Streaming Video and TCP-Friendly Congestion Control

Sugih Jamin
Department of EECS
University of Michigan
jamin@eecs.umich.edu

Joint work with:
Zhiheng Wang (UofM),
Sujata Banerjee (HP Labs)

Video Application on the Internet

Adaptive playback streaming:

Sender sends data i at time t_i

Receiver receives at time $t_i + \Delta$,
$\Delta =$ propagation delay + queueing delay

To smooth out variable queueing delay, receiver buffers some amount of data (i to $i + k, k > 0$) before playing back data i

By the time receiver is ready to play back data $i + k$, hopefully it would have arrived

Otherwise, increase buffering (hence “adaptive”)

Sugih Jamin (jamin@eecs.umich.edu)

Video Streaming

Two ways to send data:

- bulk transfer: transfer before playback
- streaming: transfer while playback

Why Streaming?

- shorter playback start time
- smaller receiver buffer requirement
- smaller interaction delay requirement

Expectations vs. Reality

Streaming media service requirements:

- resource intensive
- smooth (low variance) throughput

Internet service characteristics:

- shared resource
- variable bandwidth
- unpredictable network latency
- lossy channel

Streaming Video over the Internet

Effect of transient changes in available bandwidth:

- empty buffer on playback
- playback pause on rebuffering
- larger buffer size increases start time
  (consider live interactive sessions)

Applicable to other streaming data:
scientific visualization, massively multiplayer
gaming dynamic object, web page download

Case Study: Windows Media Player

Application characteristics:

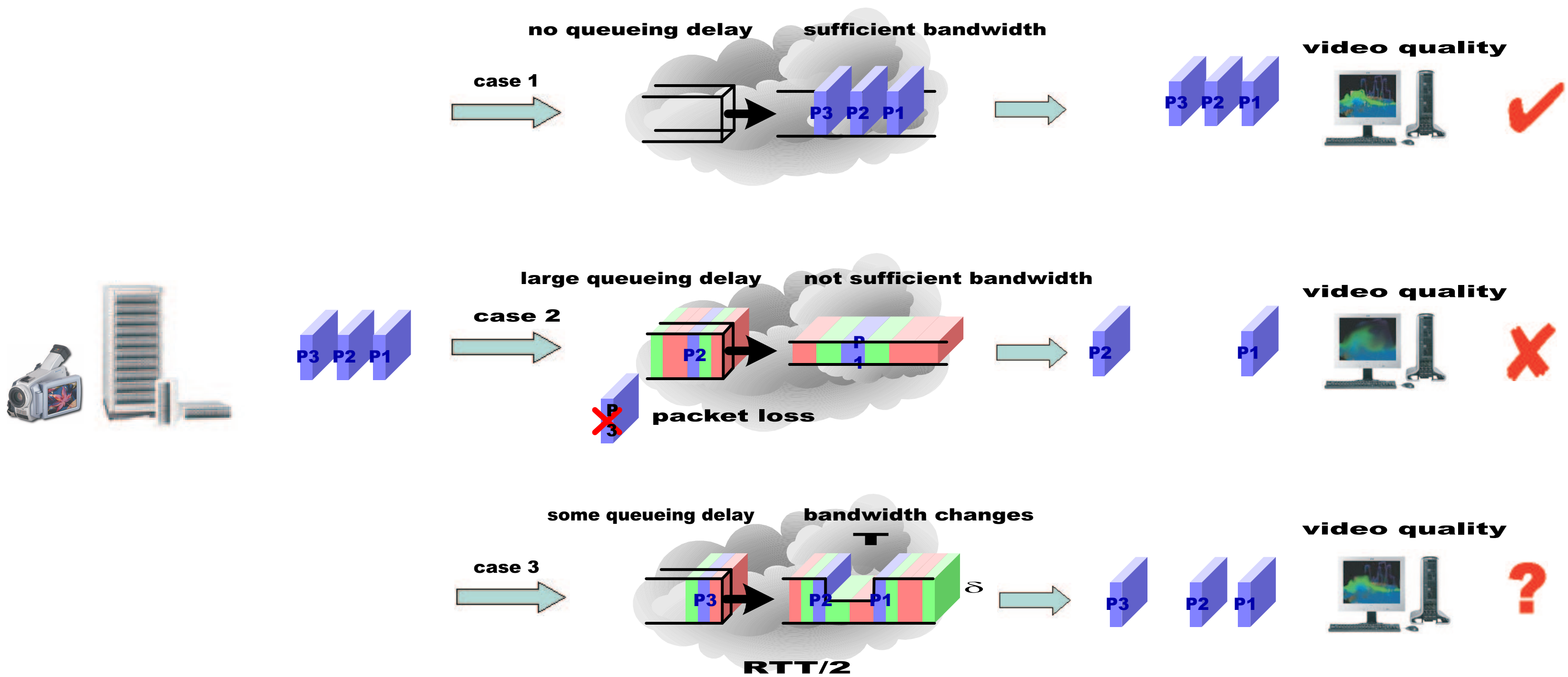
WM Server sends traffic at a constant bit rate

WMP client pauses video playback until sufficient packets have been buffered (rebuffering)

WMP client asks for retransmission to recover lost packet. If lost packet cannot be recovered, the whole frame is considered lost

WM Server reduces sending rate when lower available bandwidth is detected

Streaming Video Quality



Sugih Jamin (jamin@eecs.umich.edu)

Measuring Streaming Video Quality

Metrics:

- server transmission rate (service rate)
- client rebuffering probability
- client rebuffering duration
- client frame loss

Improving User Perceived Quality

User less annoyed with lower but consistent quality than continual rebuffering

Changes in available bandwidth cause changes in rebuffering probability and duration

Streaming video needs low loss rate and smooth available bandwidth to reduce user annoyance

Need: smooth congestion control mechanism

TCP-Friendliness

TCP is the standard transport protocol

TCP does congestion control by linear probing for available bandwidth and multiplicative decrease on congestion detection (packet loss)

"A congestion control protocol is TCP-friendly if, in steady state, its bandwidth utilization is no more than required by TCP under similar circumstances" [Floyd et al., 2000]

TCP-friendliness in a proposed protocol ensures compatibility with TCP

TCP-Friendly Rate Control (TFRC)

Goals:

- to provide streaming media with steady throughput
- to be TCP-friendly

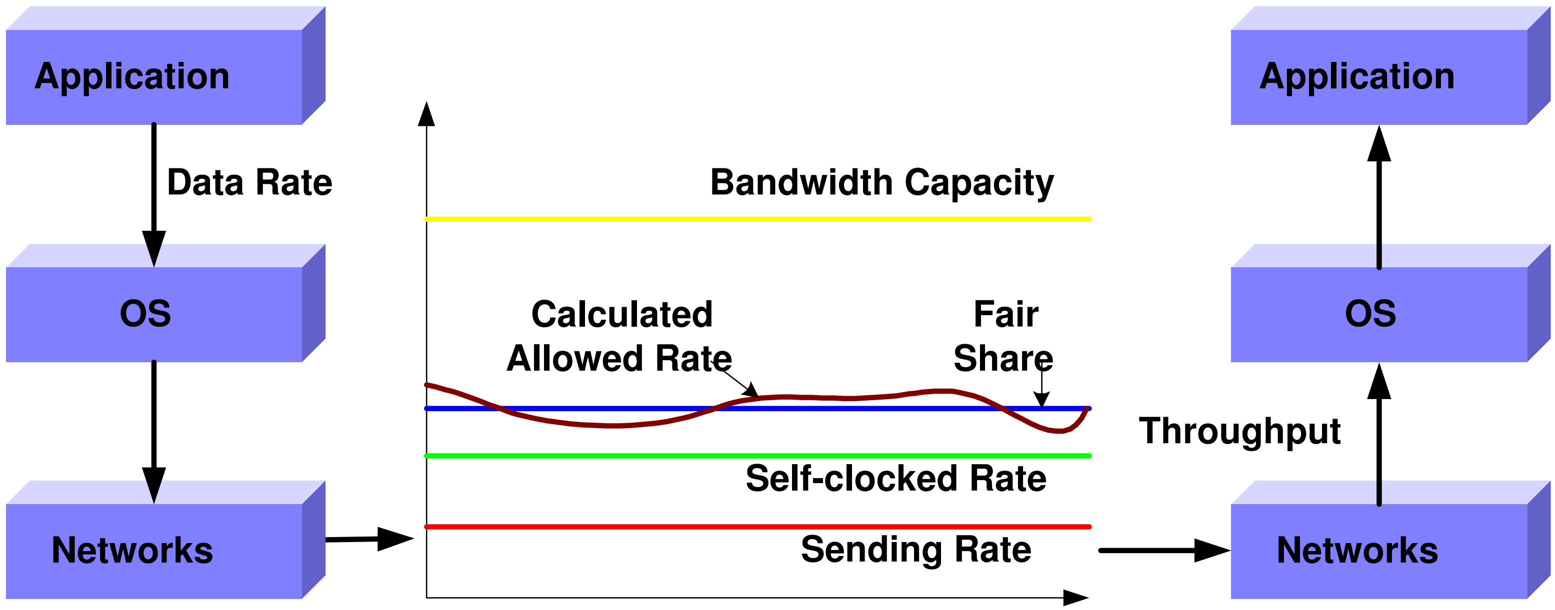
Instead of reacting to individual losses,
tries to satisfy the TCP throughput function over time:

$$T = \frac{s}{R\sqrt{\frac{2p}{3}} + t_{RTO}(3\sqrt{\frac{3p}{8}})p(1 + 32p^2)}$$

T: TCP throughput; s: packet size; p: loss rate
R: path RTT; t_{RTO}: re-transmit timeout

Sugih Jamin (jamin@eecs.umich.edu)

Terminologies



Sugih Jamin (jamin@eecs.umich.edu)

Terminologies (contd)

Data rate: the rate at which an application generates data

Sending Rate: the rate at which a connection sends data

Self-clocked rate: upper bound on the sending rate calculated by TFRC

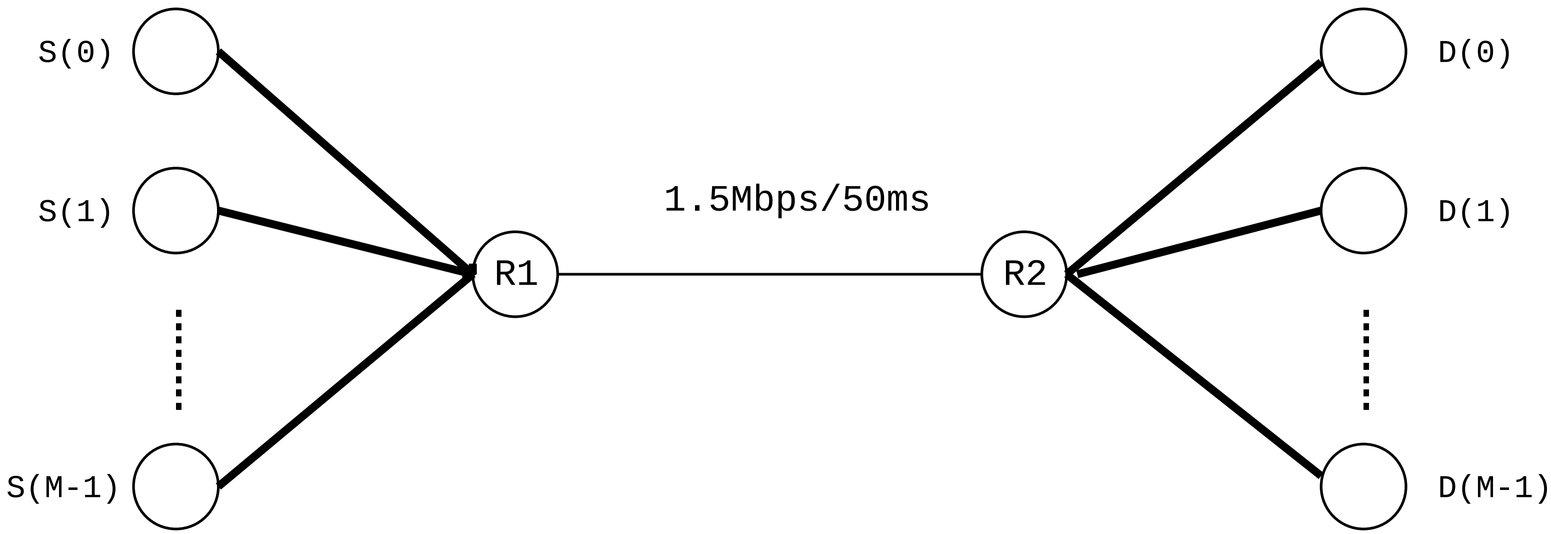
Fair share: TCP's throughput during bulk data transfer

Fair share load: ratio between the sending rate and the fair share

Throughput: the incoming traffic rate measured at the receiver

Does TFRC Provide Smoother Throughput?

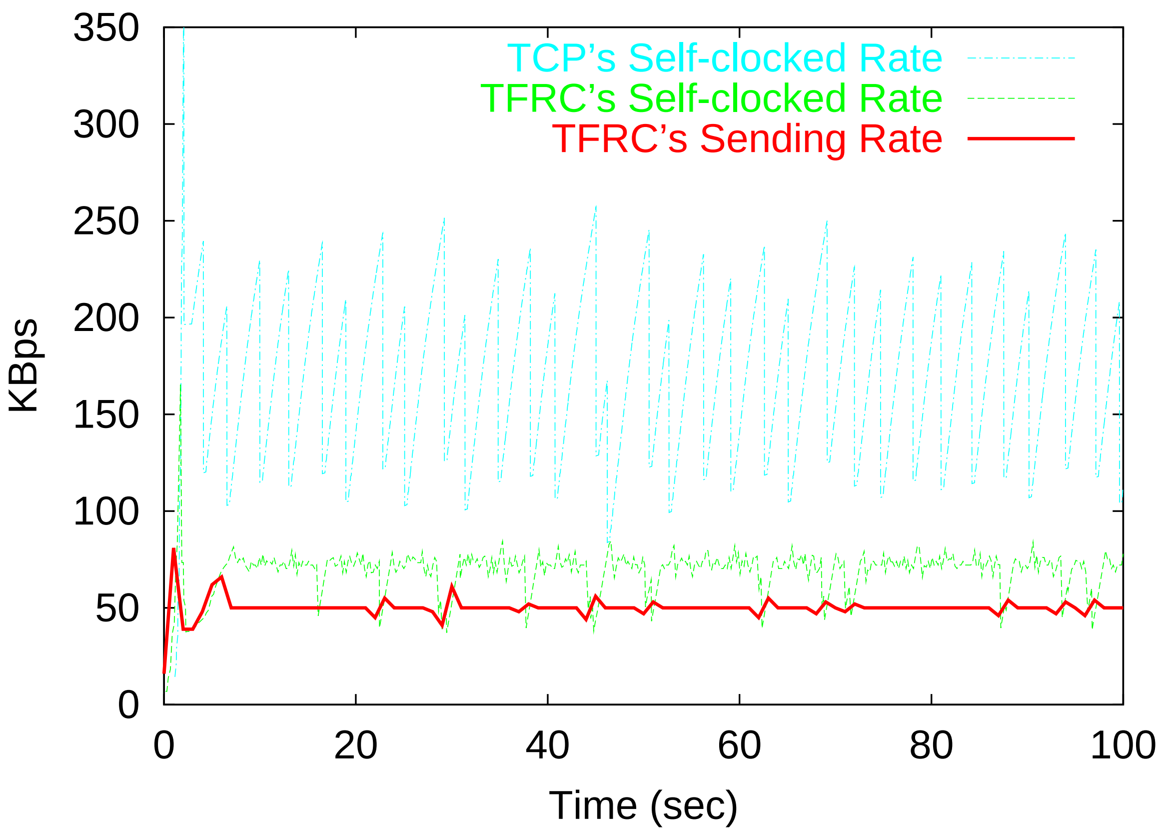
Experiment setup:



- Data source: CBR-traffic
- Background traffic
  - o long/short-lived TCP flows with infinite amount of data
  - o flash crowd: large number of short TCP bursts
  - o long-range dependent traffic: a number of Pareto distributed ON/OFF flows

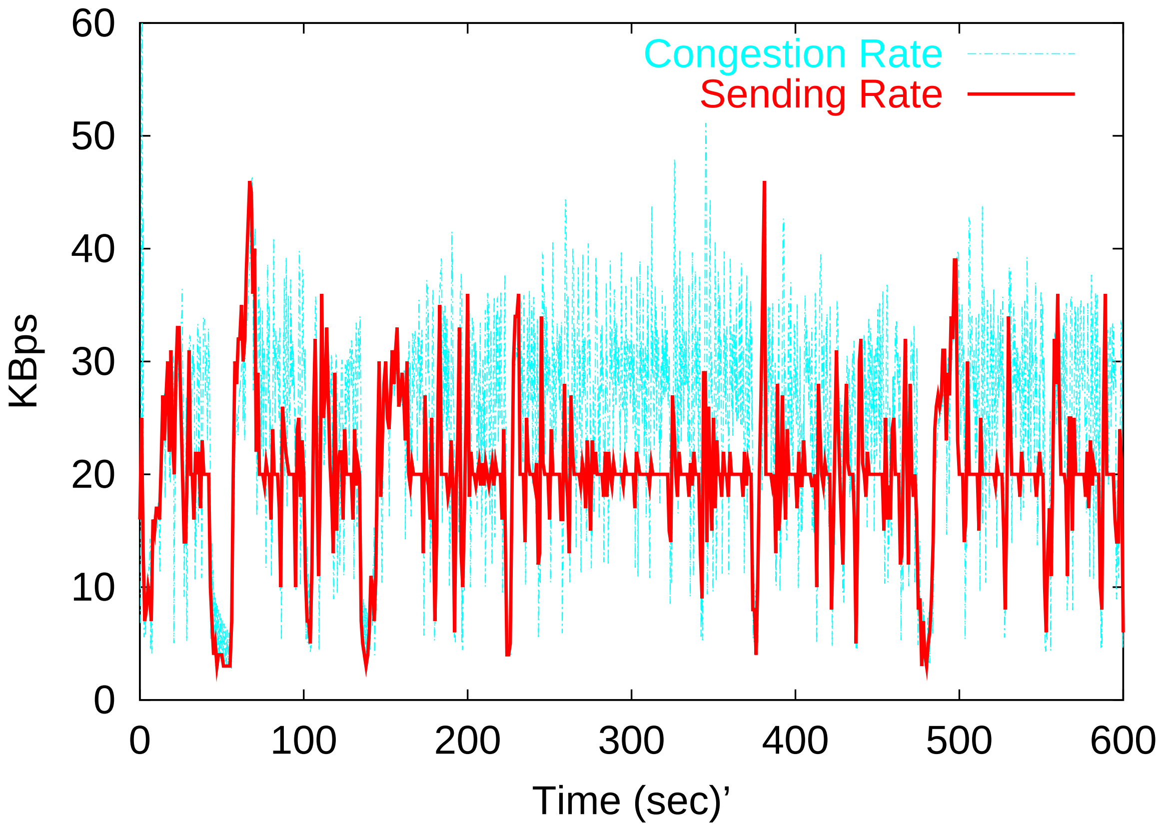
Sugih Jamin (jamin@eecs.umich.edu)

Not that Smooth



- Data rate: 50KBps
- Background traffic: 1 long-lived TCP

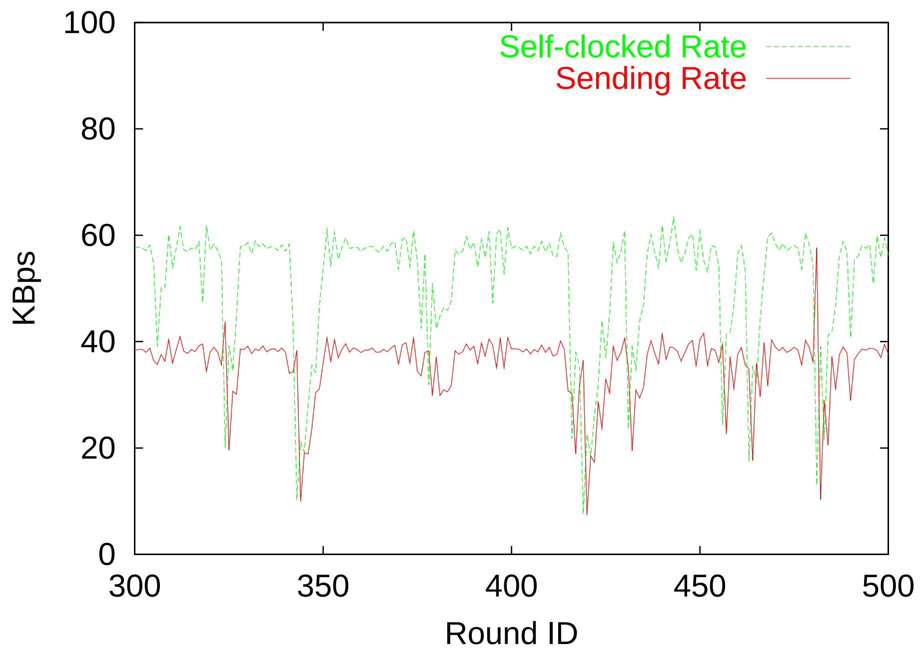
Worse with Bursty Background Traffic



- Data rate: 20KBps
- Background traffic: 1 long-lived TCP + 5 ON/OFF flows

Internet Experiments

A sample path between MI and CA



- Data rate: 40 KBps
- RTT: 67 msec
- Loss event rate: 0.24%

MARC's Design Motivation

TFRC congestion control is memoryless, whereas:

- streaming media is "well-behaved"
  when there is no congestion, streaming applications cannot always utilize their fair share fully
- but during congestion, TFRC applies the same rate reduction principle to streaming media traffic as to bulk data transfer traffic

Media Aware Rate Control (MARC) proposition:
"Well-behaved" streaming applications should be allowed to reduce their sending rate more slowly during congestion

Media-Aware Rate Control (MARC)

- Define token value C to keep track of a connection's fair share utilization

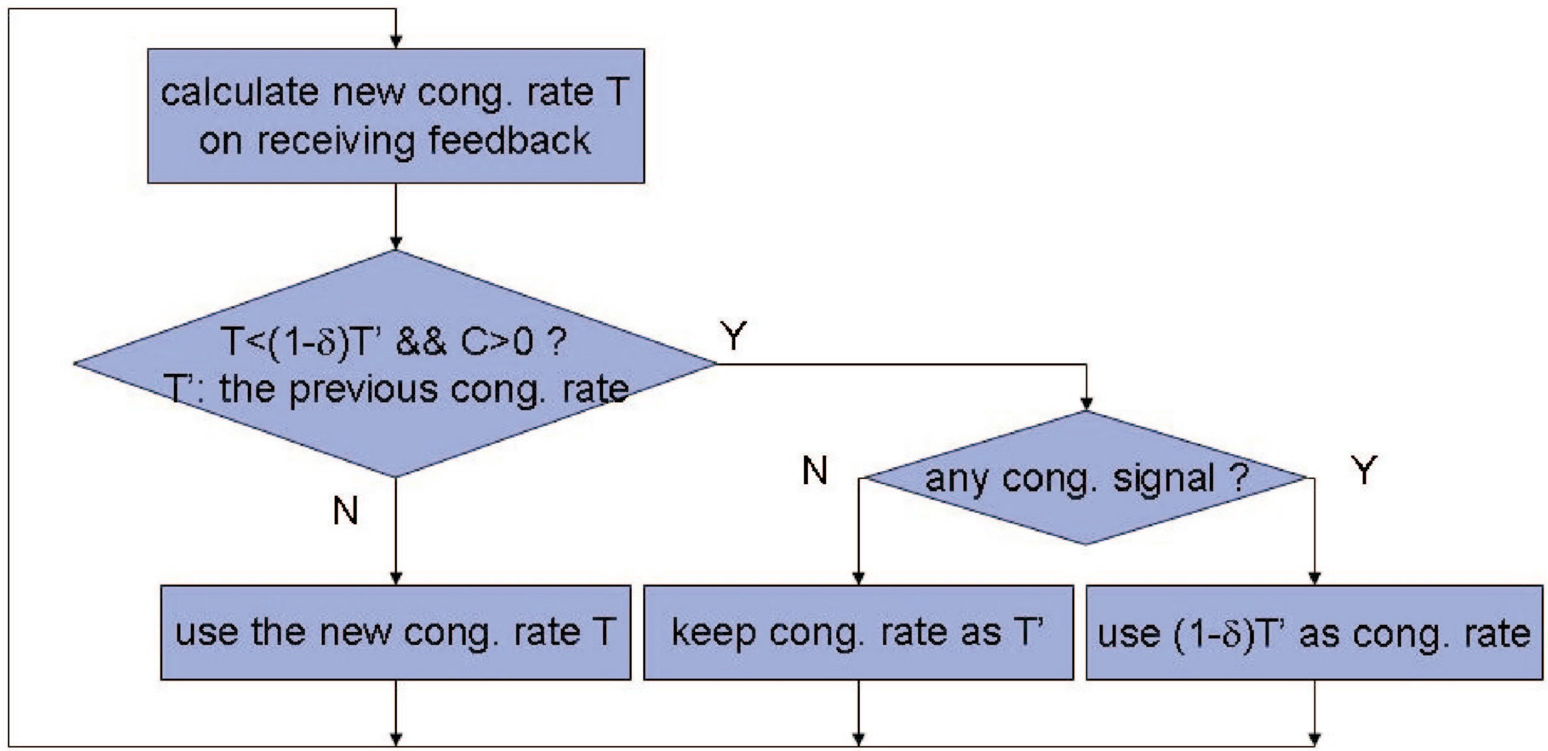
$$C = \beta C' + (T - W_{send})I$$

C: token
β: decay factor
C': previous token value
T: previous calculated self-clocked rate
W_{send}: previous sending rate
I: feedback interval

- We use $\beta = 0.9$

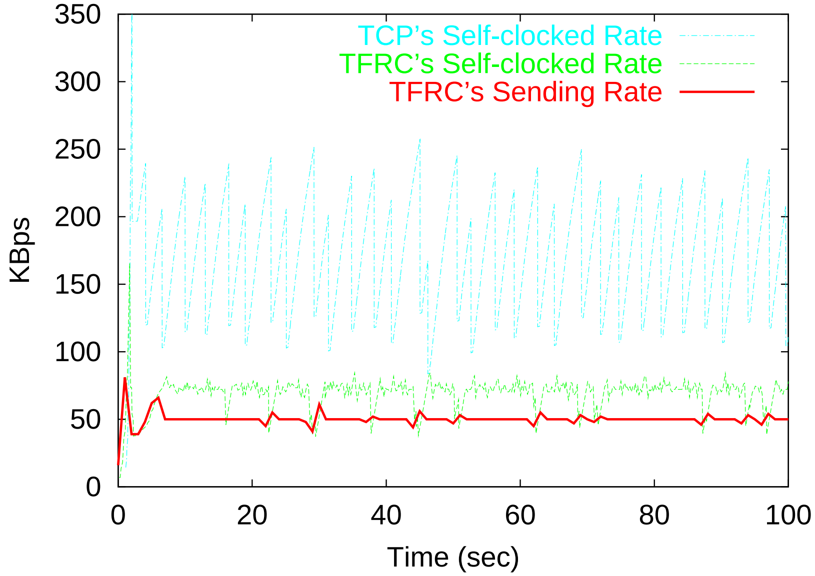
Sugih Jamin (jamin@eecs.umich.edu)

Media-Aware Rate Control (MARC)

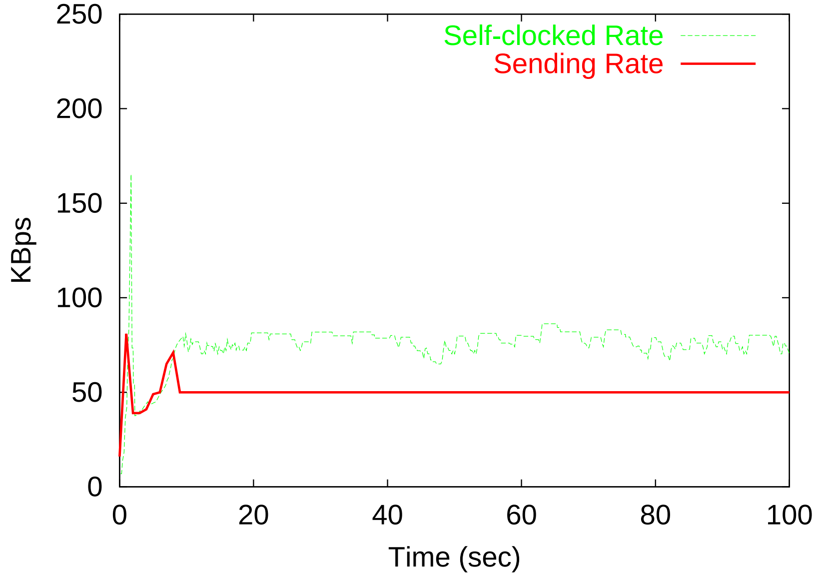


Our experiments use $\delta = 0.1$

MARC is Effective



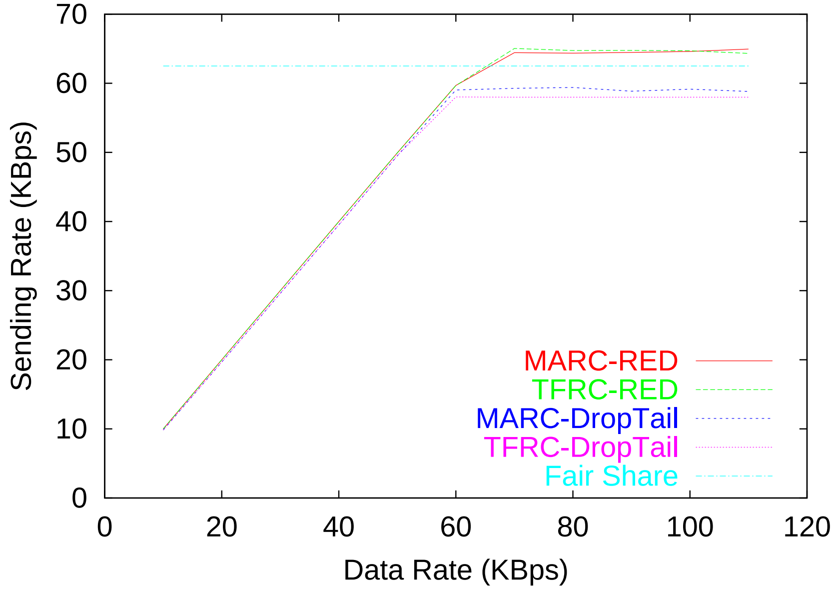
TFRC



MARC

- Data rate: 50KBps
- Background traffic: 1 long-lived TCP

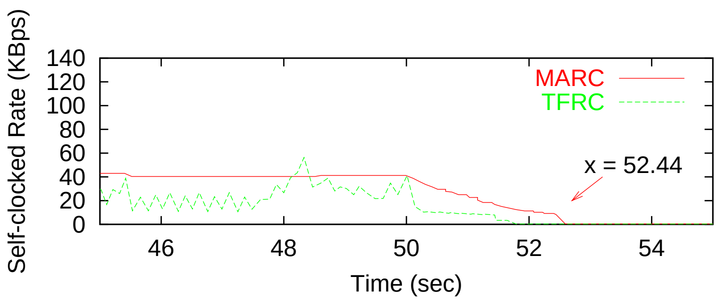
MARC is TCP-Friendly



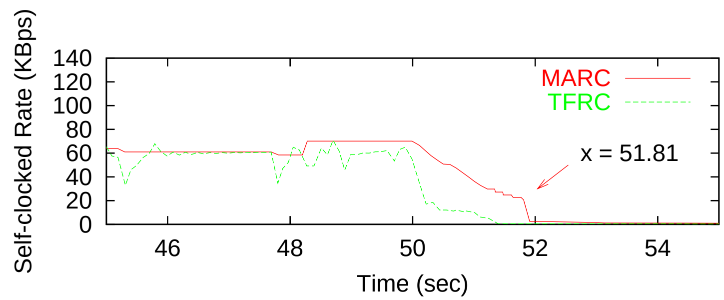
- Date rate: 10 CBR sources
- Background traffic: 10 long-lived TCP

Reaction Time to Persistent Congestion

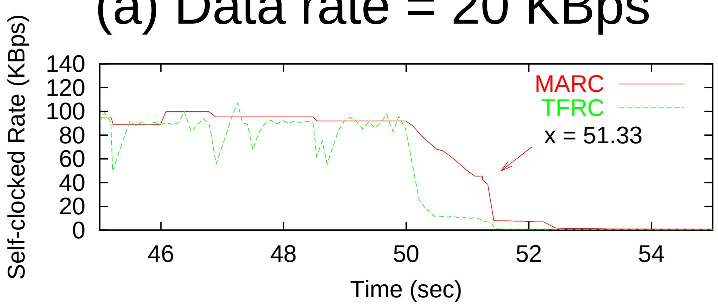
- Congestion at 50th sec, RTT: 80 msec
- Fair share before congestion: 140 KBps



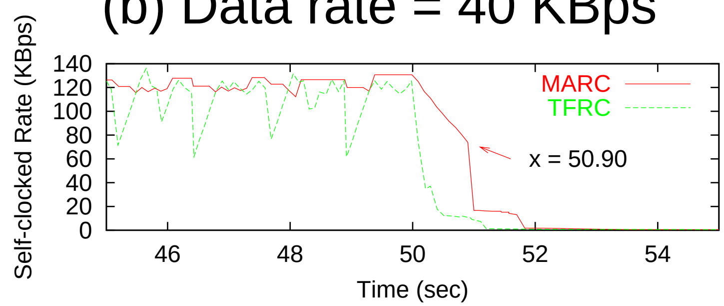
(a) Data rate = 20 KBps



(b) Data rate = 40 KBps



(c) Data rate = 60 KBps

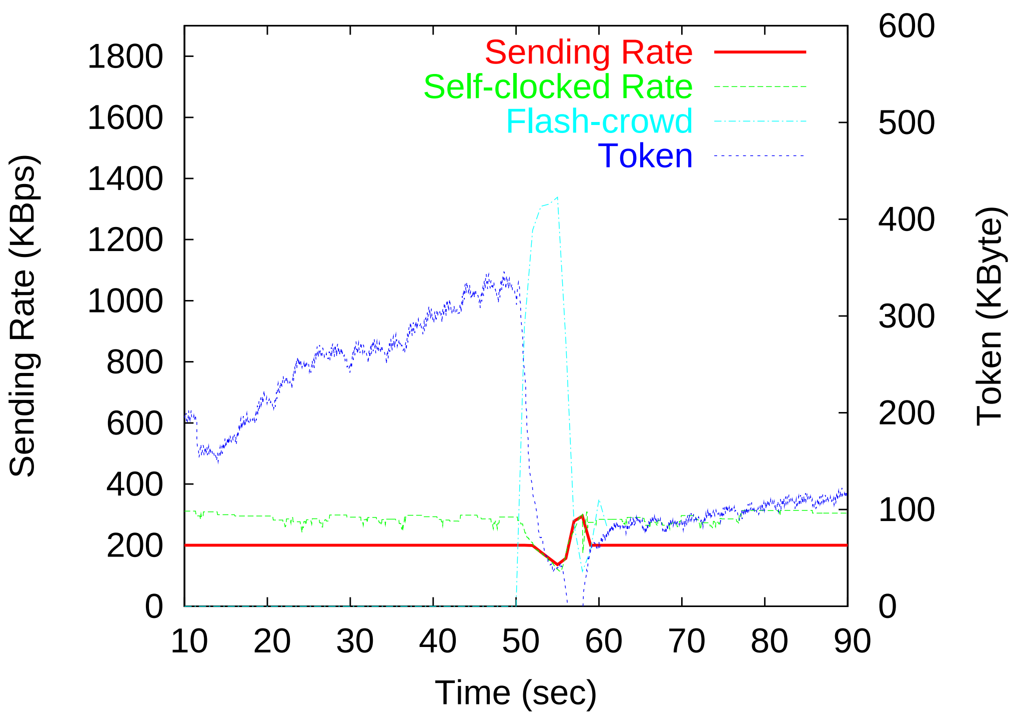


(d) Data rate = 80 KBps

Without token, MARC behaves exactly like TFRC.

Sugih Jamin (jamin@eecs.umich.edu)

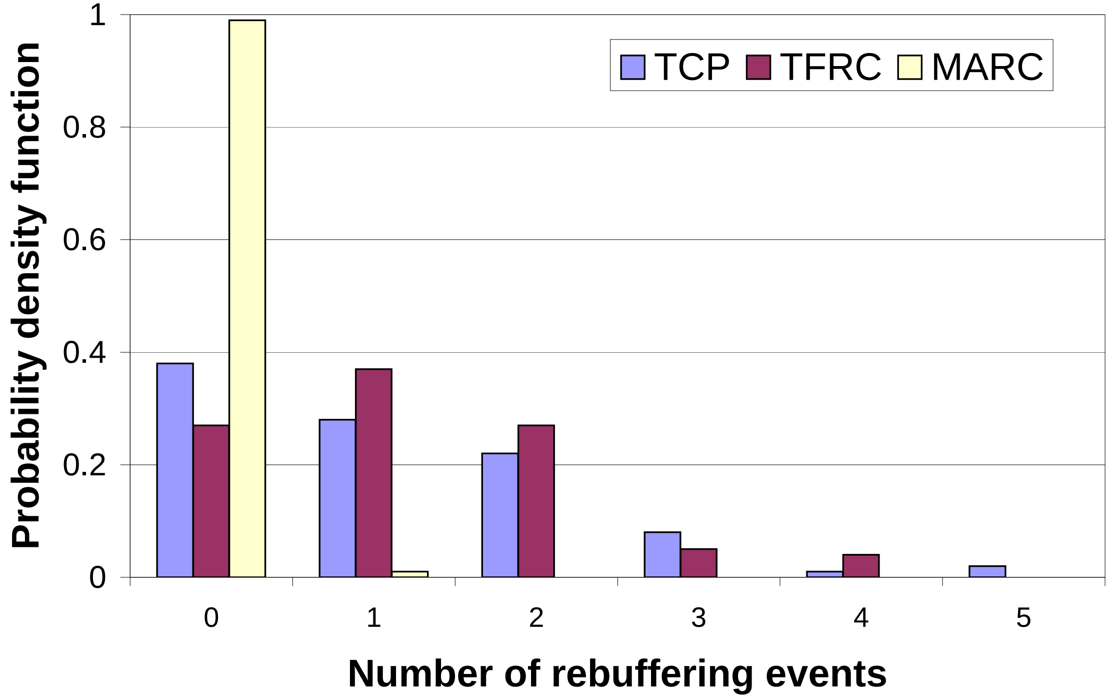
Token Dynamics



- 1 long-lived TCP and 1 MARC flows
- Data rate: 100 KBps
- Flash-crowd (800 short-lived TCP) starts at the 50th second, lasts for 5 seconds

Sugih Jamin (jamin@eecs.umich.edu)

MARC Improves User-Perceived Quality



- Data rate: 44KBps
- Background traffic: 1 long-lived TCP + 1 ON/OFF flow

Future Works

- Layered video adaptation with MARC
- Analyzing MARC
- Streaming media over end-host multicast
  - o multiple receivers
    - congestion control on end-host multicast
  - o multiple sources
    - Integrated Flow Control