PIQI-RCP: Design and Analysis of Rate-Based Explicit Congestion Control

Saurabh Jain

Joint work with Dr. Dmitri Loguinov

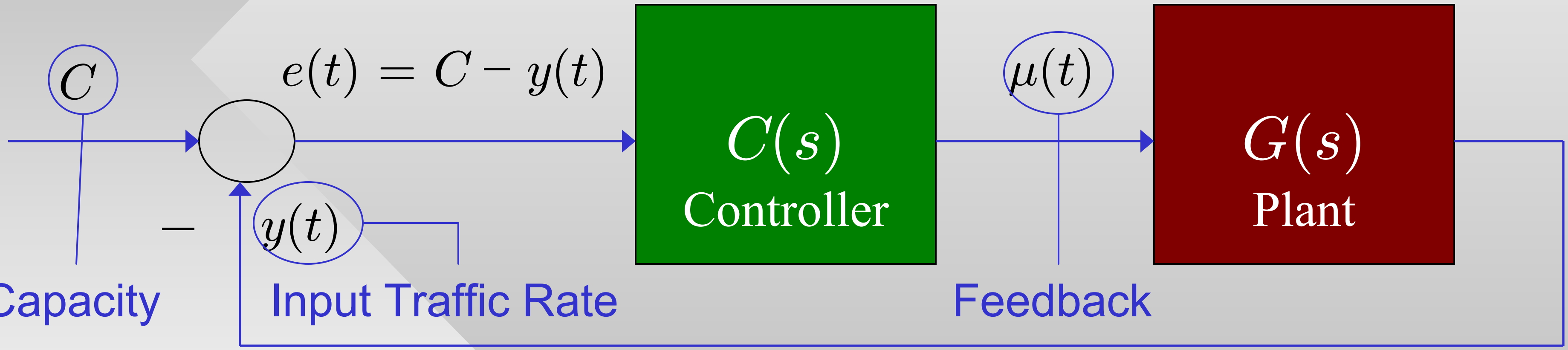
June 21, 2007

Agenda

- Introduction
- Analysis of RCP
- QI-RCP
- PIQI-RCP
- Comparison
- Wrap Up

Introduction

- Congestion control can be modeled as a delayed feedback control system



- Each flow i in the plant, upon receiving a congestion feedback $\mu_l(t)$, applies a control equation to compute its sending rate x_i as

Round-Trip Time

$$\frac{dx_i}{dt} = f_i(x_i, \mu_l(t - \overleftarrow{D}_{il}), D_i, \ldots)$$

Backward delay from controller to flow i

Introduction 1

- Congestion feedback is a function of the input traffic rate (i.e., sending rates of individual flows), link capacity, etc.

$$\frac{d\mu_l}{dt} = g(e_l(t), T, \ldots)$$

Control interval

- For a stable system, the sending rates of individual flows and the feedback converge to their equilibrium value

$$\lim_{t\to\infty} x_i(t) = x_i^*$$
$$\lim_{t\to\infty} \mu_l(t) = \mu_l^*$$

- It is also desirable to have efficiency and fairness

$$\lim_{t\to\infty} e(t) = C - y(t) = 0$$

$$x_1^* = x_2^* = \ldots = x_N^*$$

Introduction 2

- The problem can also be formulated in the discrete time domain as difference equations

$$\begin{aligned} x_i(n+1) &= x_i(n) + f_i(x_i(n), \mu_l(n - D_{il}^{\leftarrow}), D_i, ...) \\ \mu_l(n+1) &= g_l(e_l(n), \mu_l(n), T, ...) \end{aligned}$$

- Congestion feedback can be
  - Implicit such as detection of packet loss or increase in RTT due to larger queuing delays
  - Explicit such as single-bit (e.g., RED-ECN) or multi-bit notification (e.g., packet loss rate, link prices, fair rate, queuing delay, change in sending rate)
- Proposed explicit congestion control methods include XCP, MKC, JetMax, MaxNet, RCP [IWQoS 2005]

Agenda

- Introduction
- Analysis of RCP
- QI-RCP
- PIQI-RCP
- Comparison
- Wrap Up

Analysis of RCP - Drawbacks

- In RCP, each router l uses a control equation:

$$R_l(t) = R_l(t-T)\left[1+\frac{\frac{T}{d_l}\left(\alpha(C_l - y_l(t)) - \beta\frac{q_l(t)}{d_l}\right)}{C_l}\right]$$

Control rate: $R_l(t)$; Control interval: T; Link capacity: C_l; Average RTT of flows in the system: d_l; Input rate of all flows: $y_l(t)$; Queue size: $q_l(t)$

  - α and β are gain parameters

- Each flow i adjusts its sending rate $x_i(t)$ as:

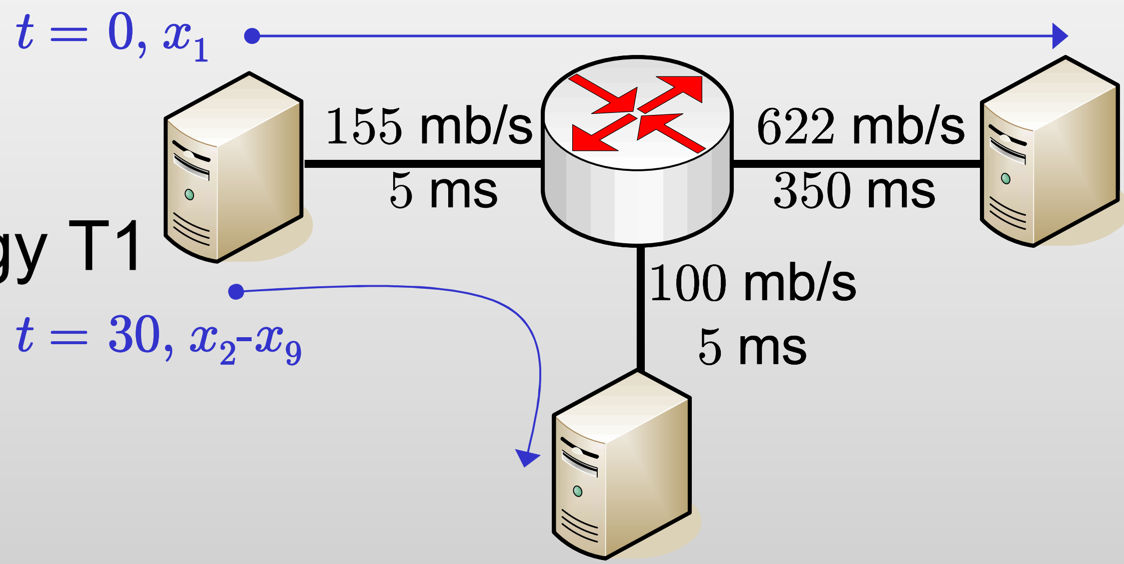
$$x_i(t) = \min R_l(t - D_i^{\leftarrow})$$

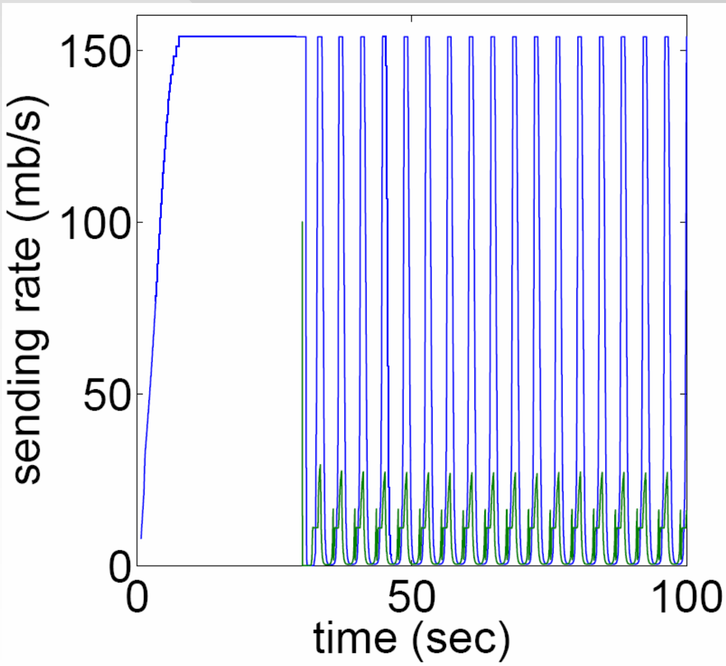
Received feedback from router l: $R_l(t - D_i^{\leftarrow})$

- Limited Understanding of Stability:
  - Stability analysis only available for homogeneous RTTs. For heterogeneous RTTs, results only available using simulations

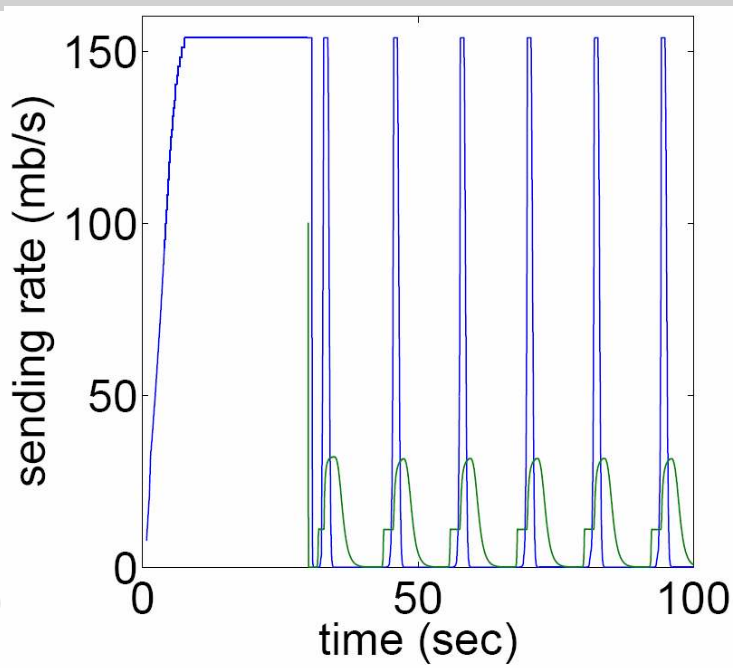
Analysis of RCP – Drawbacks 1

- We use $\alpha = 0.4$ and $\beta = 1$
- RCP is unstable in topology T1





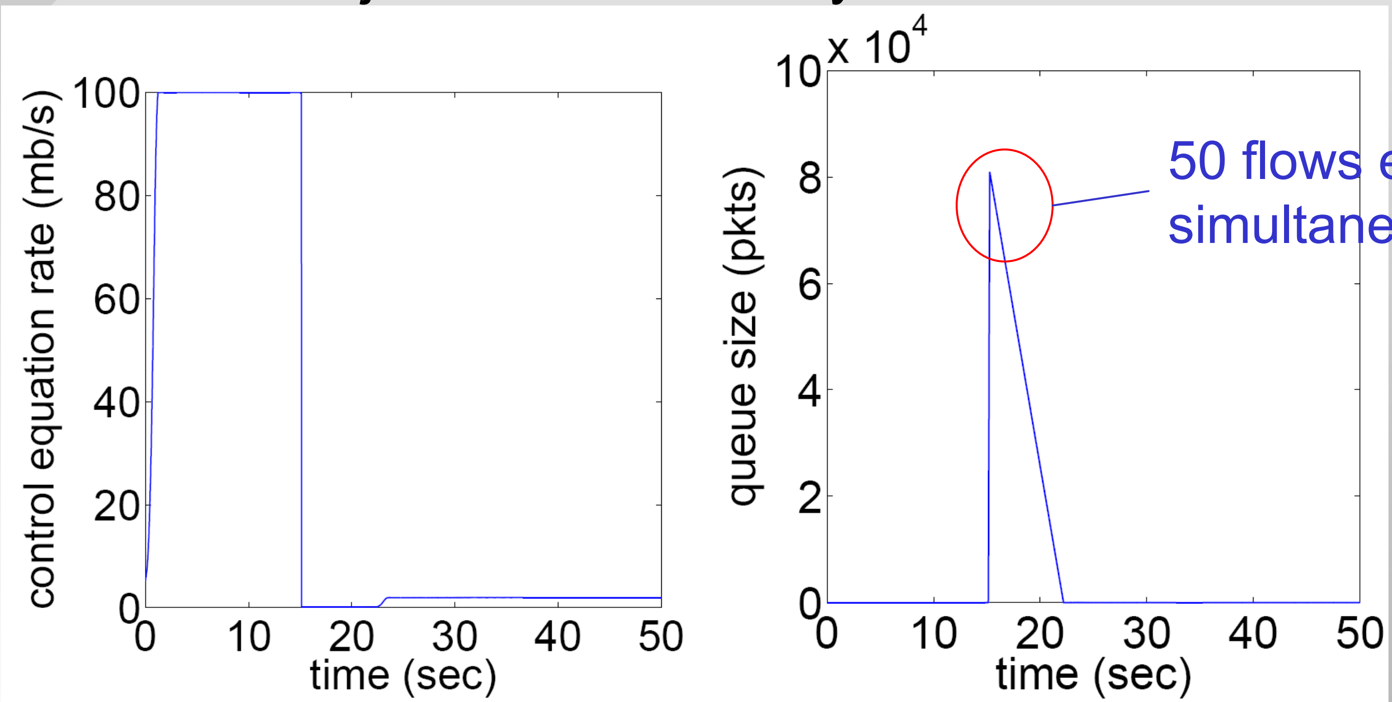
Oscillating bottlenecks



Fixed bottlenecks

Analysis of RCP – Drawbacks 2

- Link Overshoot:
  - Input traffic rate overshoots link capacity significantly when large number of flows join simultaneously



  - Significant packet losses and re-transmissions without adequate buffering at bottleneck routers

Analysis of RCP - Strengths

- Lower per-packet computations
  - To facilitate feedback computation inside router, i.e., 2 additions and 2 multiplications as against 6 additions and 3 multiplications in the case of XCP
- Smaller control header size
  - 16 bytes compared to 20 bytes in XCP, 32 bytes in JetMax, 20 bytes in MKC
- Steady-state rates achieve max-min fairness unlike XCP
- Much smaller average flow completion time (AFCT) compared to XCP and TCP

Agenda

- Introduction
- Analysis of RCP
- QI-RCP
- PIQI-RCP
- Comparison
- Wrap Up

QI-RCP

- Compared to RCP, QI-RCP decouples queue dynamics from router control equation
- Define error function $e_l(t)$ at router l as:

$$e_l(t) = 1 - \frac{y_l(t)}{\gamma_l C_l}$$

Input Traffic Rate

Scaled Link capacity

- The control equation at router l is:

$$R_l(t) = R_l(t-T)[1 + \kappa e_l(t)],$$

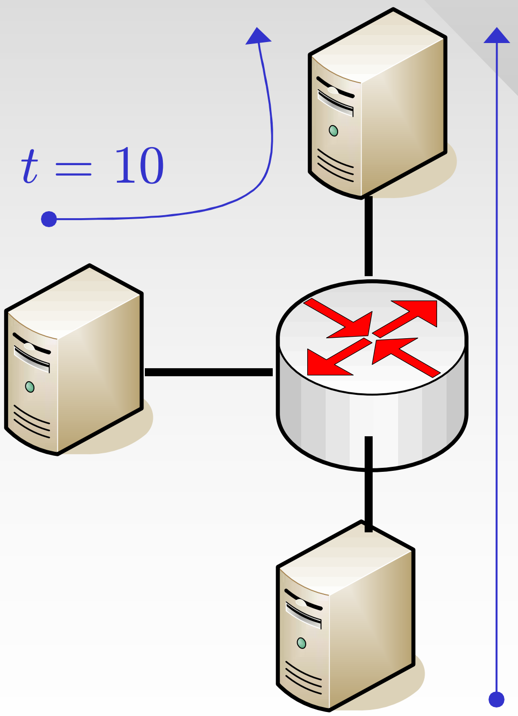
Gain Parameter

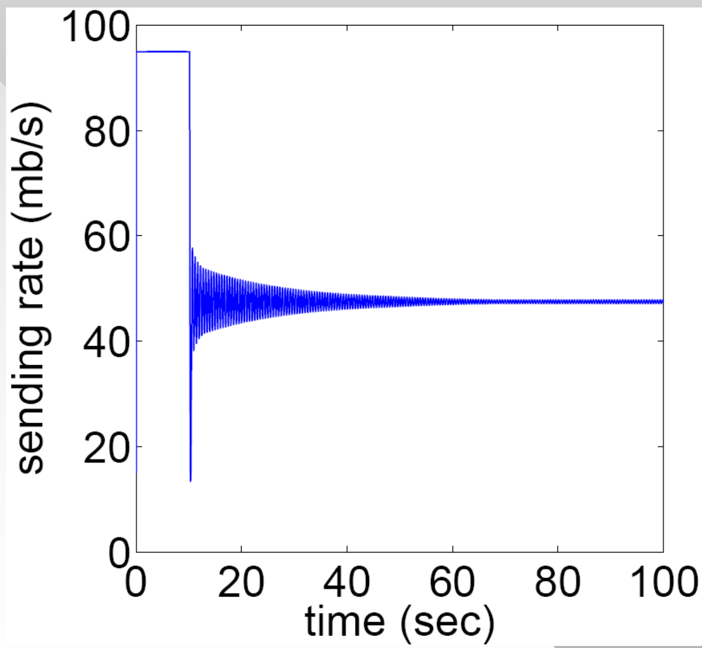
- Theorem 1: Assume N flows with heterogeneous RTTs and define $D = \max\{D_1, D_2, \ldots, D_N\}$, $D' = \lceil D/T \rceil$. The discrete version of QI-RCP is asymptotically stable if $0 < \kappa < \kappa^*$, where

$$\kappa^* = 2\sin\left(\frac{\pi}{2(2D'-1)}\right)$$

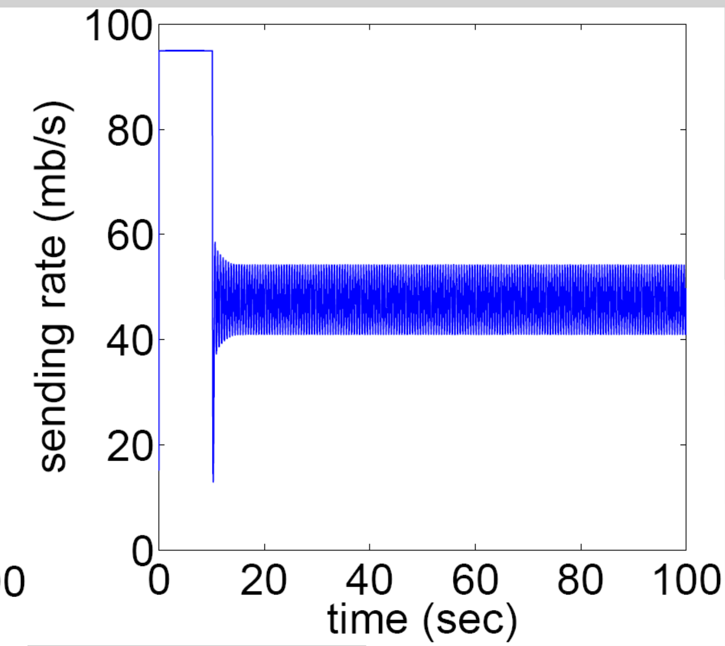
QI-RCP 1

- If flows have homogeneous RTTs (i.e., $D_i = D$), the previous condition also becomes necessary
- Verification of stability condition: $\kappa = \eta\kappa^*$, $T = 10$, $\gamma = 0.95$
  - Homogeneous delays: $D_1=D_2=122$





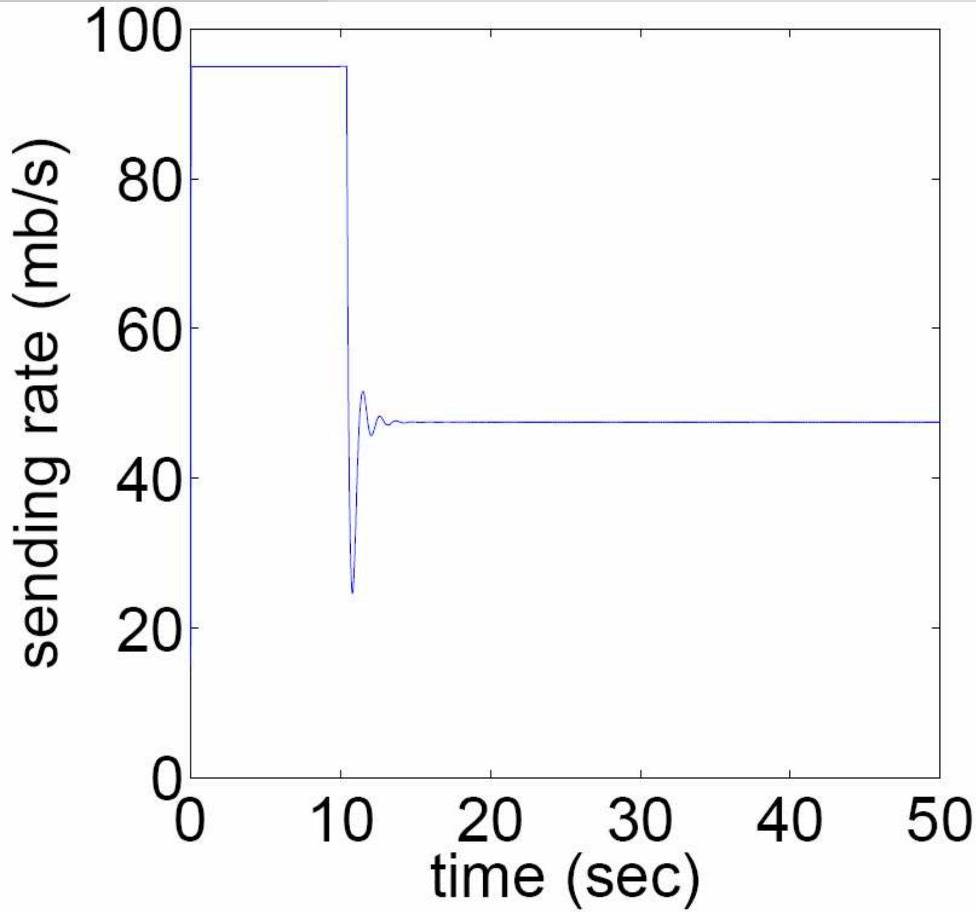
$\eta = 0.99$



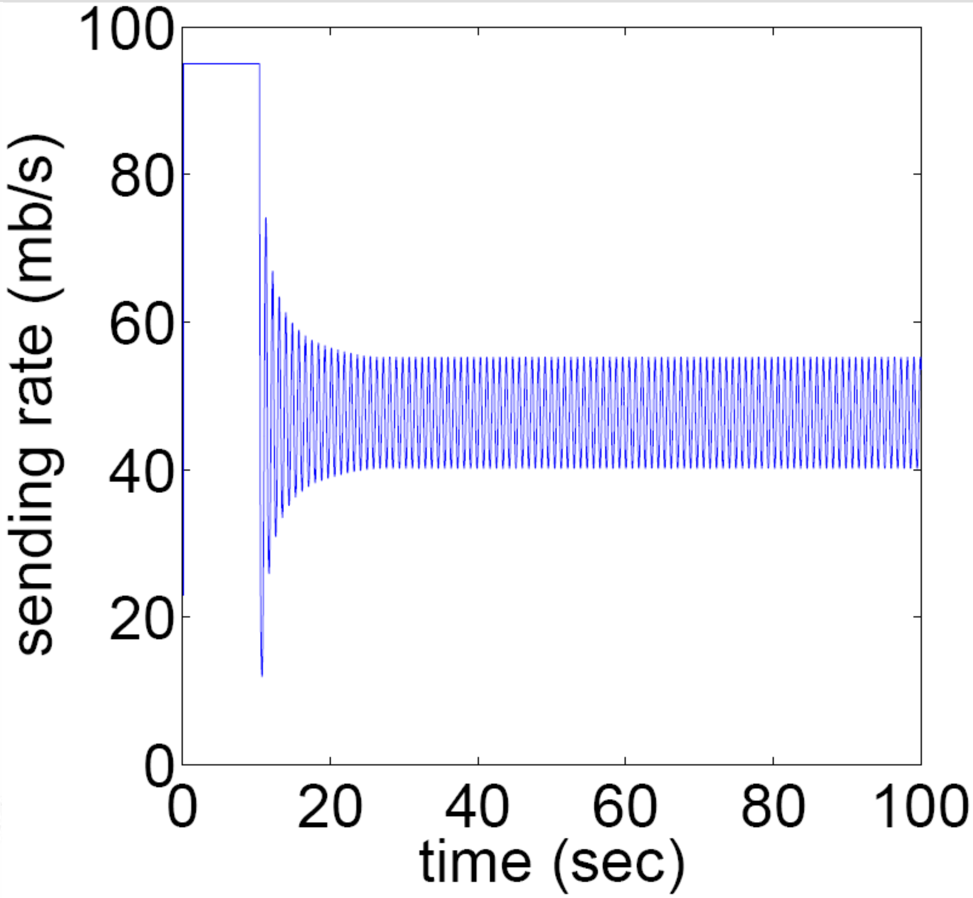
$\eta = 1.01$

QI-RCP 2

- Verification of stability condition: (cont'd)
  - Heterogeneous case: $D_1=122$, $D_2=306$



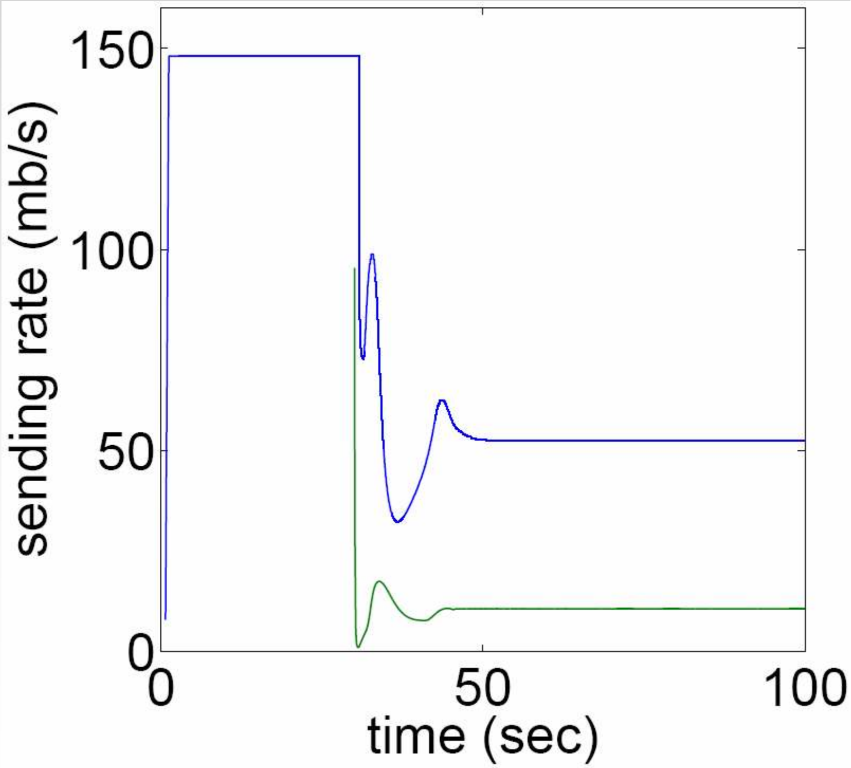
$\eta = 0.99$



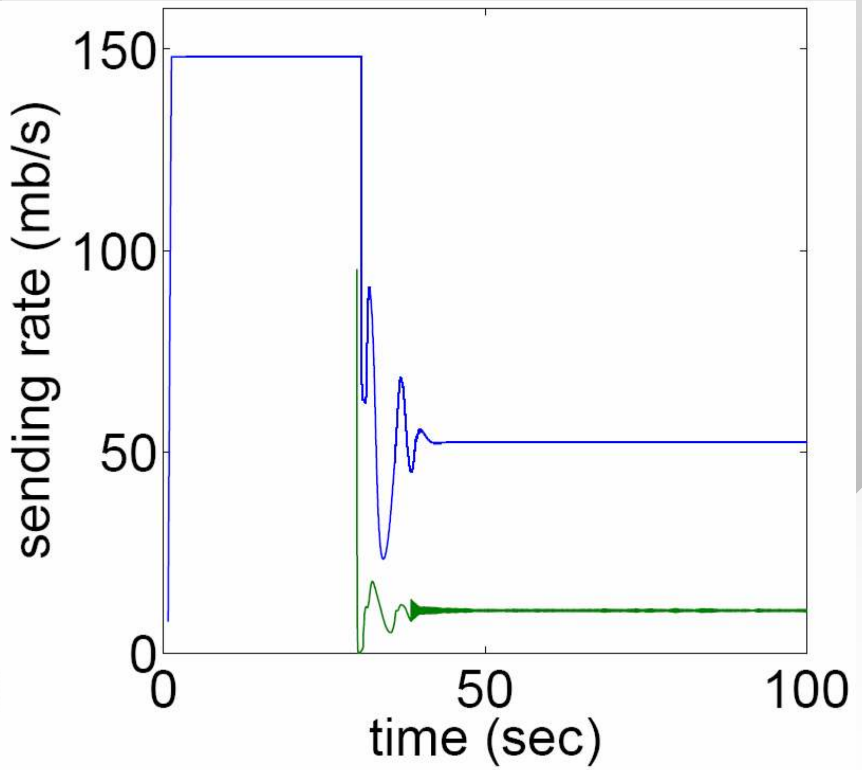
$\eta = 1.80$

QI-RCP 3

- For $T/D \approx 0,\ \kappa^* = \pi T/(2D)$. This can also be derived from the continuous version of QI-RCP
- QI-RCP is stable in topology T1 where RCP was unstable



$\eta = 0.5$



$\eta = 0.99$

Agenda

- Introduction
- Analysis of RCP
- QI-RCP
- PIQI-RCP
- Comparison
- Wrap Up

PIQI-RCP ("Picky-RCP")

- Controller at the router is a Proportional-Integral (PI) controller:

$$R_l(t) = R_l(t-T)[1 + \kappa_1 e_l(t) + \kappa_2 e_l(t-T)],$$

- At the source (end-user), define:
  - Difference between target rate and previous sending rate

$$e_i(t) = R_l(t - D_i^{\leftarrow}) - x_i(t-T)$$

  - Difference between last two consecutive feedbacks

$$\delta_i(t) = R_l(t - D_i^{\leftarrow}) - R_l(t - T - D_i^{\leftarrow})$$

- Controller at the source:

$$x_i(t) = x_i(t-T) + \tau_1 e_i(t) + \tau_2 \delta_i(t),$$

  - τ_2 affects only when router controller is in its transient state

PIQI-RCP 1

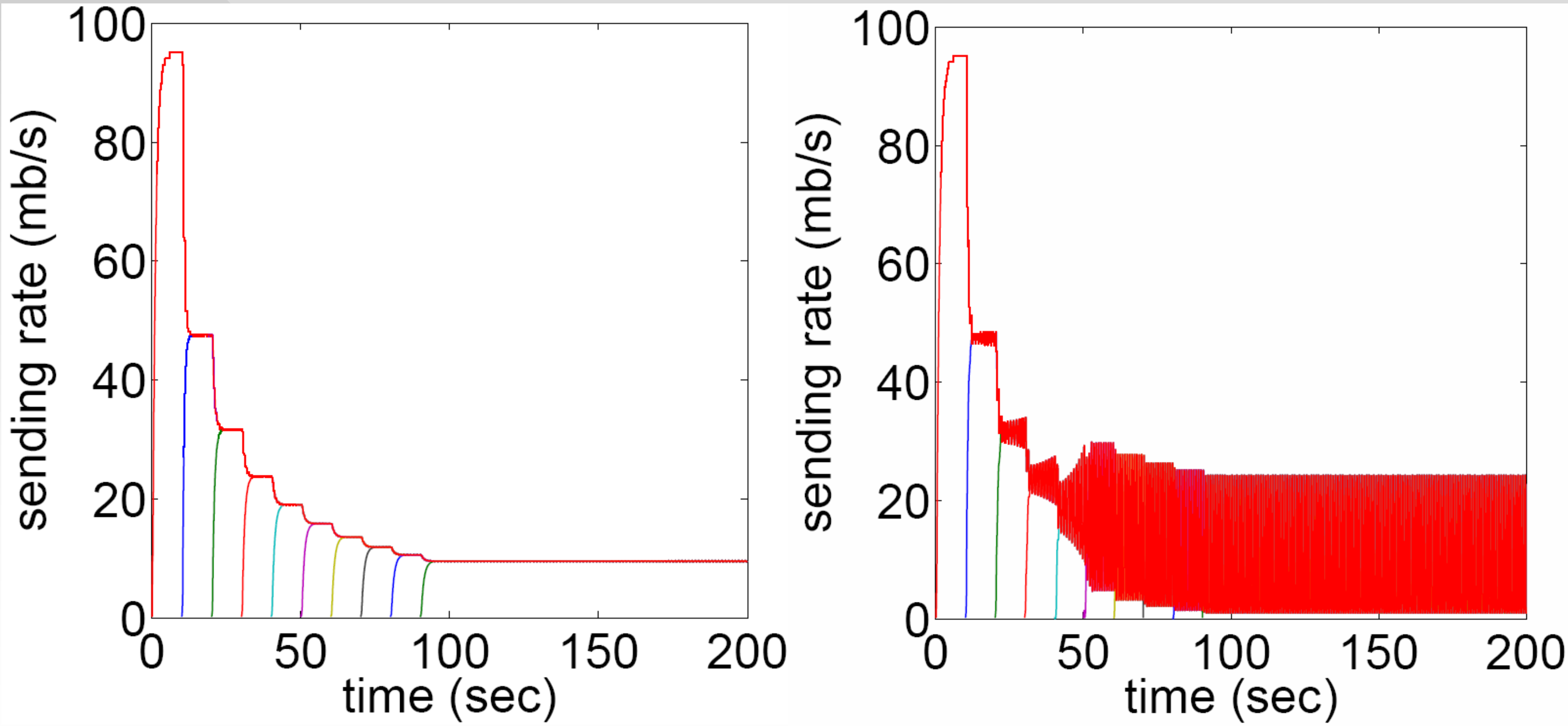
- For simplicity, we assume $\kappa_1 = \kappa_2 = \kappa$
- Theorem 2: Assume N flows with heterogeneous RTTs and define $D = \max\{D_1, D_2, \ldots, D_N\}$, $D' = \lceil D/T \rceil$. The discrete version of PIQI-RCP with sufficiently small T is locally asymptotically stable if $0 < \tau_1 < 1$, $0 < \tau_1 + 2\tau_2 < 2$ and $0 < \kappa < \kappa^*$, where

$$\kappa^* = \sin\left(\frac{\pi}{2(2D' - 1)}\right)$$

- If flows have homogeneous RTTs (i.e., $D_i = D$), the previous condition also becomes necessary
- Stability condition for sufficiently small T, τ_1, and τ_2 is half of that in QI-RCP

PIQI-RCP 2

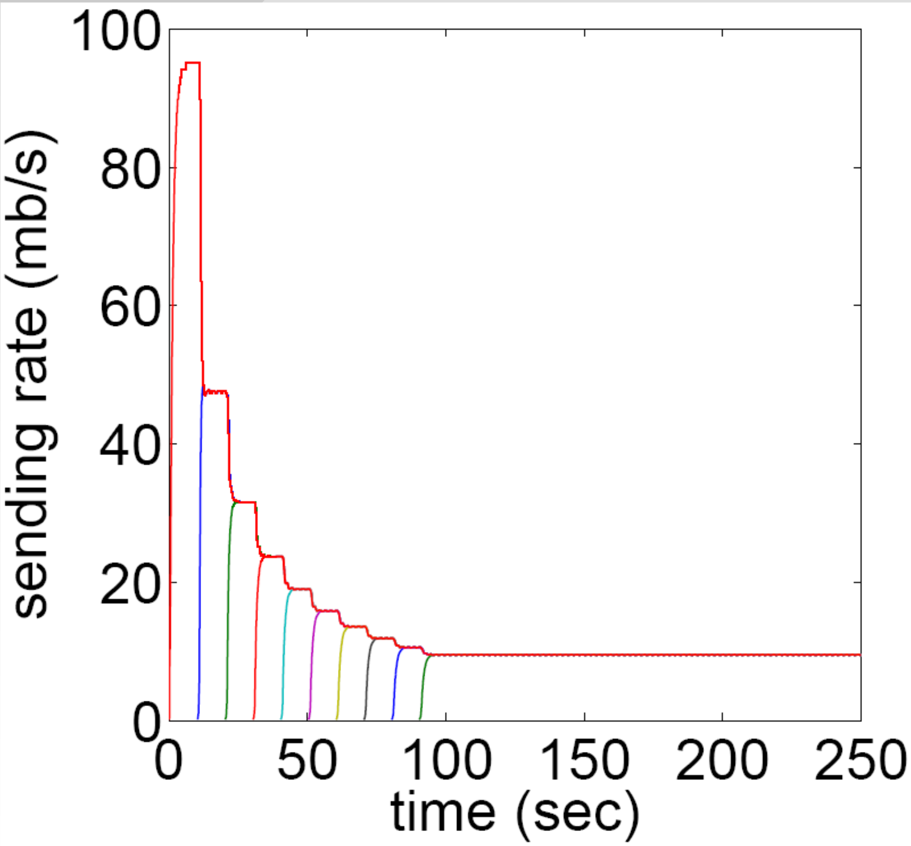
- Verification of stability condition: $\kappa = \eta\kappa^*$, $T = 10$, $\gamma = 0.95$, $\tau_1 = 0.005$, $\tau_2 = 0.5$
  - Homogeneous case: $D_1 = D_2 = \ldots = D_{10} = 120$



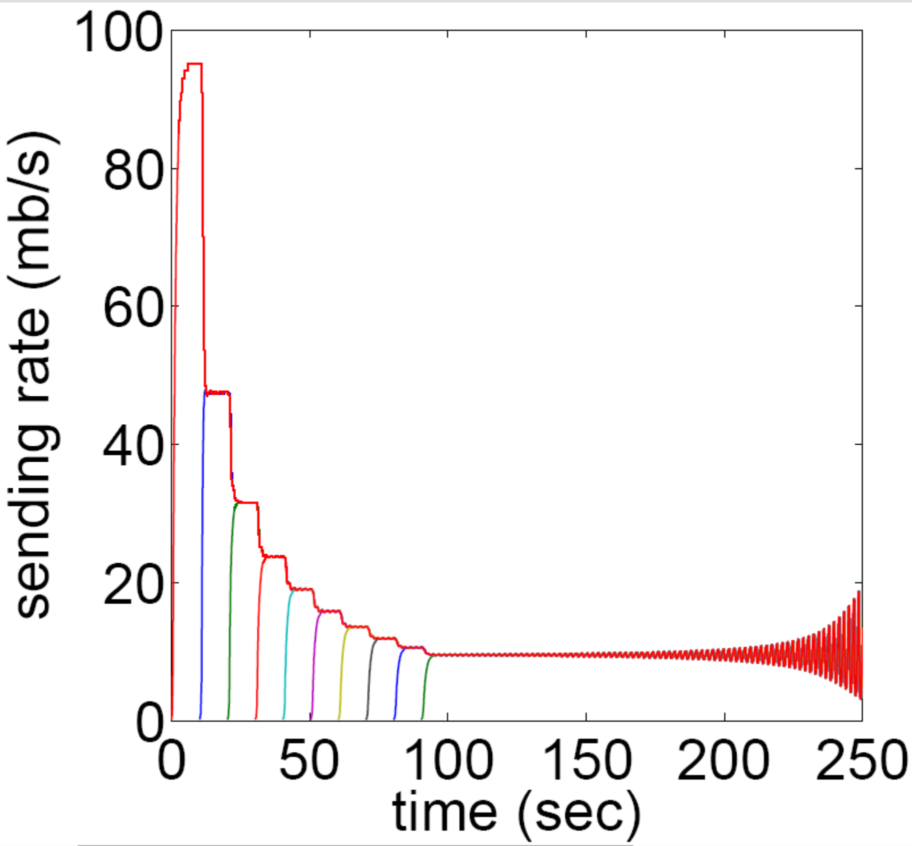
$\eta = 0.99$ $\eta = 1.01$

PIQI-RCP 3

- Verification of stability condition: (cont'd)
  - Heterogeneous case: $D_1=120,\ D_2=\ldots=D_{10}=300$



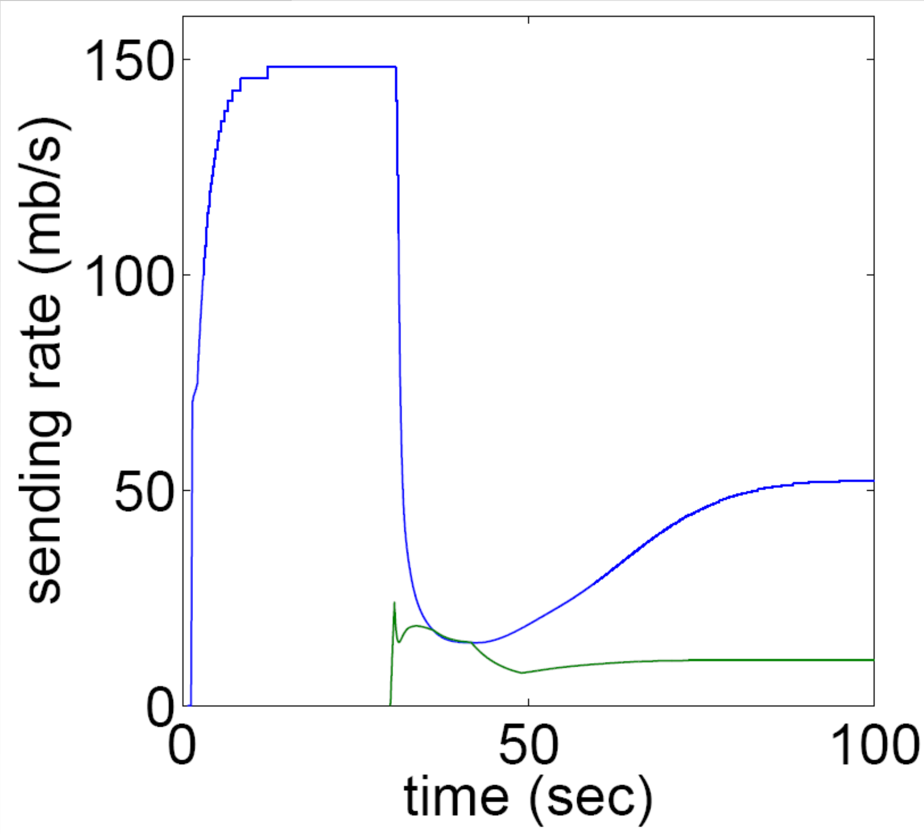
$\eta = 0.99$



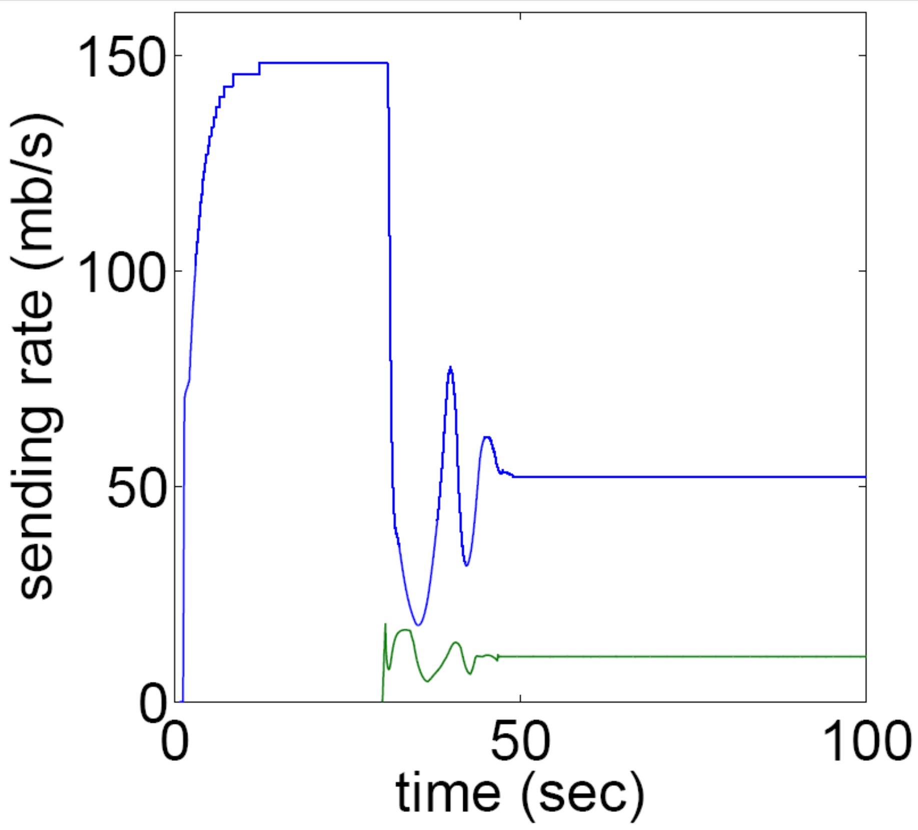
$\eta = 1.01$

PIQI-RCP 4

- PIQI-RCP is stable in topology T1 where RCP was unstable



$\eta = 0.50$



$\eta = 0.99$

Agenda

- Introduction
- Analysis of RCP
- QI-RCP
- PIQI-RCP
- Comparison
- Wrap Up

Comparison

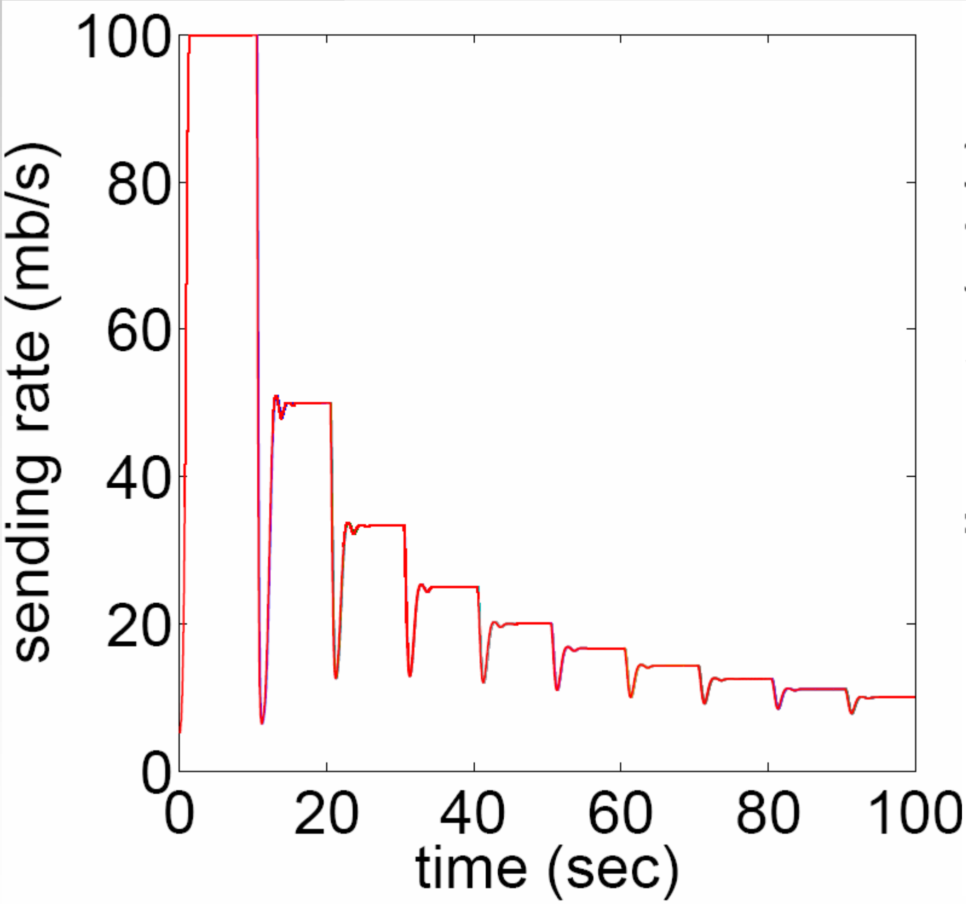
- We next compare RCP and PIQI-RCP using NS-2 simulations
- To prevent computing sine function inside routers, the upper bound κ^* is approximated as κ_*

$$\kappa < \kappa_* = \frac{T}{2(T+D)} \leq \kappa^*,$$

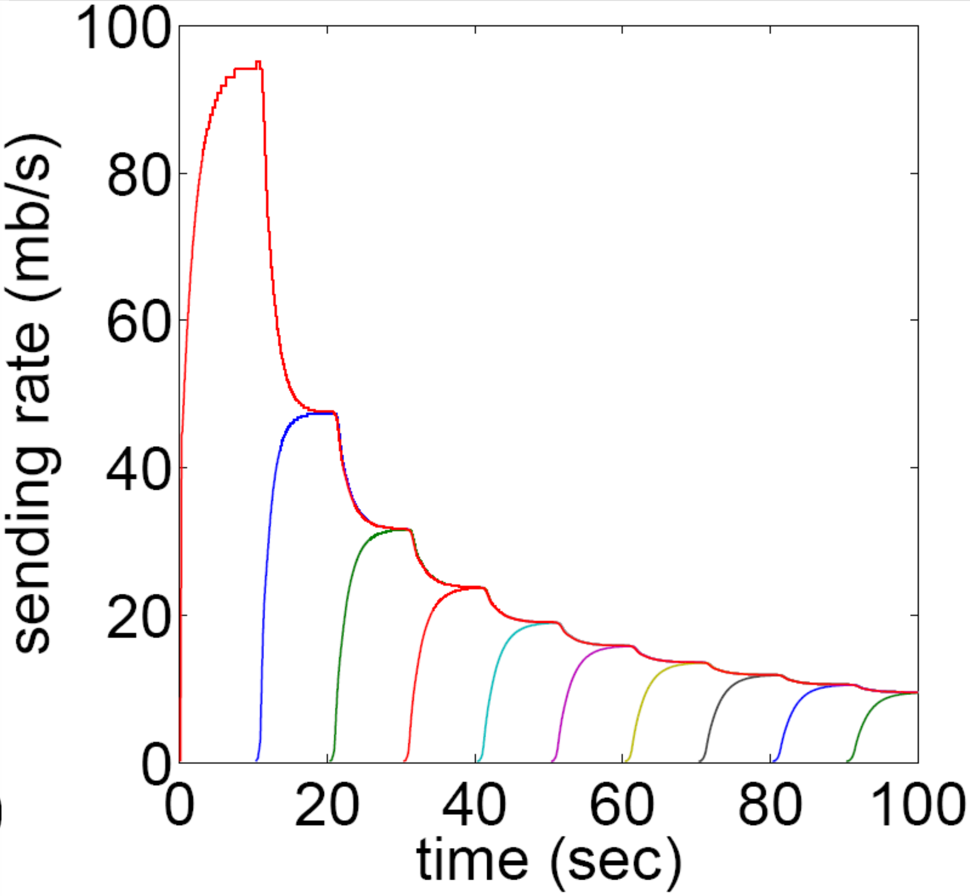
- For RCP, we set $\alpha = 0.4$, $\beta = 1$, $T = 10$
- For PIQI-RCP, we set $\kappa = 0.95\kappa_*$, $T = 10$, $\gamma = 0.95$, $\tau_1 = 0.005$, $\tau_2 = 0.5$

Comparison 1

- Single Bottleneck Topology: $D_1 = 120,\ D_2 = \ldots = D_{10} = 300$
  - Sending Rate:



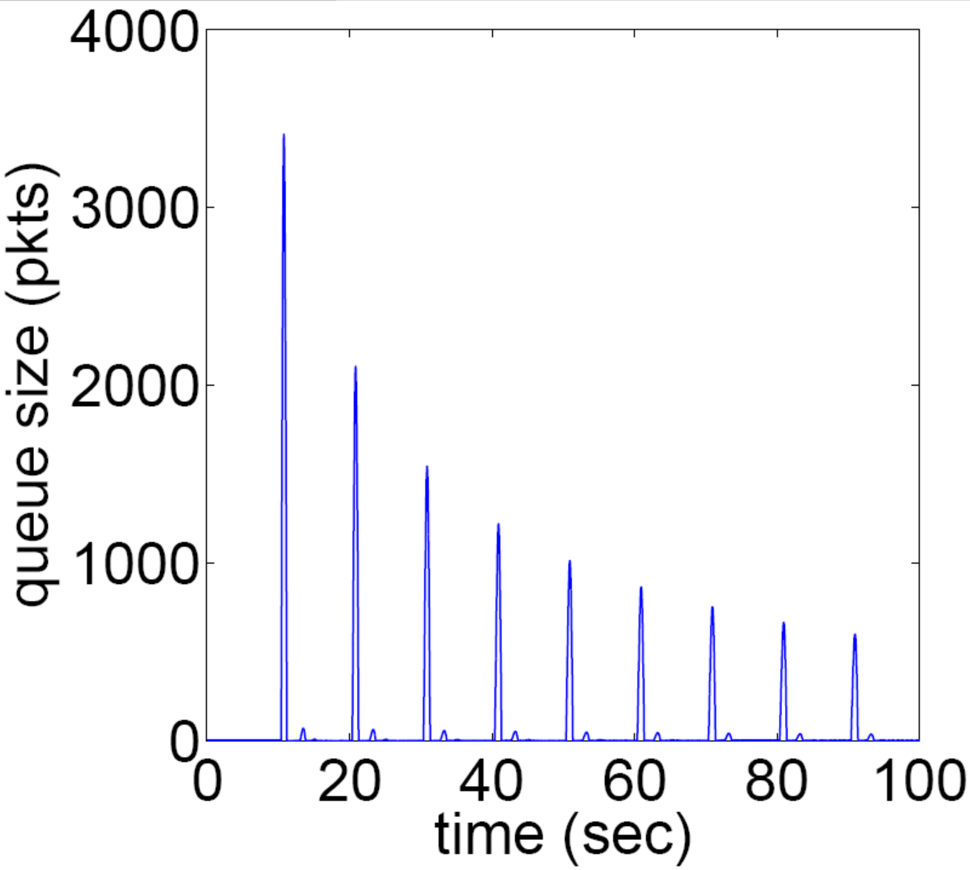
RCP



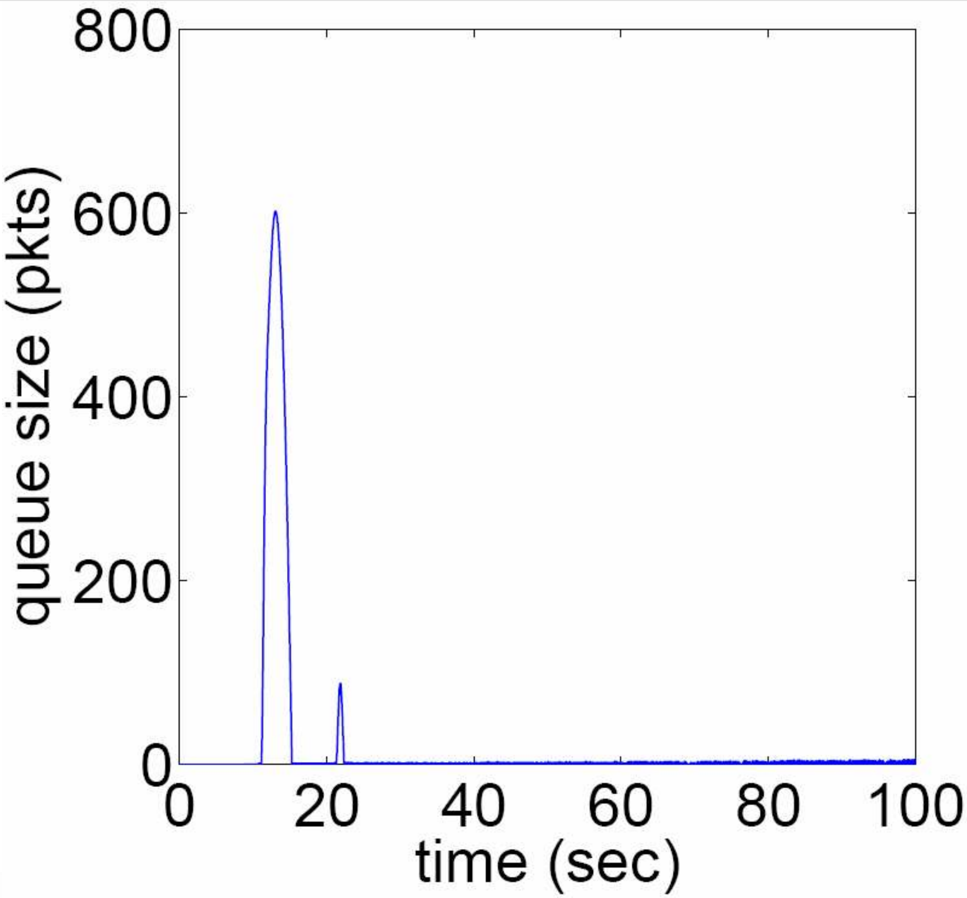
PIQI-RCP

Comparison 2

- Single Bottleneck Topology: (cont'd)
  - Queue Size:



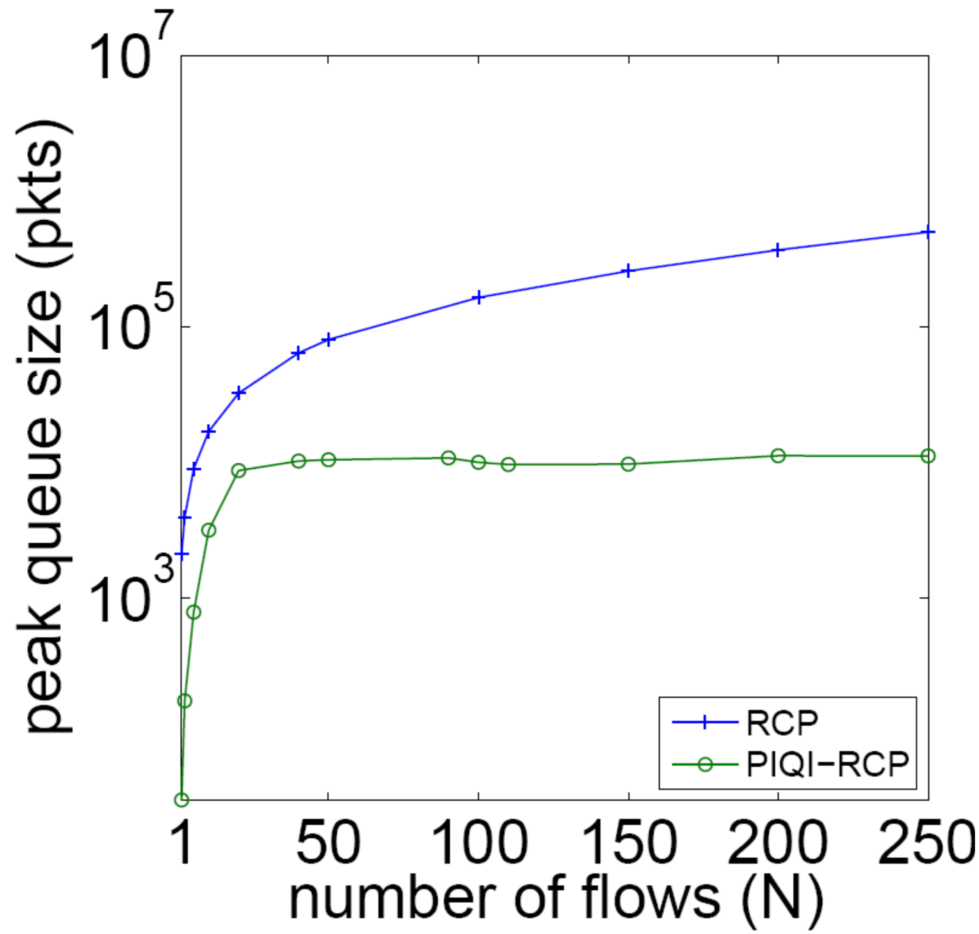
RCP



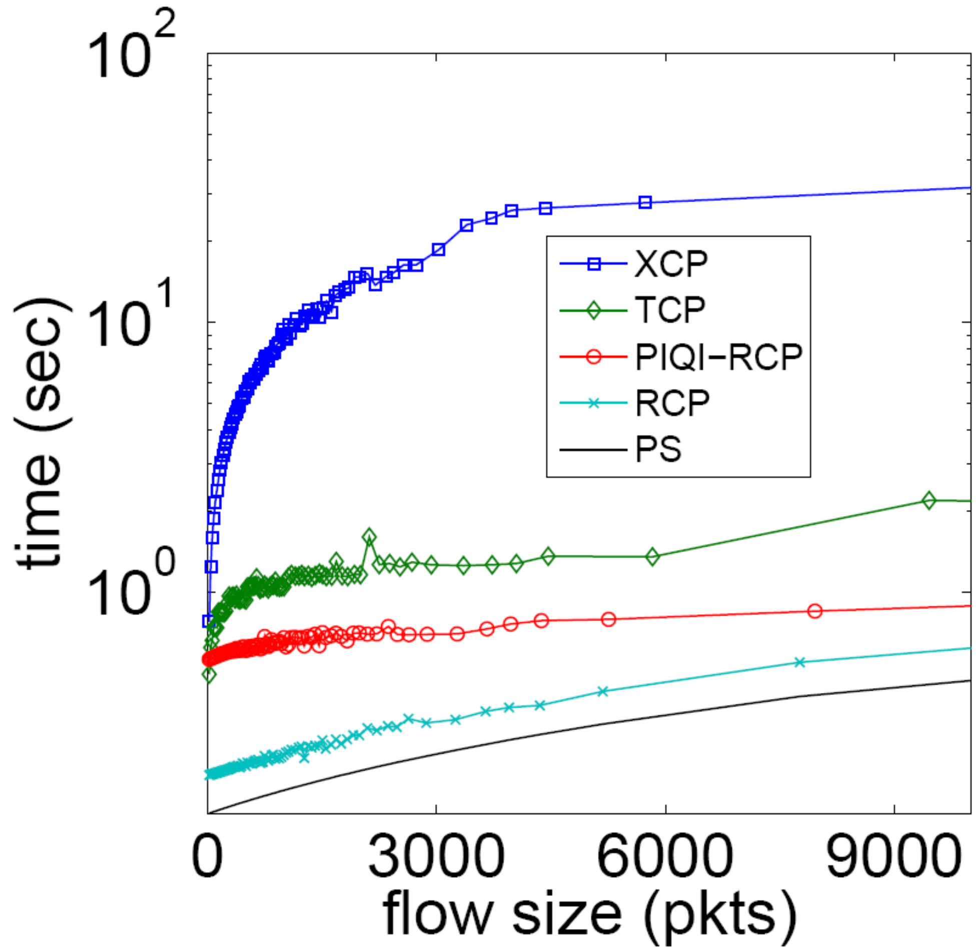
PIQI-RCP

Comparison 3

- Single Bottleneck Topology: (cont'd)
  - Peak Queue Size and AFCT:



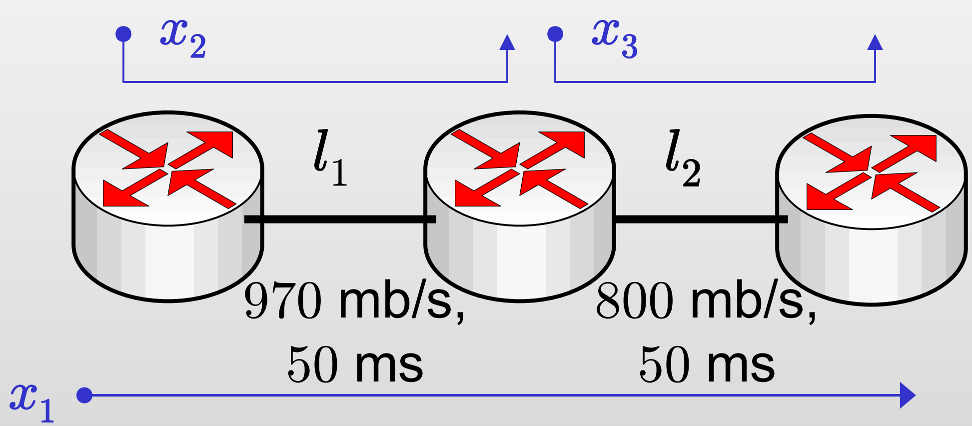
Peak Queue Size

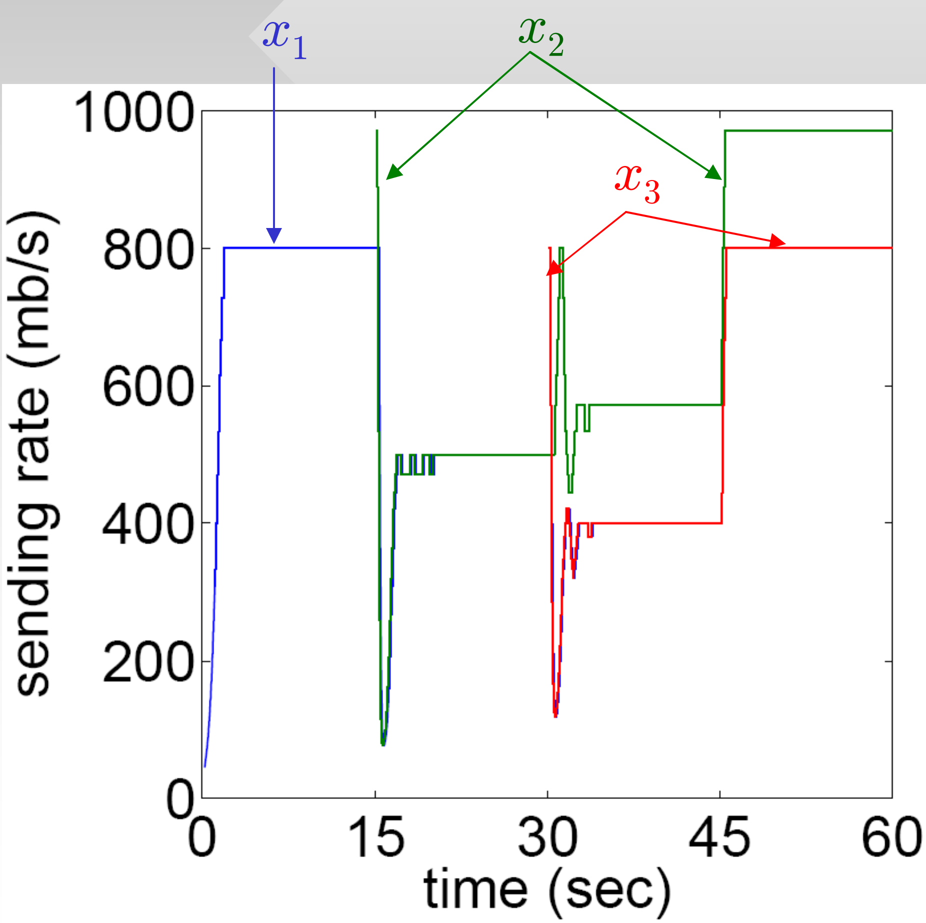


AFCT

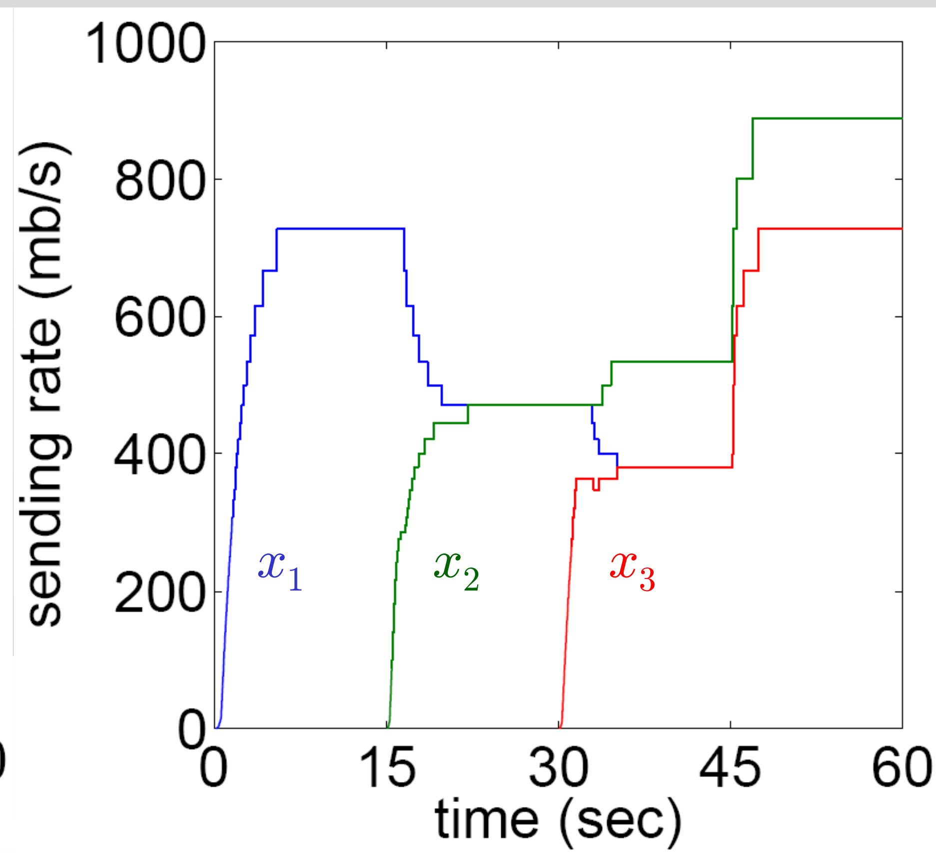
Comparison 4

- Multi-Bottleneck Topology:





RCP



PIQI-RCP

Comparison - Linux

- Implemented both RCP and PIQI-RCP inside Linux kernel for further comparison using real systems and gigabit network
- As observed in NS-2 simulations, Linux experimental results also indicate better performance of PIQI-RCP as compared to RCP
  - In both single- and multi-link topologies
  - With abrupt changes in traffic demands
  - Using both long and mice flows
- Future work includes comparing PIQI-RCP with other explicit congestion control methods

Wrap Up

- Stability analysis in the presence of heterogeneous delays is of fundamental importance in the design of congestion control
- Use of average RTT in control equation without proper analysis and flow identification (i.e., responsive or unresponsive) may not be appropriate
- PIQI-RCP mitigates drawbacks of RCP with slight tradeoff in link utilization (γ) and AFCT
- More in the paper:
  - Proofs of theorems
  - Results from Linux experiments conducted in Emulab

Thank You!