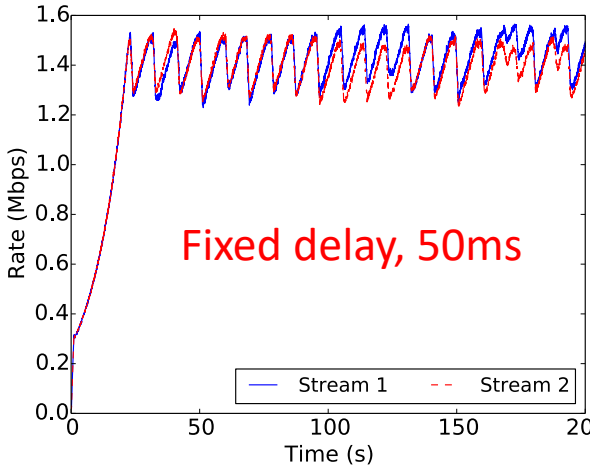
Media Pause and Resume (GCC)

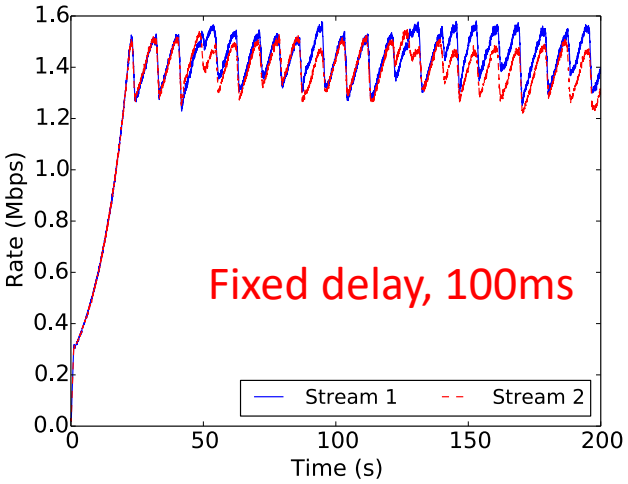
- One way delay 50ms and Path capacity 3.5 Mbps

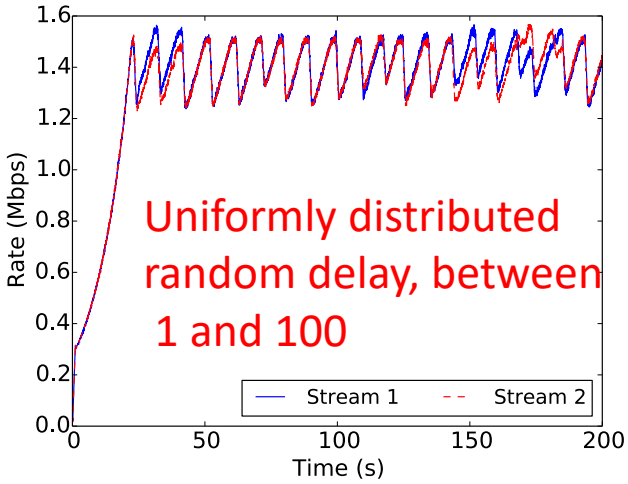


Delayed feedback test

- Delay between stream 1 and the FSE is delayed







Q&A