One More Bit Is Enough

Yong Xia, RPI

Lakshmi Subramanian, UCB

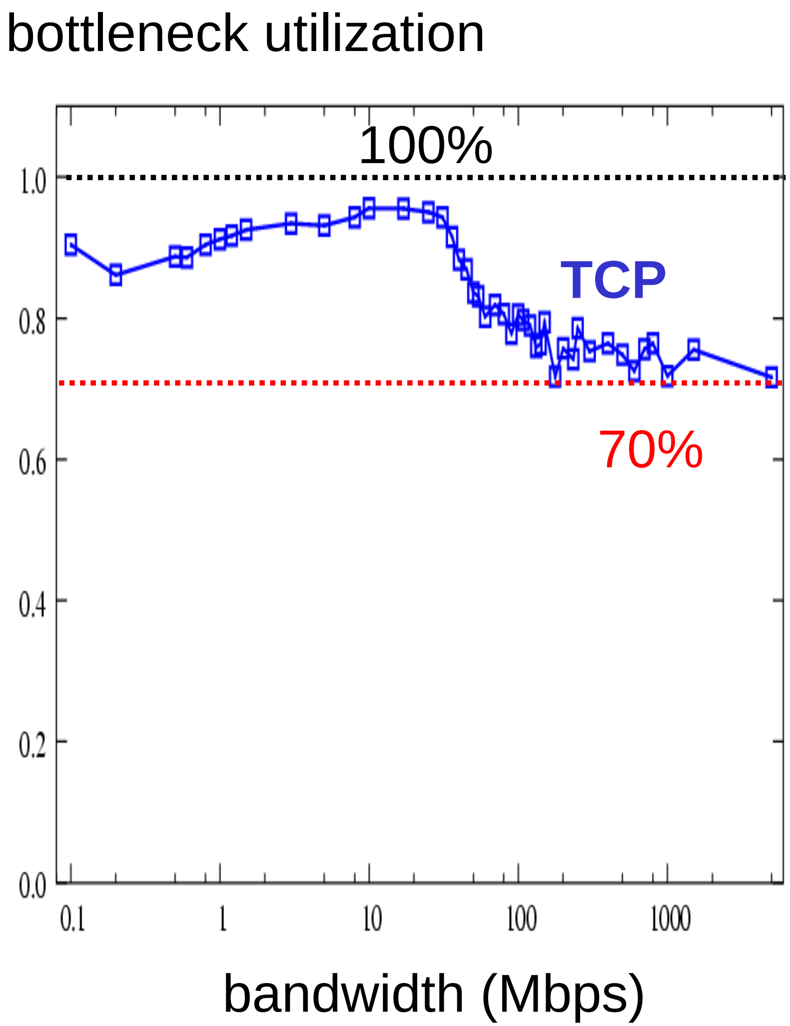
Ion Stoica, UCB

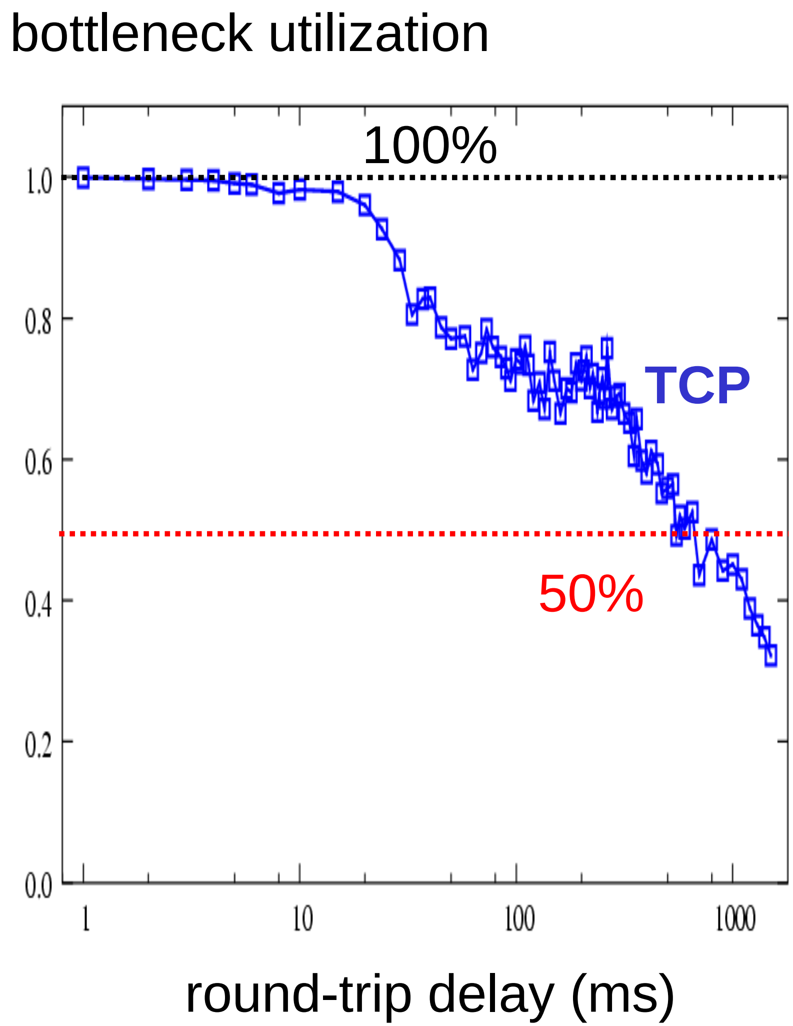
Shiv Kalyanaraman, RPI

SIGCOMM'05, Philadelphia, PA

08 / 23 / 2005

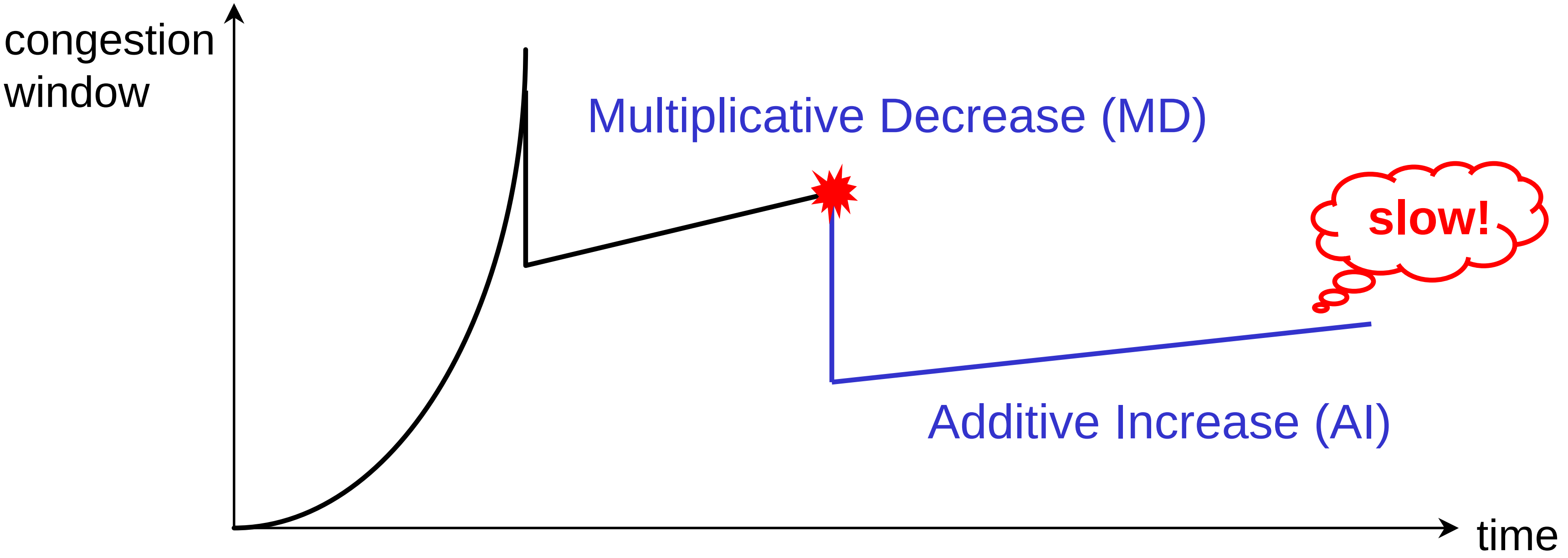
Motivation #1: TCP doesn't work well in high b/w or delay





Motivation #1: Why TCP does not scale?

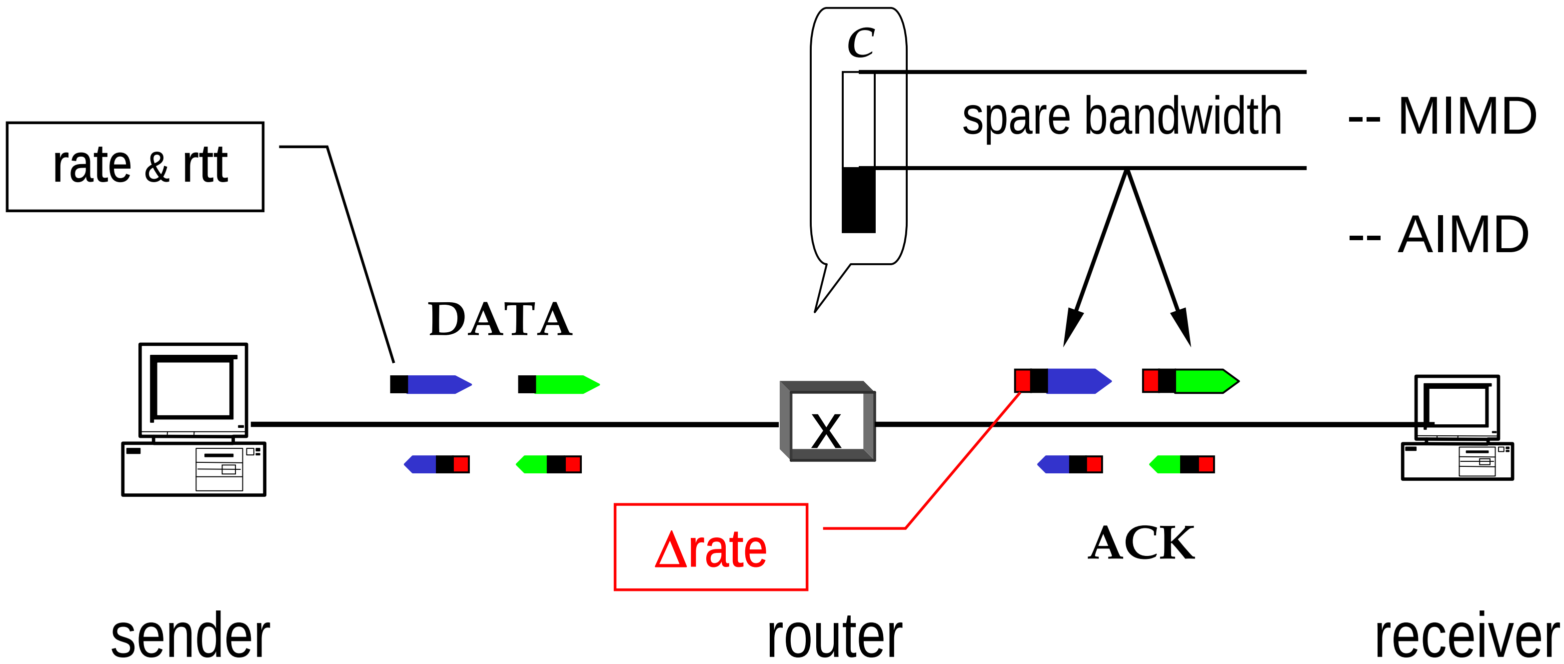
- TCP uses binary congestion signals, such as loss or one-bit Explicit Congestion Notification (ECN)



- AI with a fixed step-size can be very slow for large bandwidth

Motivation #2: XCP scales

- XCP decouples efficiency control and fairness control



- But, XCP needs multiple bits (128 bits in its current IETF draft) to carry the congestion-related information from/to network

Goal

Design a TCP-like scheme that:

- requires a small amount of congestion information (e.g., 2 bits)
- scales across a wide range of network scenarios

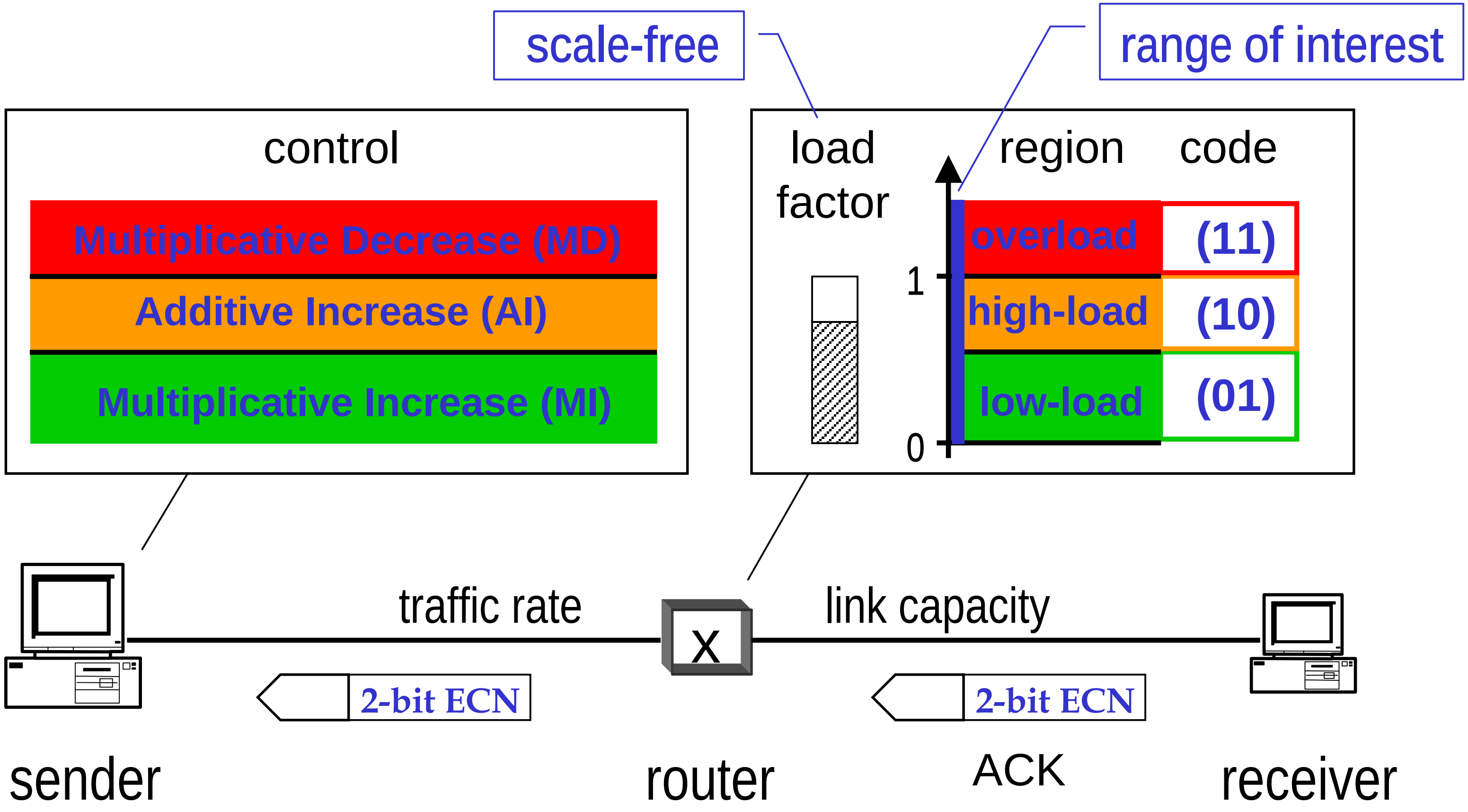
Key Observation

Fairness is not critical in low-utilization region

- Use Multiplicative Increase (MI) for fast convergence onto efficiency in this region
- Handle fairness in high-utilization region

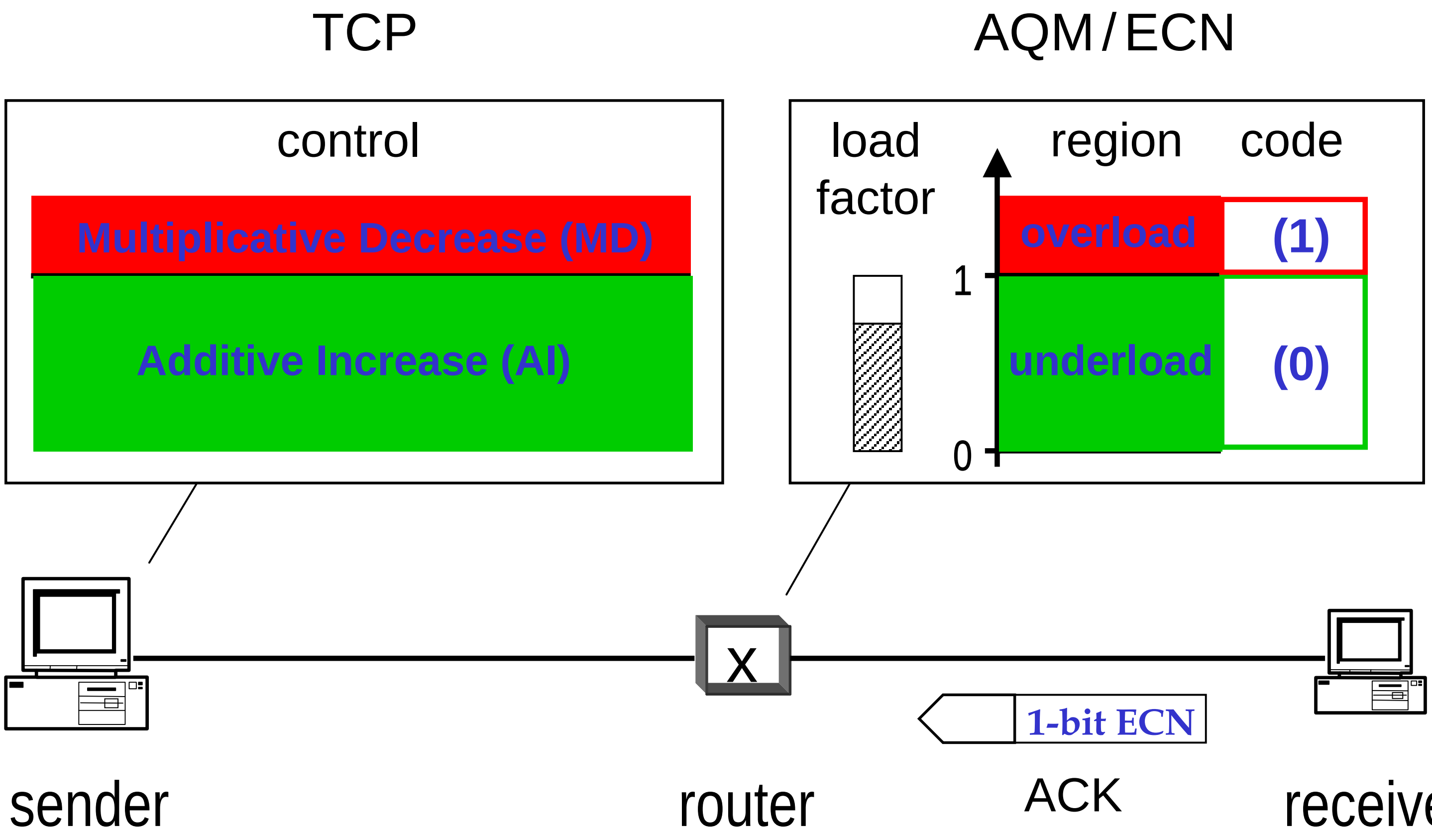
Variable-structure congestion Control Protocol (VCP)

- Routers signal the level of congestion
- End-hosts adapt the control algorithm accordingly



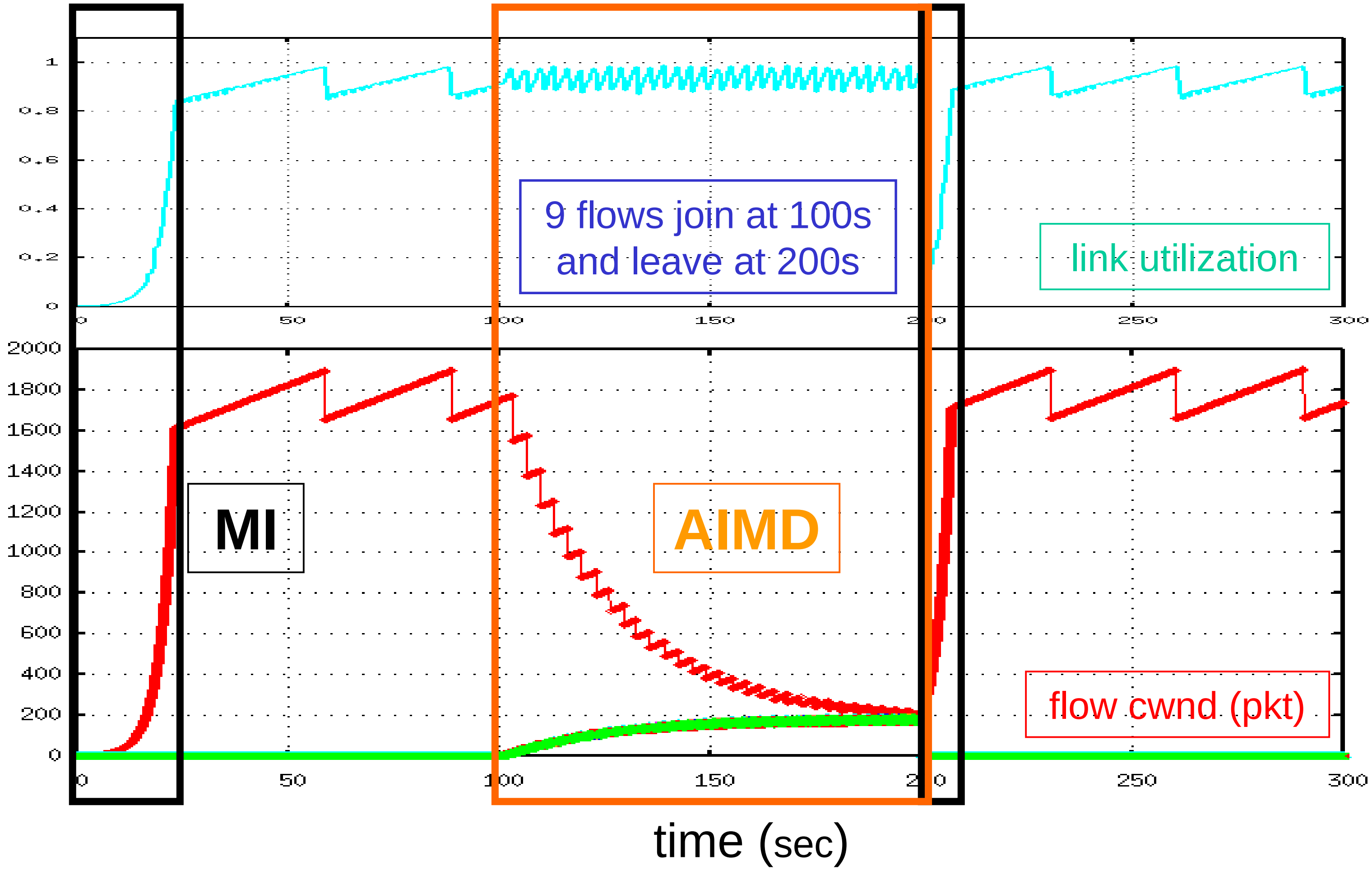
VCP vs. ECN

- ECN doesn't differentiate between low-load and high-load regions

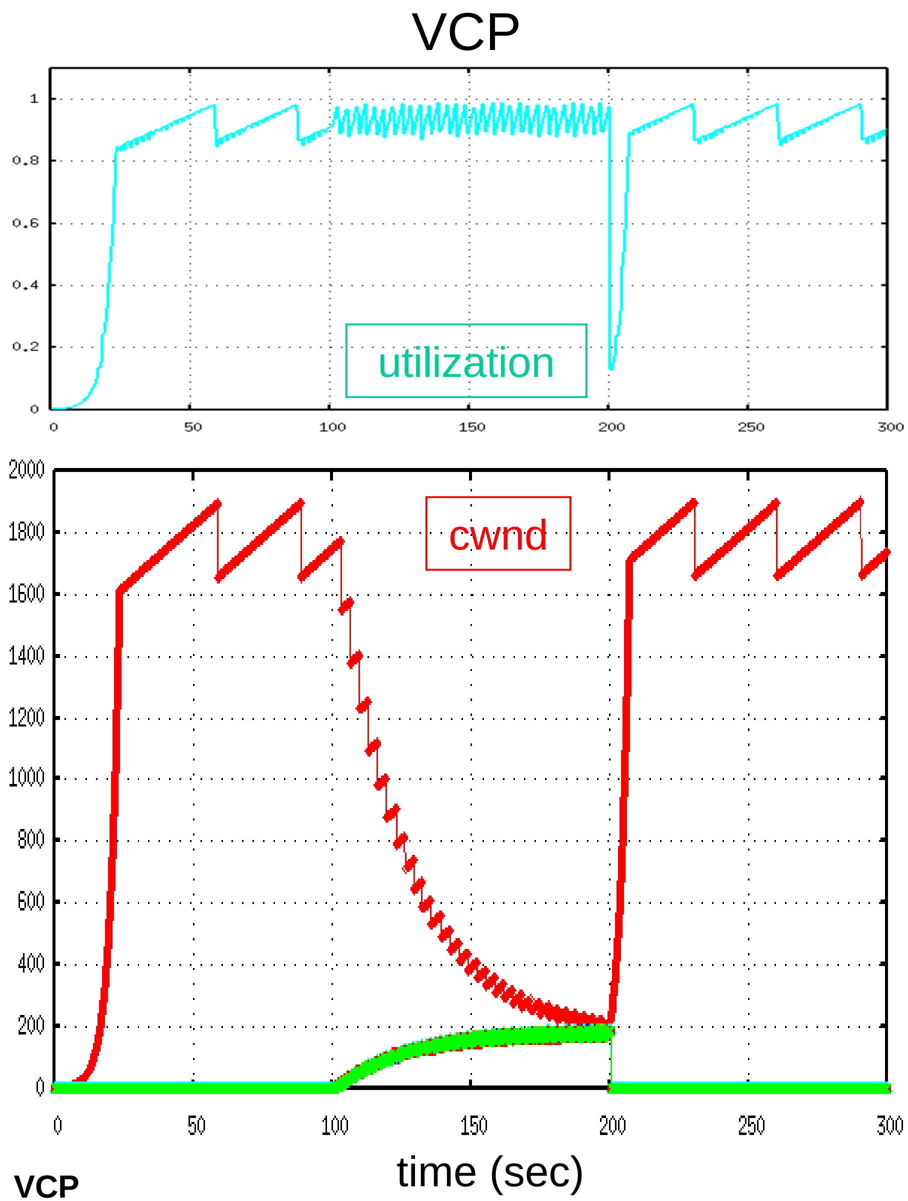


An illustration example

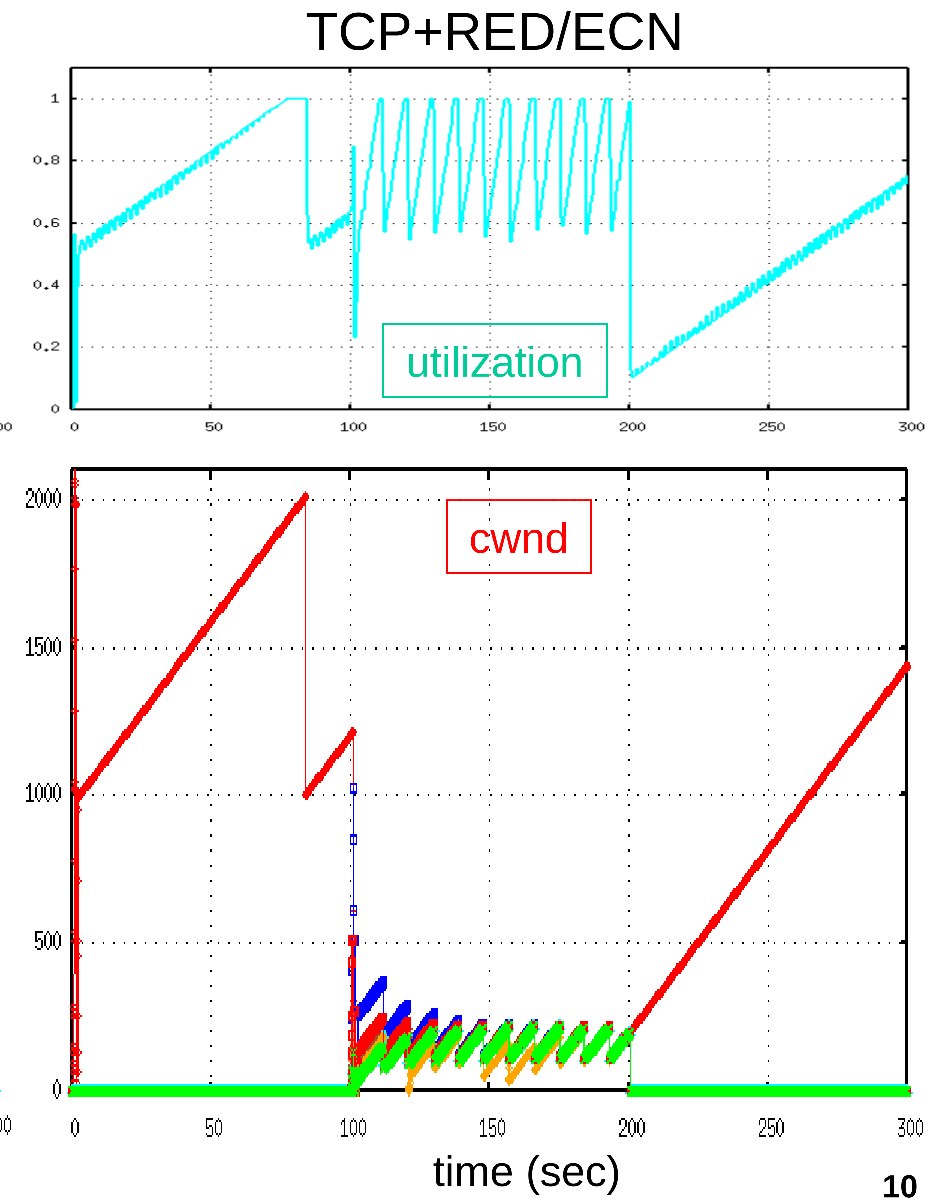
- MI tracks available bandwidth exponentially fast
- After high utilization is attained, AIMD provides fairness



VCP vs. ECN

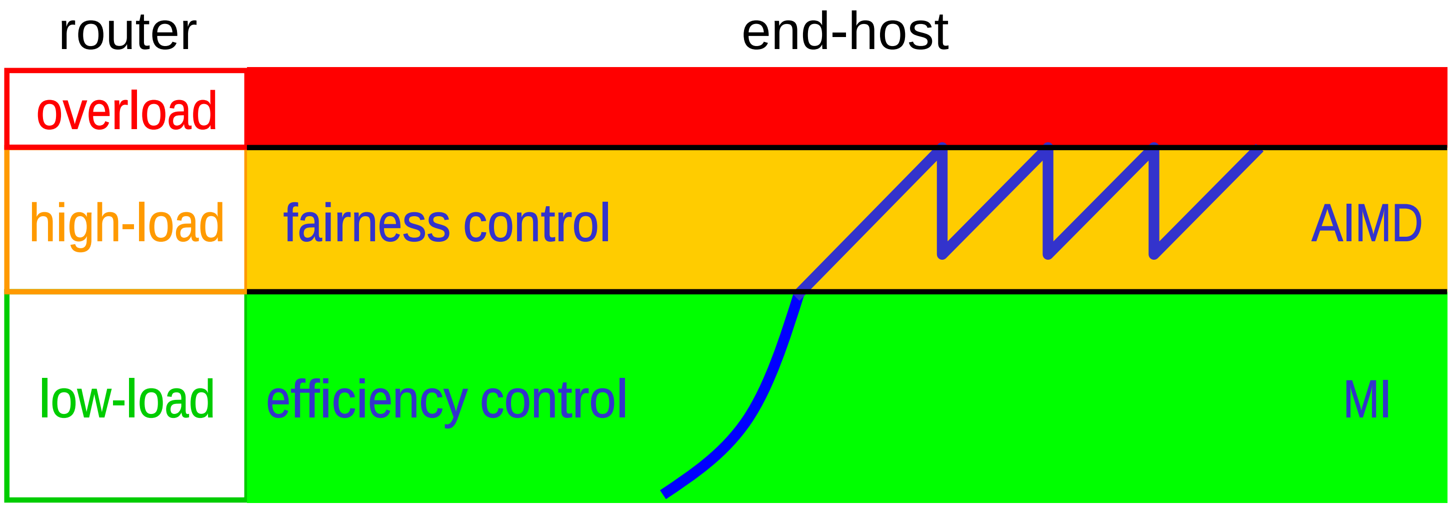


VCP
1
0.8
0.6
0.4
0.2
0
0
50
100
150
200
250
300
utilization
2000
1800
1600
1400
1200
1000
800
600
400
200
0
cwnd
time (sec)



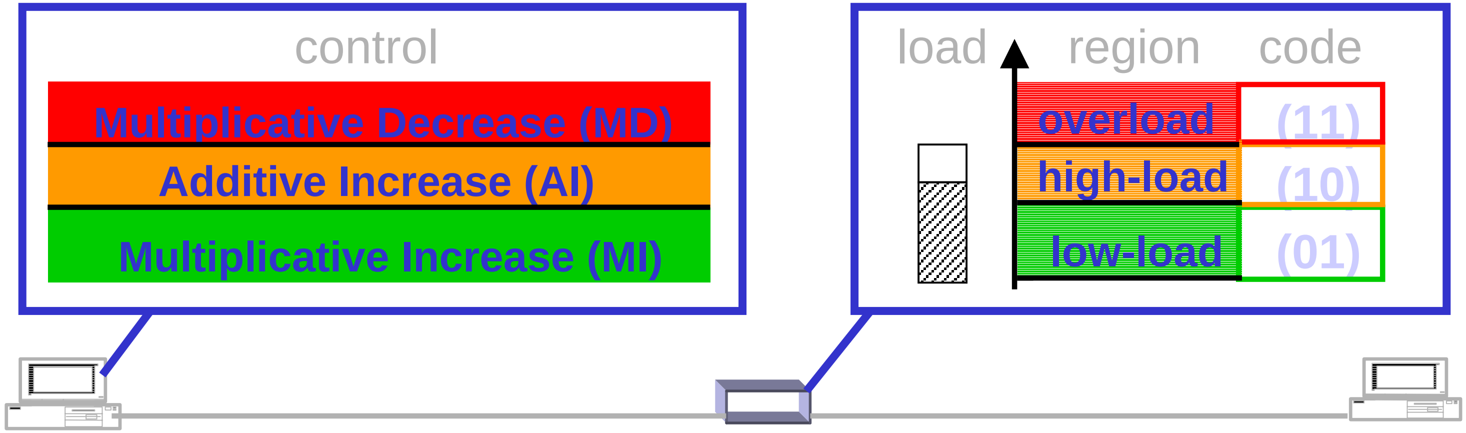
TCP+RED/ECN
1
0.8
0.6
0.4
0.2
0
0
50
100
150
200
250
300
utilization
2000
1500
1000
500
0
cwnd
time (sec)

VCP key ideas and properties



- Use network link load factor as the congestion signal
- Decouple efficiency and fairness controls in different load regions
- Achieve high efficiency, low loss, and small queue
- Fairness model is similar to TCP:
  - Long flows get lower bandwidth than in XCP (proportional vs. max-min fairness)
  - Fairness convergence much slower than XCP (solvable with even more, e.g., 8 bits)

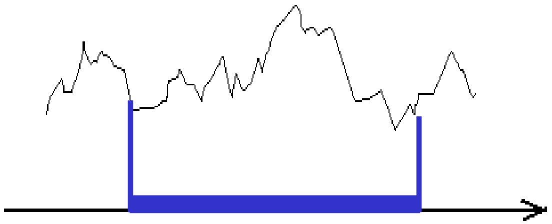
Major design issues



- At the router
  - How to measure and encode the load factor?
- At the end-host
  - When to switch from MI to AI?
  - What MI / AI / MD parameters to use?
  - How to handle heterogeneous RTTs?

Design issue #1: measuring and encoding load factor

- Calculate the link load factor ρ



t_ρ = 200ms >≈ most rtt

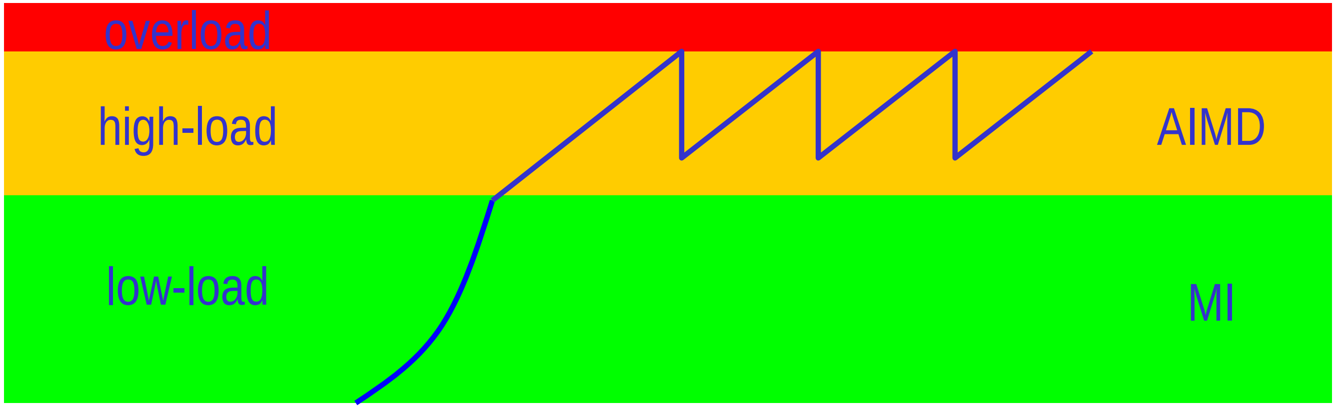
$$load\_factor = \frac{demand}{capacity}$$

$$= \frac{arrival\_traffic + queue\_size}{link\_bandwidth * t_\rho}$$

- The load factor is <u>quantized</u> and encoded into the two ECN bits

information loss → affects fairness

Design issue #2: setting MI/AI/MD parameters (ξ, α, β)



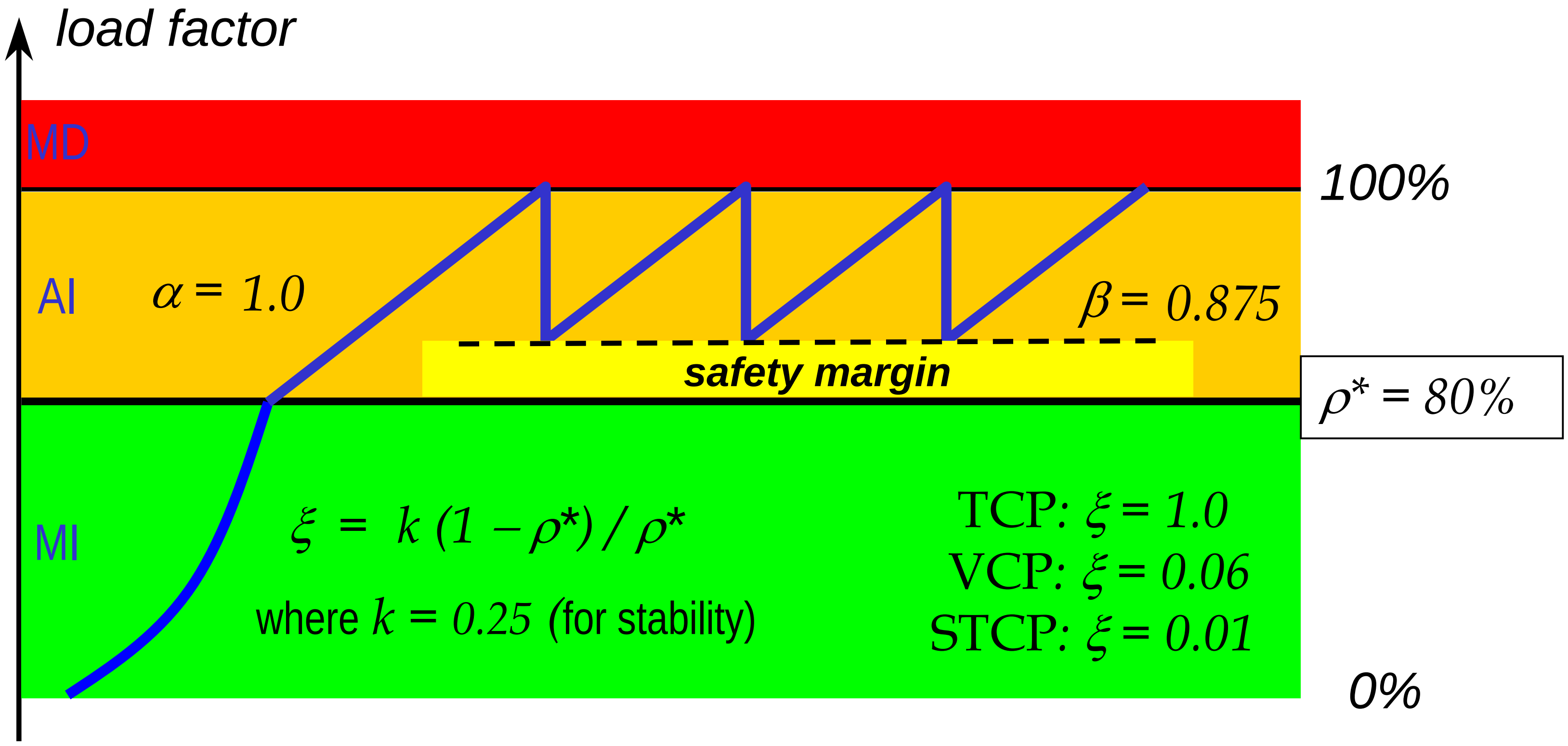
$$\text{MI}: \quad cwnd(t + rtt) = cwnd(t) \times (1 + \xi)$$

$$\text{AI}: \quad cwnd(t + rtt) = cwnd(t) + \alpha$$

$$\text{MD}: \quad cwnd(t + \delta t) = cwnd(t) \times \beta$$

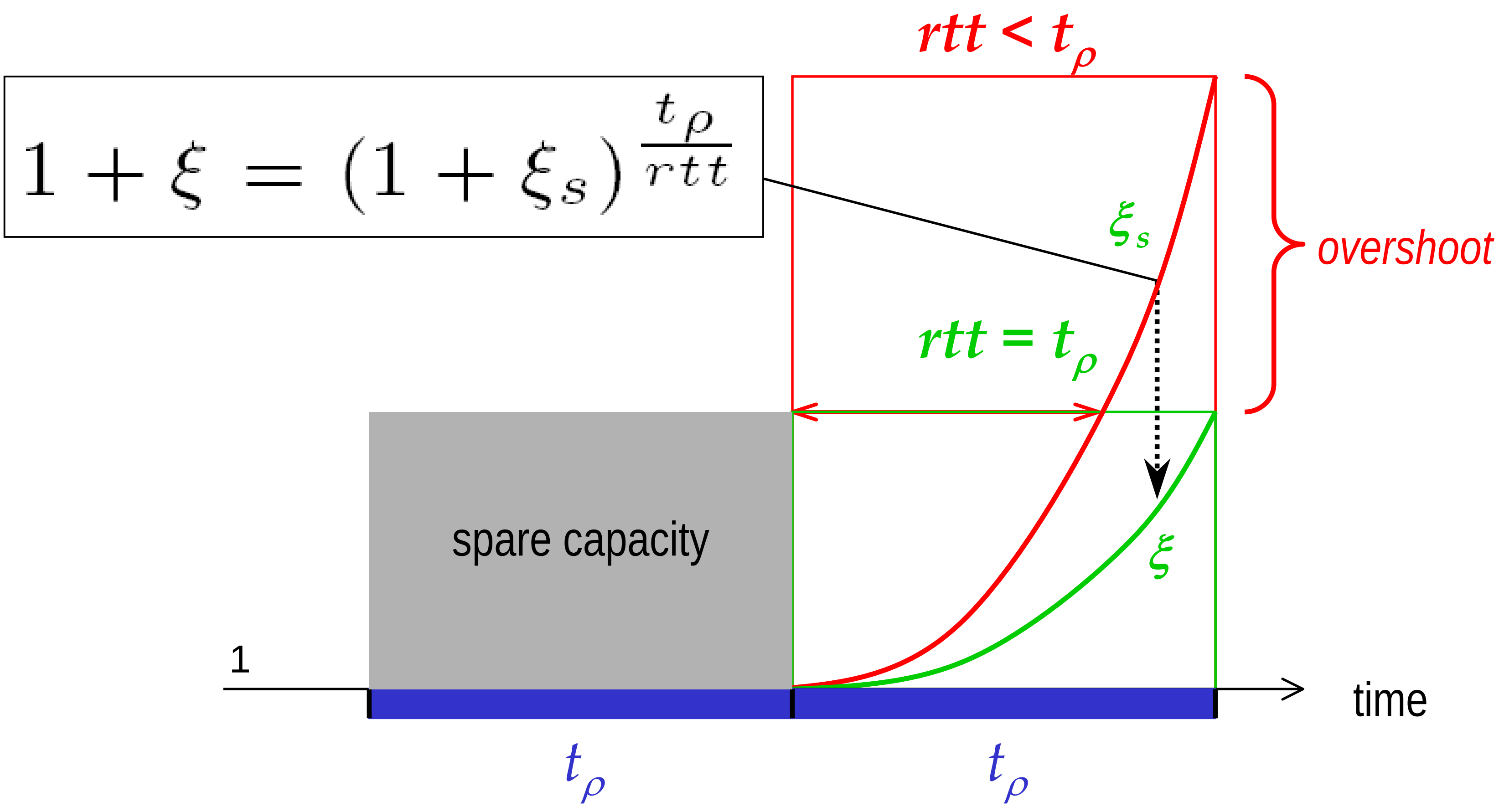
Design issue #2: setting MI/AI/MD parameters (ξ, α, β)

- Q: load factor transition point ρ^* for MI → AI?



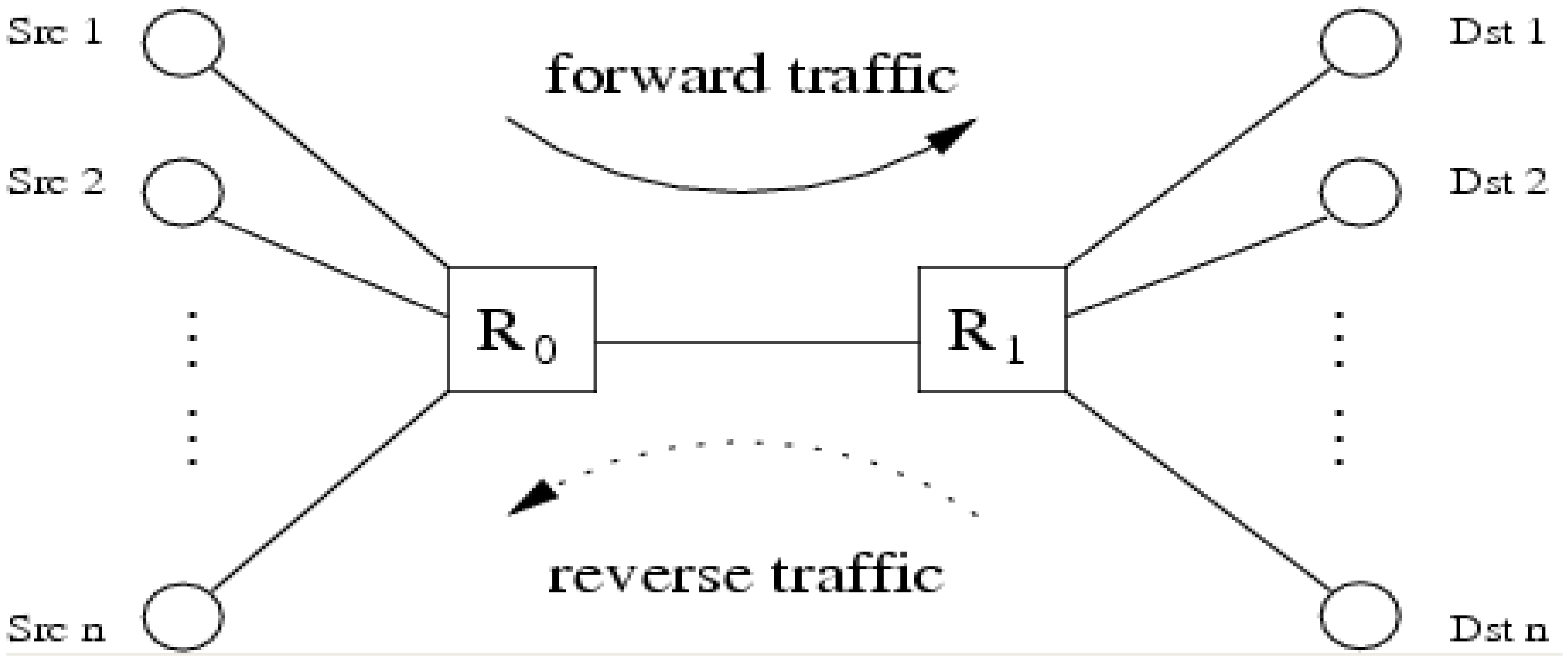
Design issue #3: Handling RTT heterogeneity for MI/AI

- Scale ξ to prevent MI from overshooting capacity when RTT is small



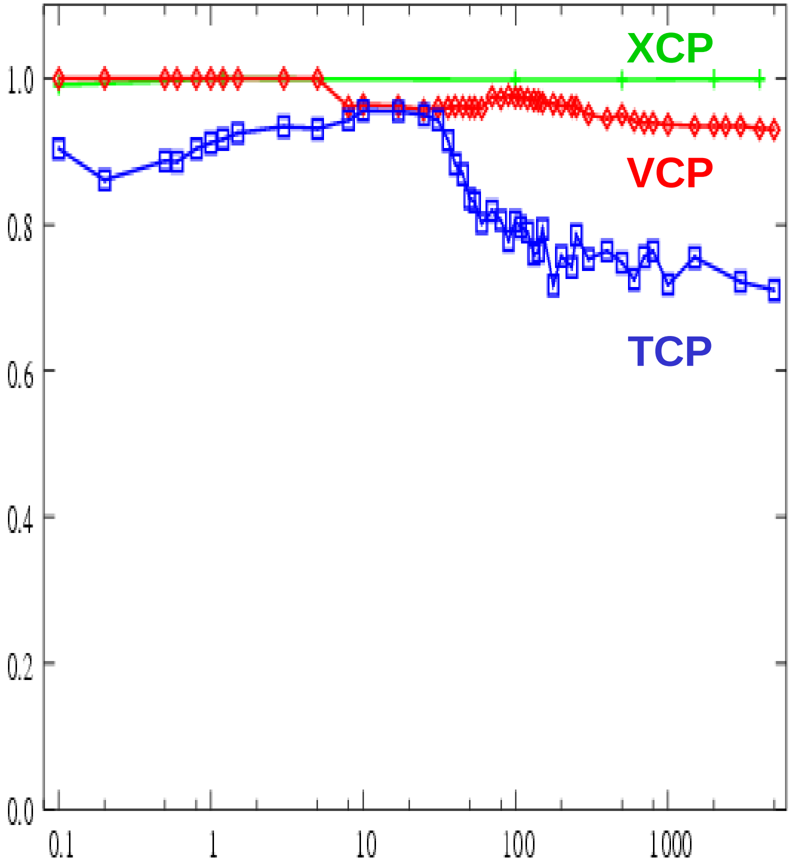
VCP scales across b/w, rtt, num flows

- Evaluation using extensive ns2 simulations
- 150Mbps, 80ms, 50 forward flows and 50 reverse flows



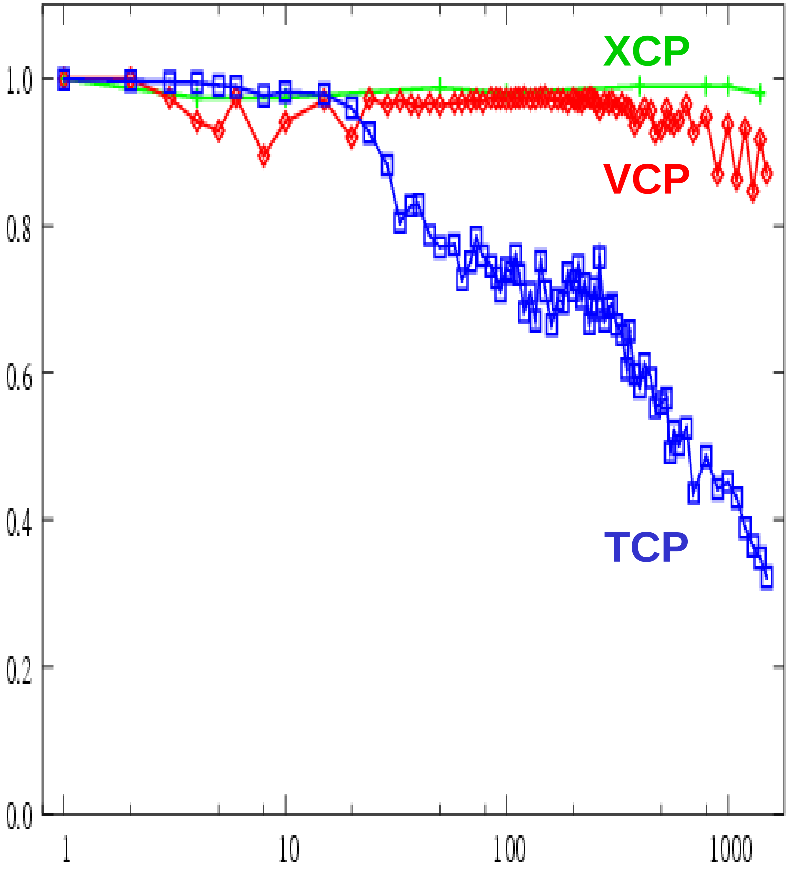
VCP achieves high efficiency

bottleneck utilization



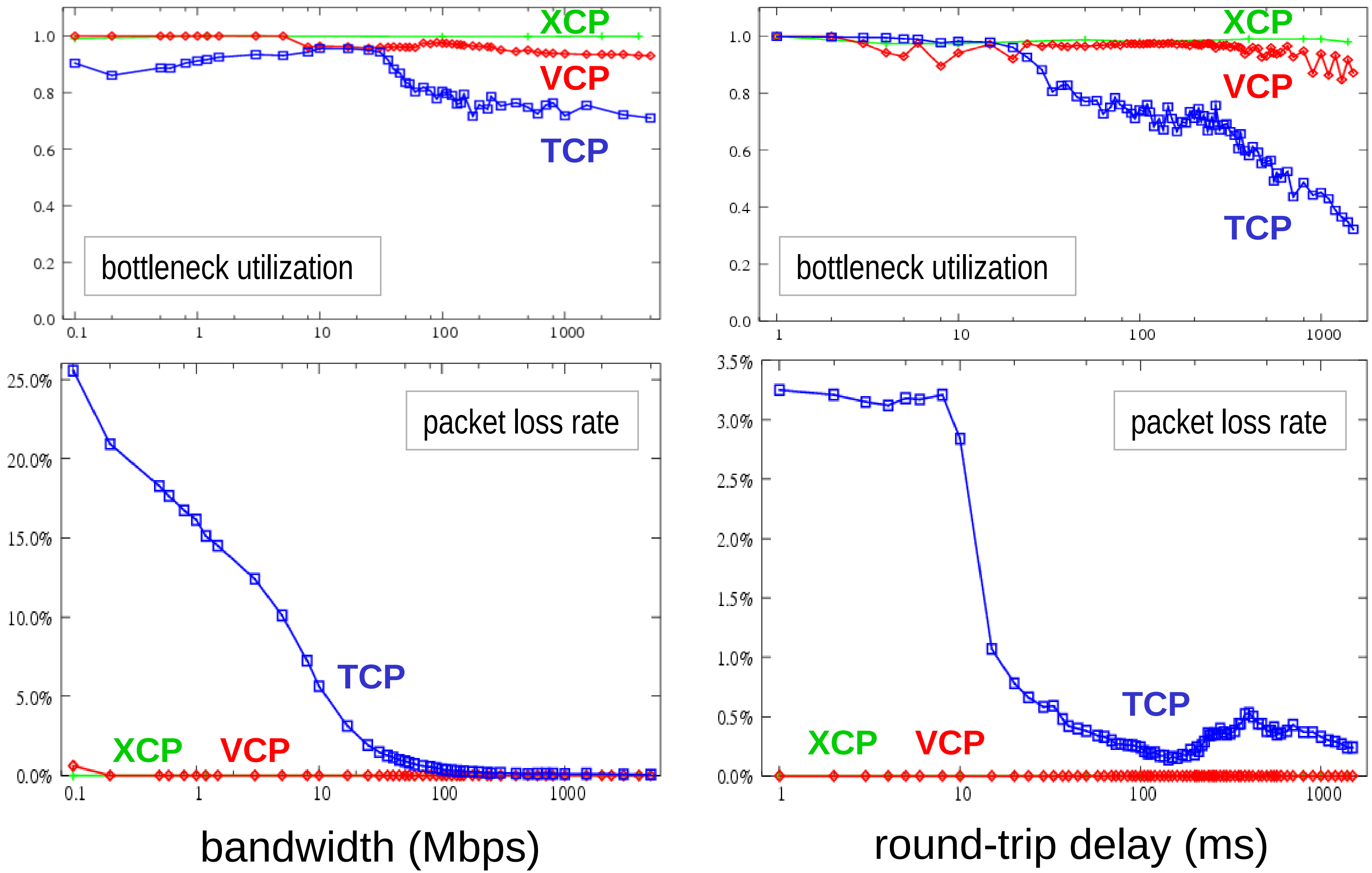
bandwidth (Mbps)

bottleneck utilization



round-trip delay (ms)

VCP minimizes packet loss rate



VCP comparisons

- Compared to TCP+AQM/ECN
  - Same architecture (end-hosts control, routers signal)
  - Router congestion detection: queue-based → load-based
  - Router congestion signaling: 1-bit → 2-bit ECN
  - End-host adapts (MI/AI/MD) according to the ECN feedback
  - End-host scales its MI/AI parameters with its RTT

- Compared to XCP
  - Decouple efficiency/fairness control across load regions
  - Functionality primarily placed at end-hosts, <u>not</u> in routers

Theoretical results

- Assumptions:
  - One bottleneck of infinite buffer space is shared by synchronous flows that have identical RTTs;
  - The exact value of load factor is echoed back.
- Theorem for the VCP fluid model:
  - It is globally stable with a unique and fair equilibrium, if $k \leq 0.5$;
  - The equilibrium is max-min fair for general topologies;
  - The equilibrium is optimal by achieving all the design goals.
- VCP protocol differs from the model in fairness.

Conclusions

- With a few minor changes over TCP+AQM/ECN, VCP is able to approximate the performance of XCP
  - High efficiency
  - Low persistent bottleneck queue
  - Negligible congestion-caused packet loss
  - Reasonable (i.e., TCP-like) fairness

Future work

- How do we get there, <u>incrementally</u>?
  - End-to-end VCP
  - TCP-friendliness
  - Incentive
- Extensions
  - Applications: short-lived data traffic, real-time traffic
  - Environment: wireless channel
- Security
  - Robust signaling, e.g., ECN nonce

The end

Thanks!

Design issue #3: Handling RTT heterogeneity for MI/AI

- TCP throughput is biased against flows with large RTT

$$rate \approx \frac{packet\_size}{rtt \cdot sqrt(\,loss\_rate\,)}$$

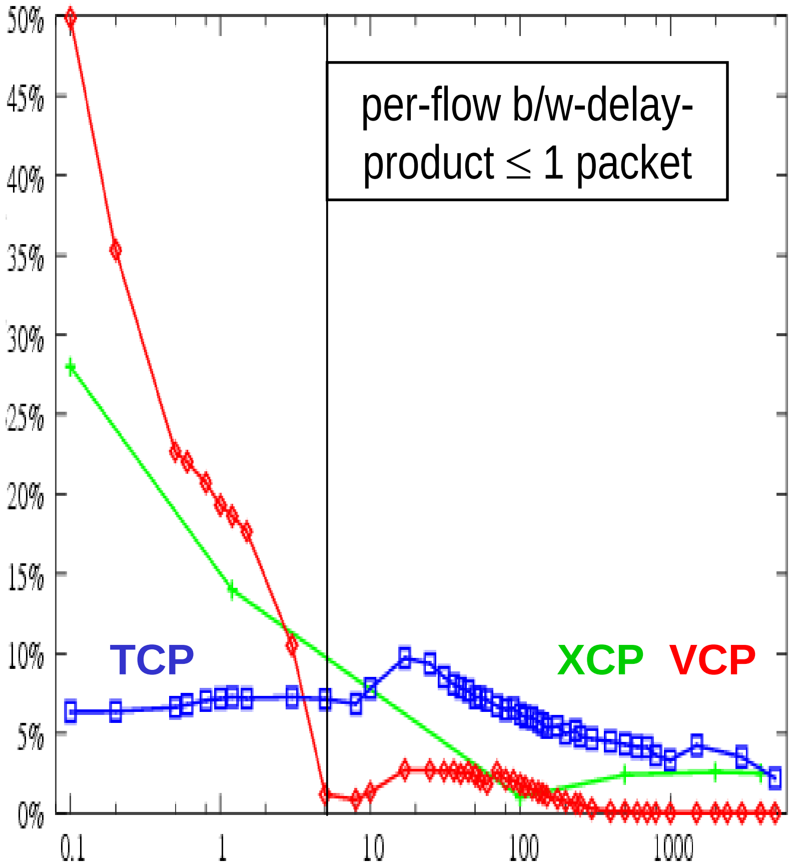
- VCP scales α for fair rate sharing (regardless of RTT)

$$\texttt{For AI :} \quad \alpha_{rate} \leftarrow \alpha \cdot \frac{rtt}{t_\rho} \cdot \frac{rtt}{t_\rho}$$

rate = cwnd / rtt

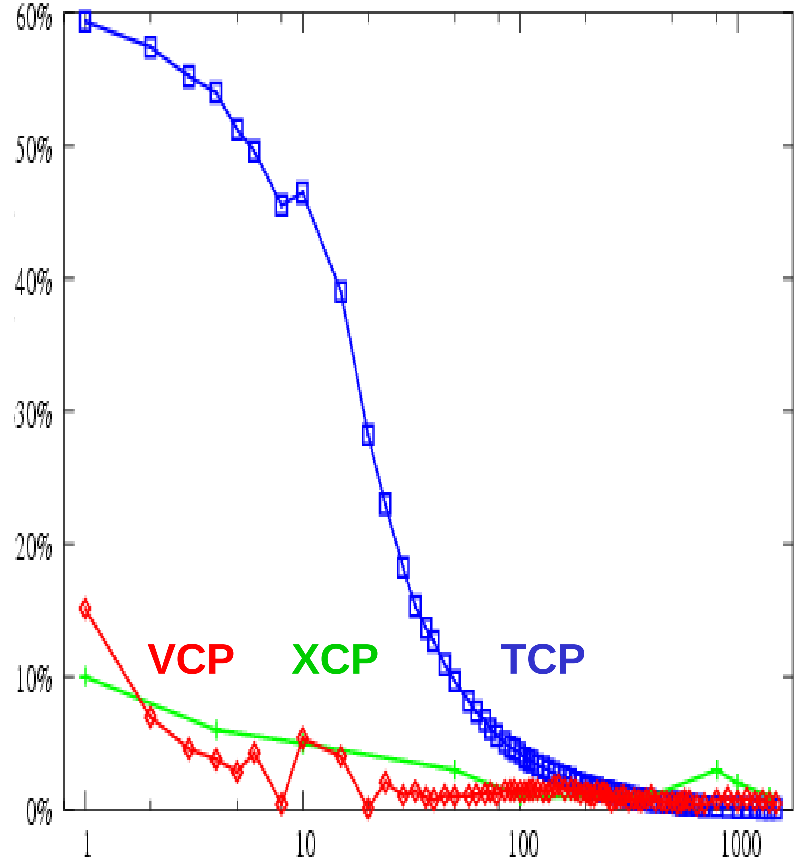
VCP keeps small bottleneck queue

queue length in % buffer size



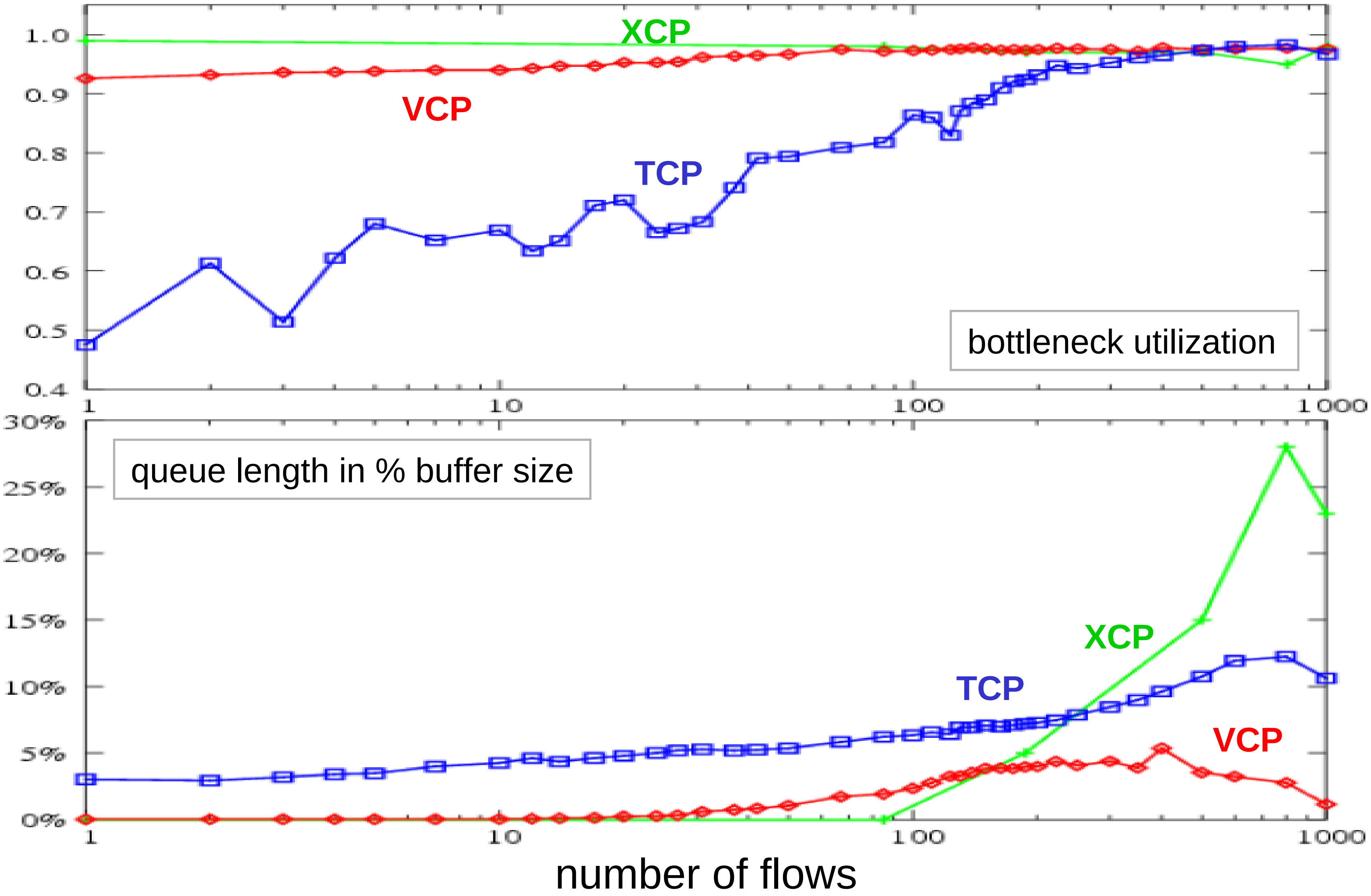
bandwidth (Mbps)

queue length in % buffer size



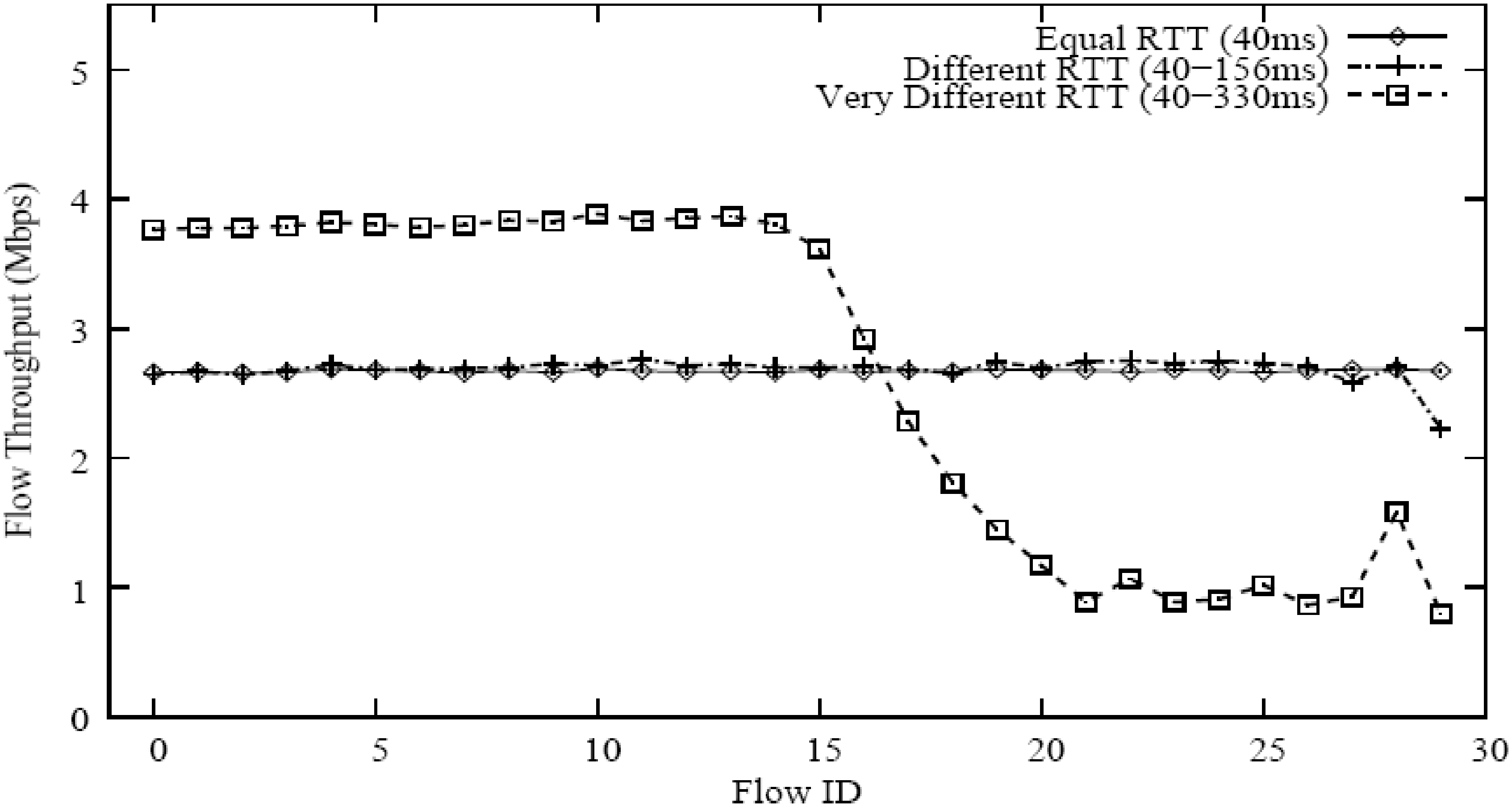
round-trip delay (ms)

Vary the number of flows

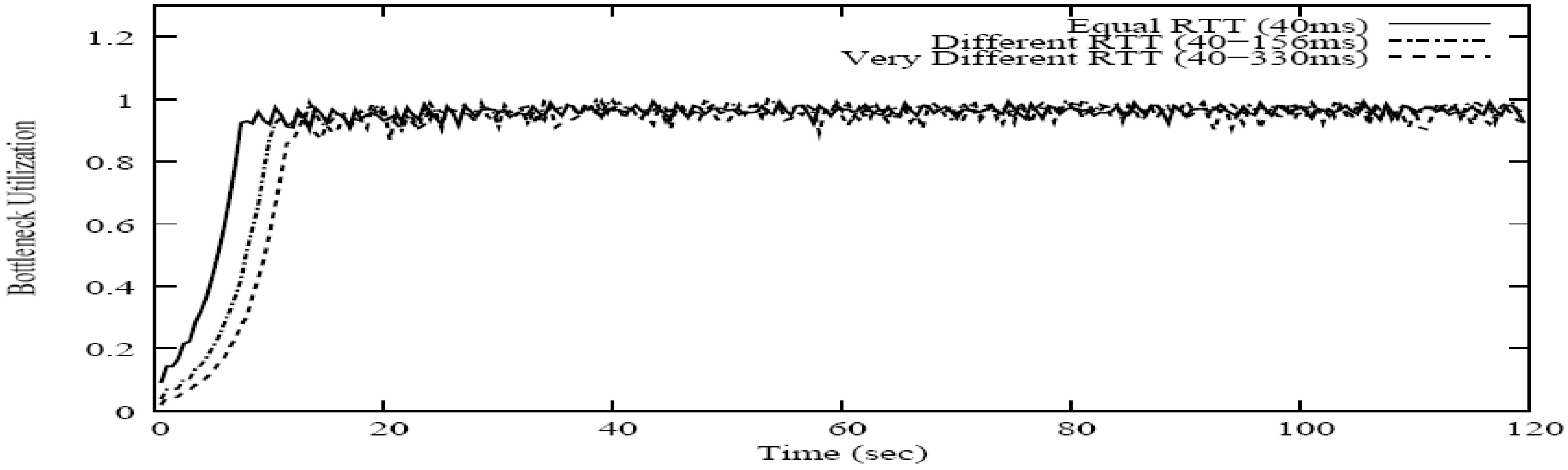


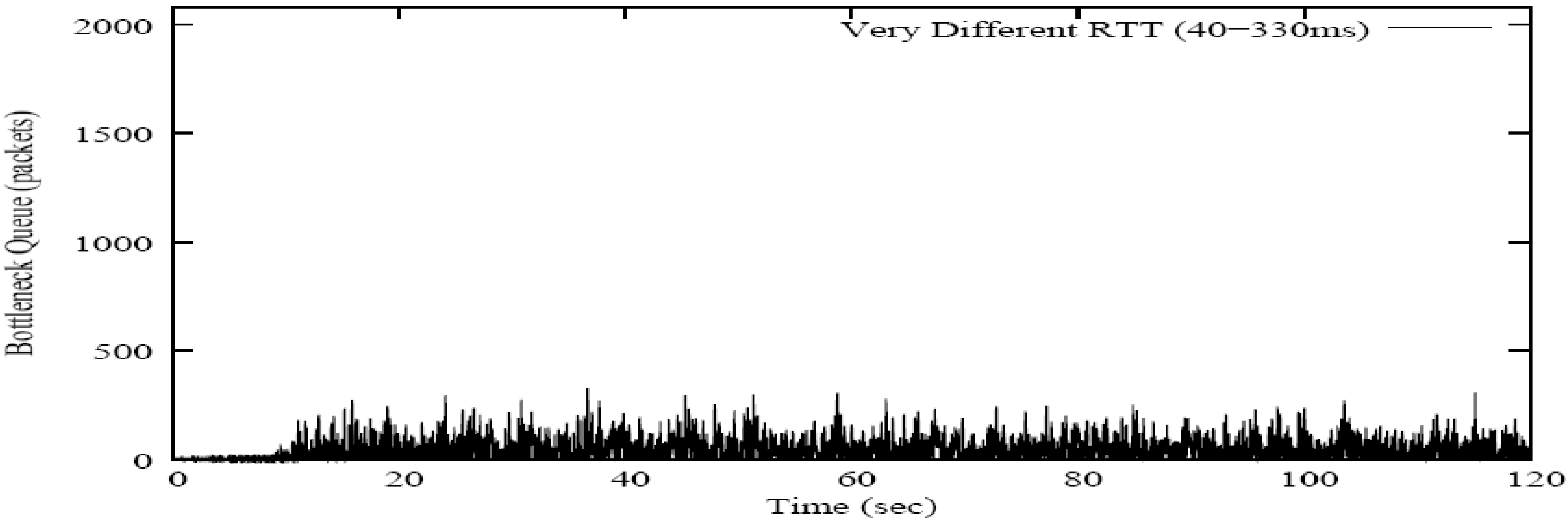
Influence of RTT on fairness

- To some extent, VCP distributes reasonably fairly

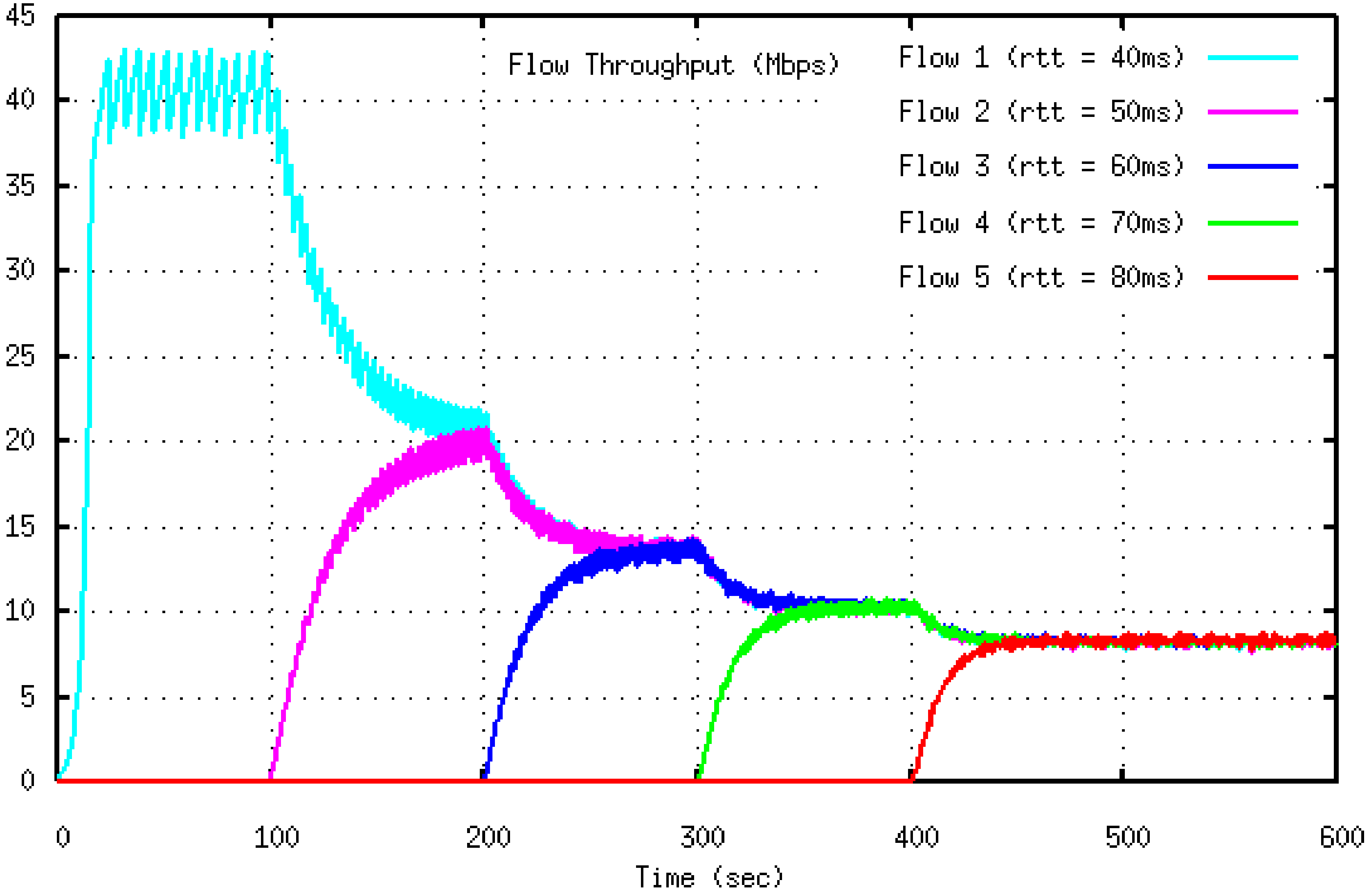


Influence of RTT on fairness (cont'd)

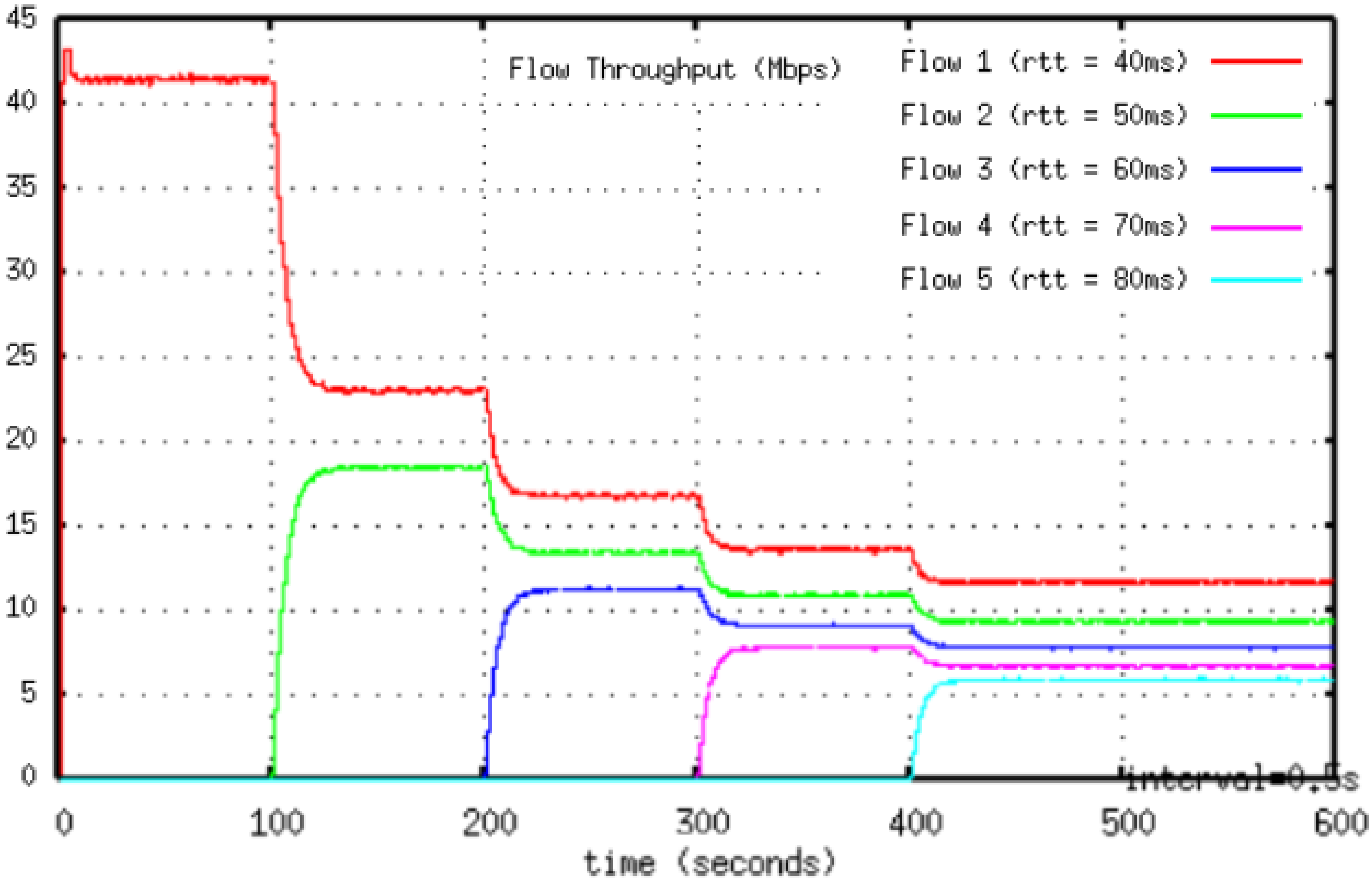




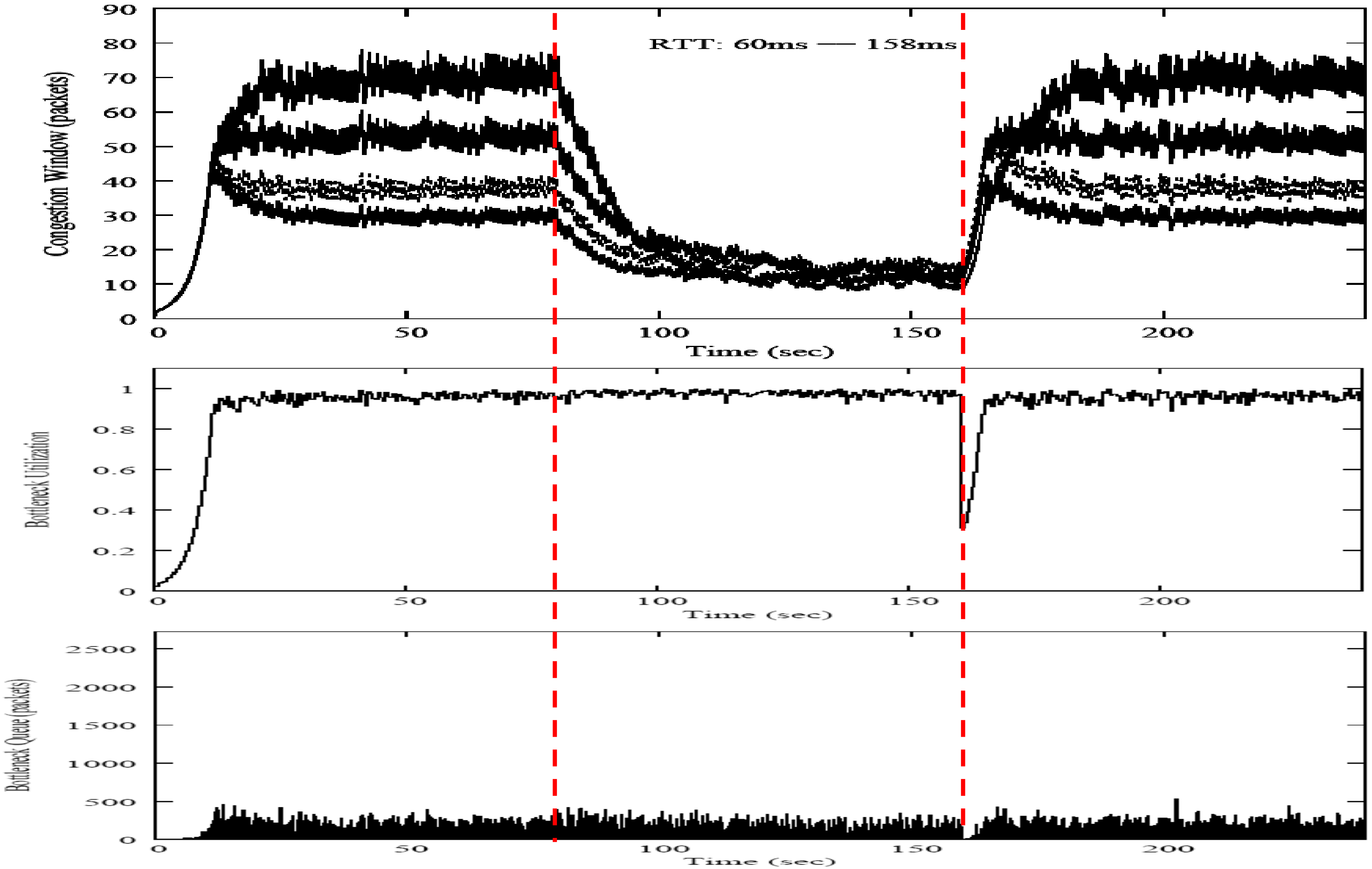
VCP converges onto fairness



VCP converges onto fairness faster with 8 bits



Responsiveness



50 flows + 150 flows – 150 flows