



# Implementing and Experimenting with XCP

Ted Faber, Aaron Falk, Yuri Pryadkin, Bob Braden,  
Eric Coe, Aman Kapoor, Amit Yajurvedi, Nirav Jasapara

USC/ISI  
USC Viterbi School of Engineering

28 Sept 2005

# Outline

---

Talk Layout

## XCP Overview

- Basic algorithms
- Keys for investigation

## Current Implementation Work

- FreeBSD implementation
- XCP Performance Enhancing Proxy
- Removing Divisions/IXP

## Coming Attractions

- Bursty flows in XCP
- XCP in a mixed environment

# XCP

---

eXplicit Congestion Protocol

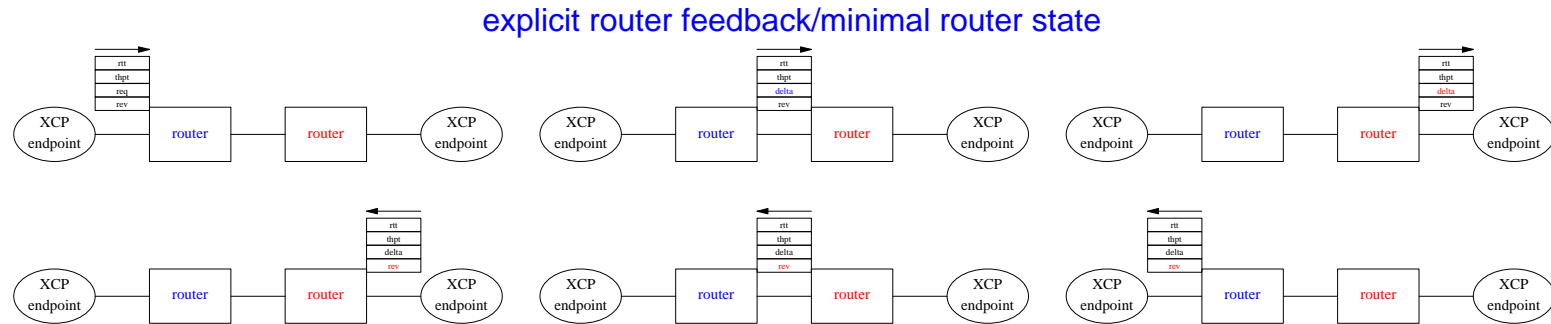
## History

- Proposed by Katabi, Handley & Rohrs in SIGCOMM '02
- Fleshed out in Katabi's Thesis at MIT
- Analysis continues, e.g. Low, et al.

## USC/ISI's interest

- Move XCP from Theory to Practice
- Implementation
- Standardization
- Fill in some details
- Research on practical elements

# XCP in Action



One XCP instance per queue

Each instance assigns feedback based on local information

- allocations only go down (implicit min on feedback)

Sampling and control changes every Control Interval (CI)

- CI is an estimate of mean RTT

# XCP Features

---

why is XCP not YARACCP?

## Dual controllers - Utilization/Fairness

### ■ Utilization

- all sources increase/decrease the same

### ■ Fairness

- sources increase same
- decrease inversely proportional to current throughput

## Distributed (time-shifted) state

- Routers store throughput allocations at endpoints

## Minimal Router State

- RTT sums to compute control interval (avg. RTT)
- utilization over control interval
- standing queue over control interval
- link capacity

# XCP Outside the Sim

---

Can anyone use it?

## Implementations and Interoperability

- Specification
- Reference Implementation
- Experience

## Router Support

- Reference Implementation
- Complexity

## Deployment

- Incremental Deployments
- Mixed Environments

# XCP Outside the Sim

---

Can anyone use it?

## Implementations and Interoperability

- Specification
- Reference Implementation
- Experience

## Router Support

- Reference Implementation
- Complexity

## Deployment

- Incremental Deployments
- Mixed Environments

# FreeBSD Implementation

---

integration with TCP code

## Layer 3 1/2 implementation

- Little space in options
- Router code also must process

## System stuff

- parameter representation
- IPv6
- delayed checksums
- logging and performance evaluation

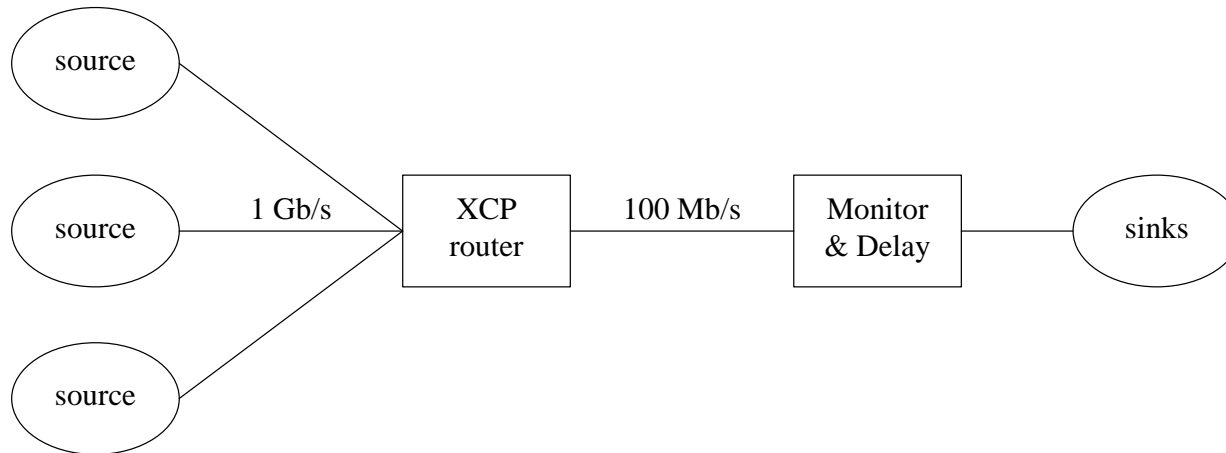
## Updated release in days

- <http://www.isi.edu/isi-xcp>

# XCP Topology

---

One of our Regression Tests



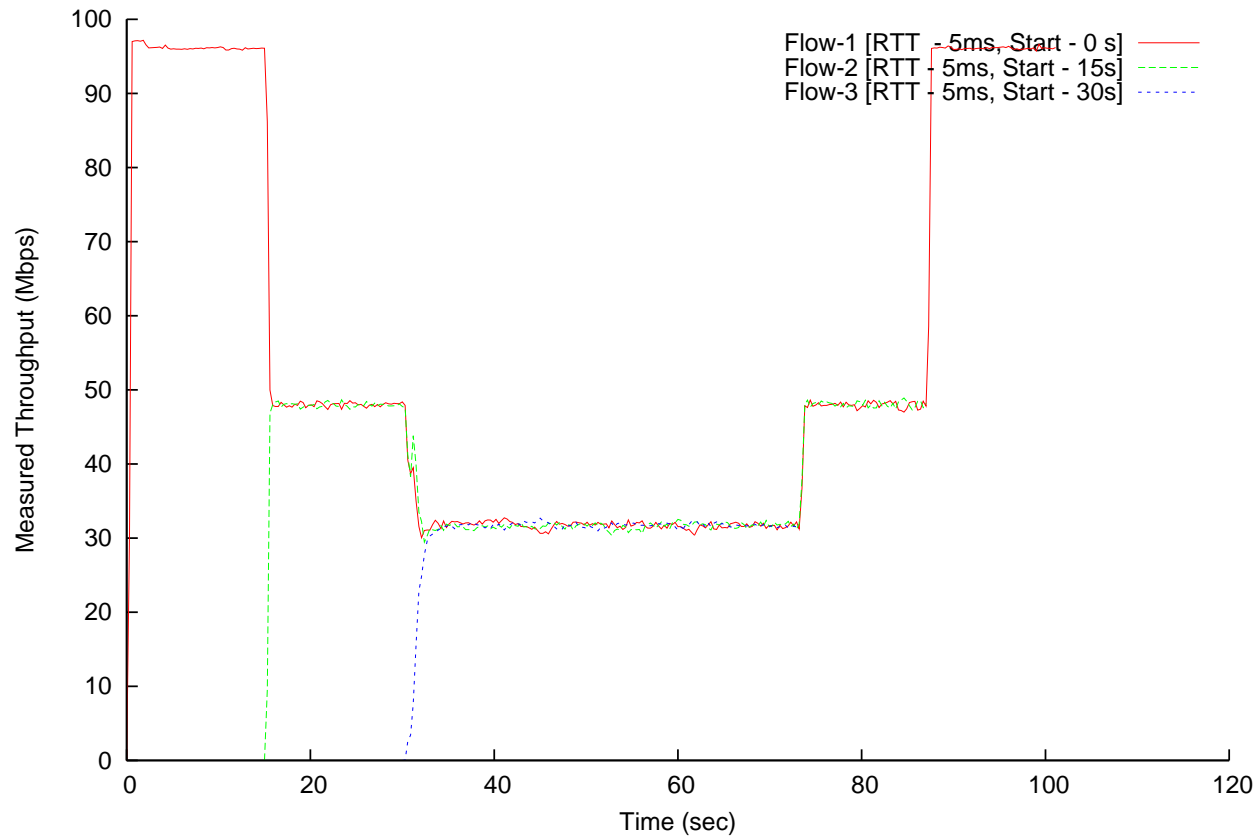
ACKs delayed to increase BDP (5ms)  
Throughput measured at 300 ms intervals  
Sources join sequentially

# Example From Regression Suite

---

## Typical Run

Throughput at the router



# XCP Outside the Sim

---

Can anyone use it?

## Implementations and Interoperability

- Specification
- Reference Implementation
- Experience

## Router Support

- Reference Implementation
- Complexity

## Deployment

- Incremental Deployments
- Mixed Environments

# Differential Deployment

---

Satellite Environments

Probing Systems (TCP) have trouble with scale

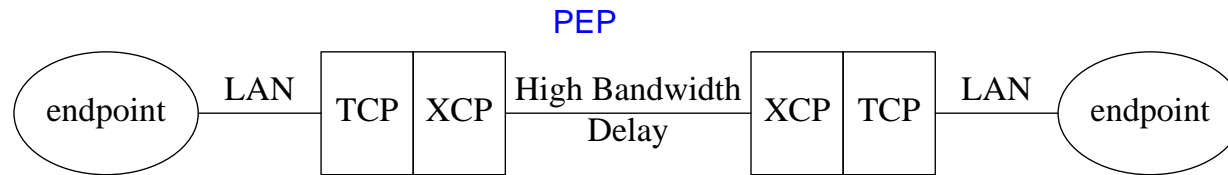
Often one of:

- Slow to acquire large allocations
- Too aggressive in scarce resources

XCP has capacity knowledge, but faces deployment issues

# Performance Enhancing Proxy

---



## Connection splitting proxy

- End-to-end semantics sacrificed
- Each congestion system works in its environment
- End-to-end throughput improves (time to full utilization)

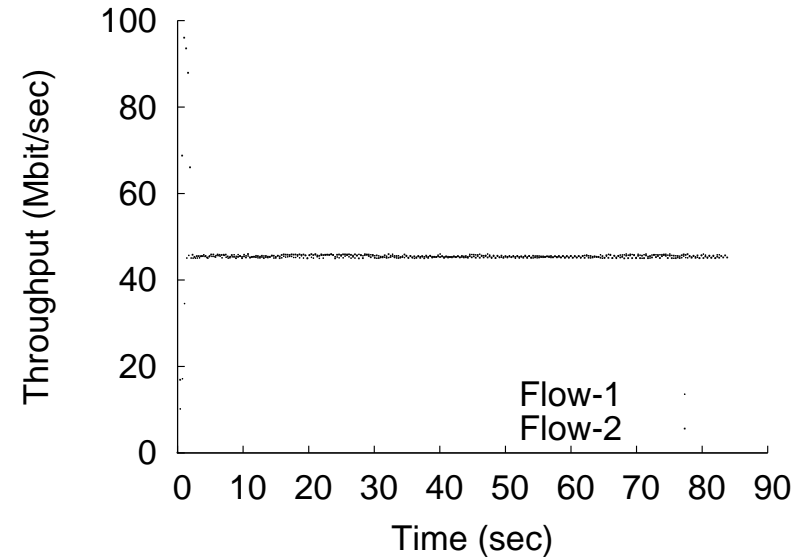
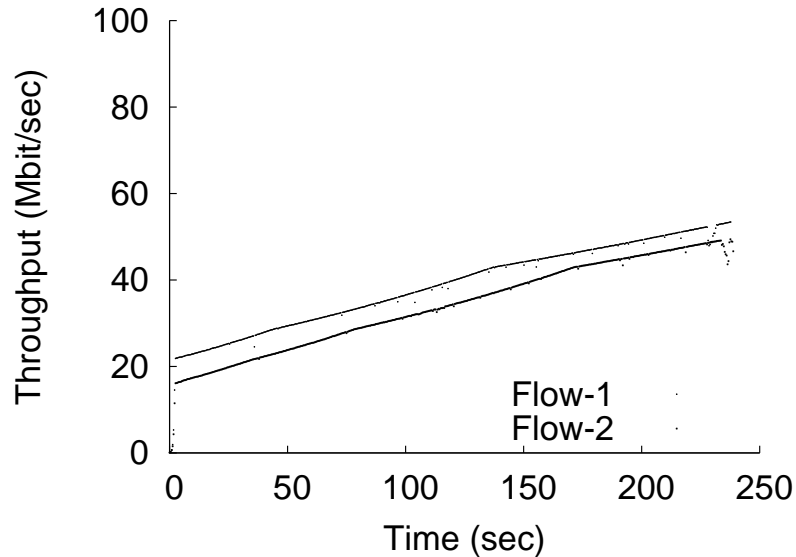
## Suitable for environments with

- Application level end-to-end checks
- Network with high BDP and control

## Satellite Networks

# Long Link w/o PEP

Throughput Comparison



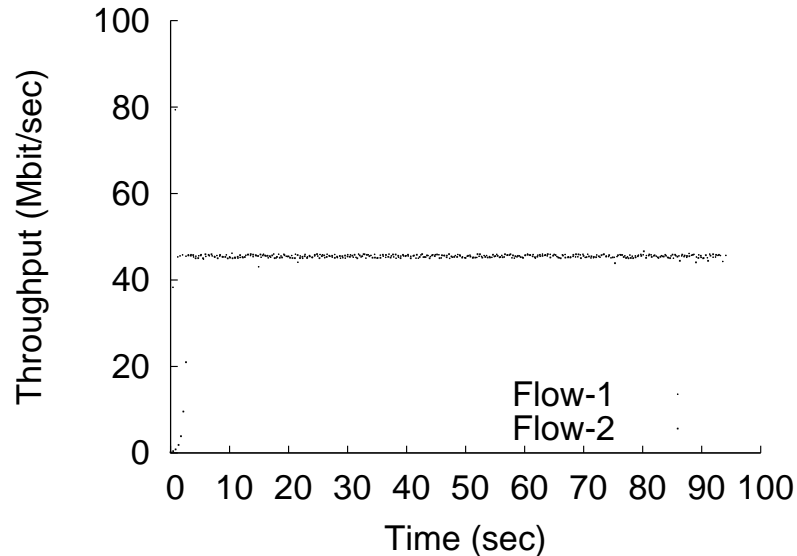
TCP slow to open the window

XCP requires end to end deployment

# Long Link with PEP

---

Best of Both Worlds



## The gory details:

- Kapoor, Falk, Faber, Pryadkin, "Achieving Faster Access to Satellite Link Bandwidth", Global Internet Symposium, March 2005
- <http://www.isi.edu/isi-xcp/docs/kapoor-pep-gi2005.final.pdf>

# XCP Outside the Sim

---

Can anyone use it?

## Implementations and Interoperability

- Specification
- Reference Implementation
- Experience

## Router Support

- Reference Implementation
- Complexity

## Deployment

- Incremental Deployments
- Mixed Environments

# Removing Router Divisions

---

Reparameterizing the Algorithm

XCP per-packet calculations include integer divisions

Division used 2 places:

- Calculating feedback
- Determining control interval

Routers/Network processors not mathematically capable

- No integer division on Intel IXP processors

SIGCOMM paper suggests removing division

# Divide in Endpoints

---

as usual offload work to endpoints

Both places the relevant division is packet size/rate

- Send that value (X) instead of throughput

What is X?

- Intuition: idealized interpacket time
- Actually: current packet size / current router-assigned throughput
- Format: 32-bit unsigned 4 bits left of radix in sec

# No Division Parameter Ranges

---

nitty gritty

Parameter	Min.	Max.
X	3.725ns	16s
RTT	3.725ns	16s
Throughput (packet size = 64B)	32 b/s	137.4 Gb/s
Throughput (packet size = 64kB)	32.8 kb/s	140.7 Tb/s
Throughput (packet size = 4GB)	2.15 Gb/s	9.22 Eb/s

These are encoding limits

Earlier plots made with this encoding

# XCP Outside the Sim

---

Can anyone use it?

## Implementations and Interoperability

- Specification
- Reference Implementation
- Experience

## Router Support

- Reference Implementation
- Complexity

## Deployment

- Incremental Deployments
- Mixed Environments

# Differing RTTs and Burstiness

---

Burstiness is Ted's Pet Peeve

Burstiness: sending patterns that are not smoothly clocked

- follow throughput allocation over long time intervals (RTT)
- may exceed throughput allocation over shorter times ( $<RTT$ )

N.B.: Short timescales

Add widely differing RTTs and XCP can be disrupted

Observed in implementation, being studied in simulation

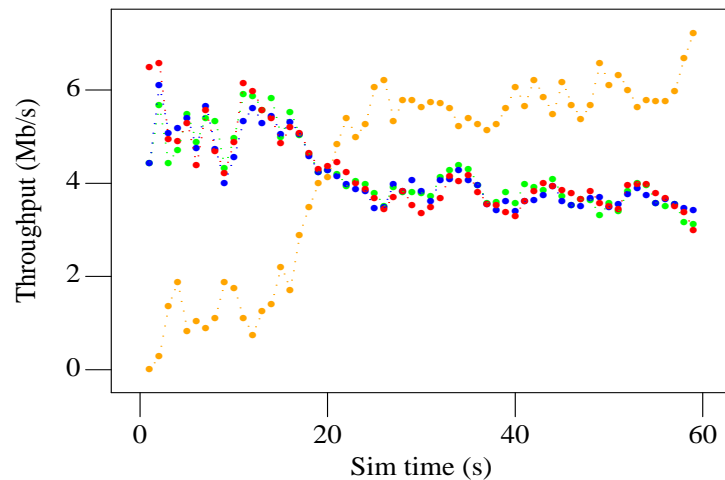
# Disruption Example

Simulation to screen out implementation weirdness

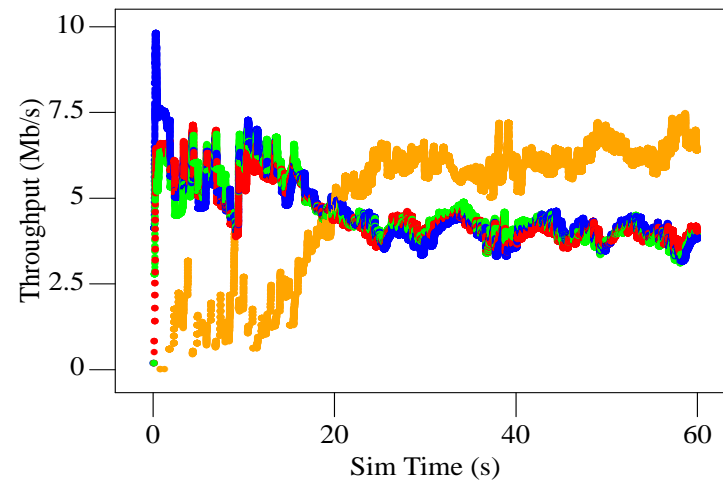
## Simulated 4 Source dumbbell topology, FTP sources

- bottleneck 20 Mb/s, line rate 100 Mb/s
- src 1-3 RTT=40ms, src4 RTT=500ms (orange)

Throughput (calculated at router - 1 s interval)

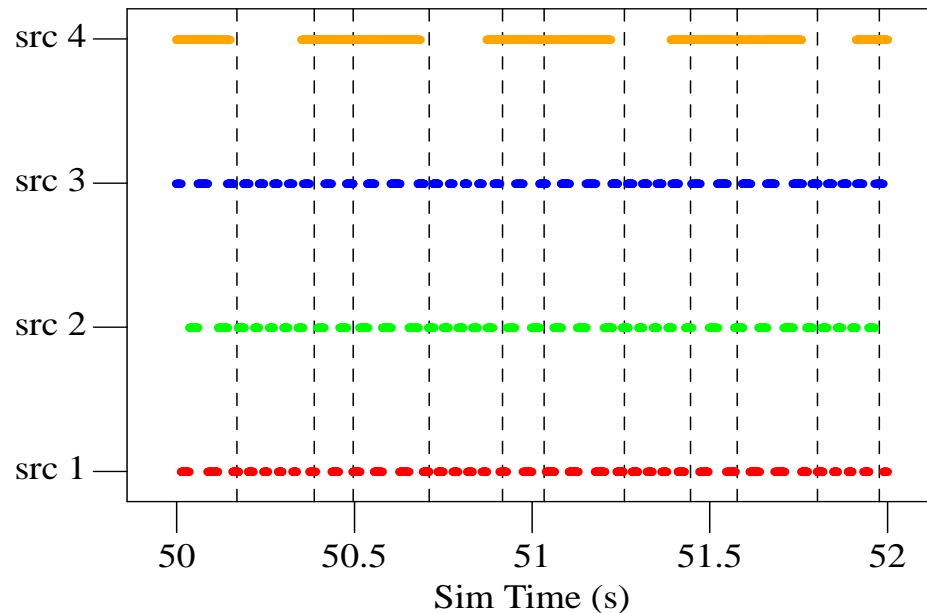


Throughput (reported at router)



# How Do We Know There's Burstiness?

Well, we looked...  
Packet Arrival at Router



Bursts are more pronounced in long RTT flows  
Some Control Intervals have few or no packets

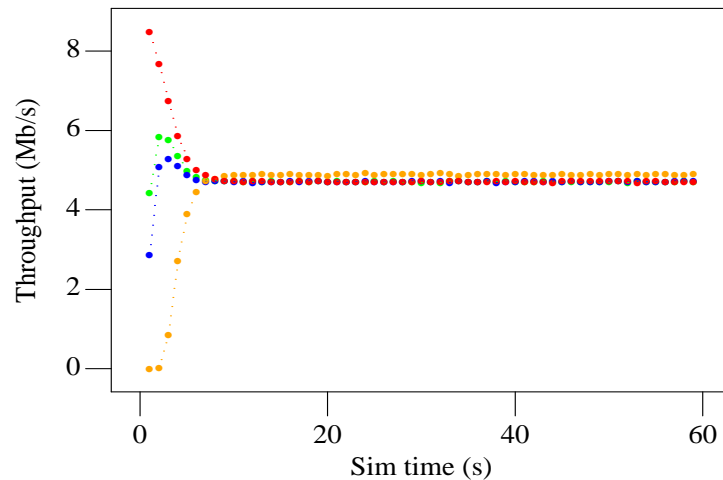
# How Do We Know Burstiness Is Connected?

Try again with smoothed traffic

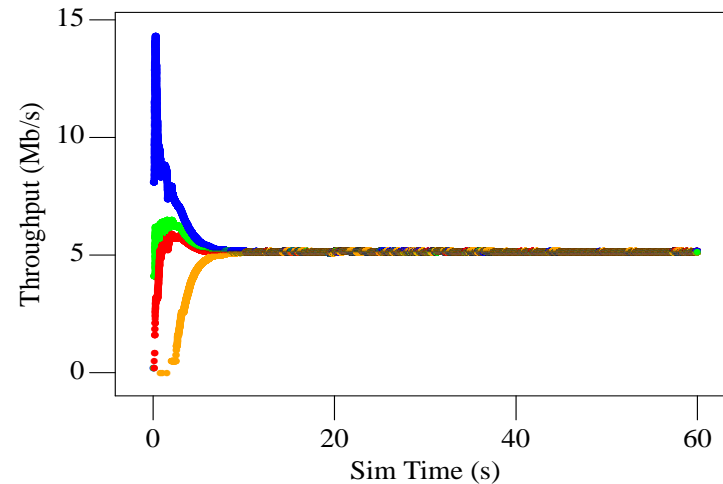
## Modified Source Behavior

- Sources now send evenly spaced packets for a given cwnd
- Just a test to see if burstiness is the problem

Throughput (calculated at router - 1 s interval)



Throughput (reported at router)

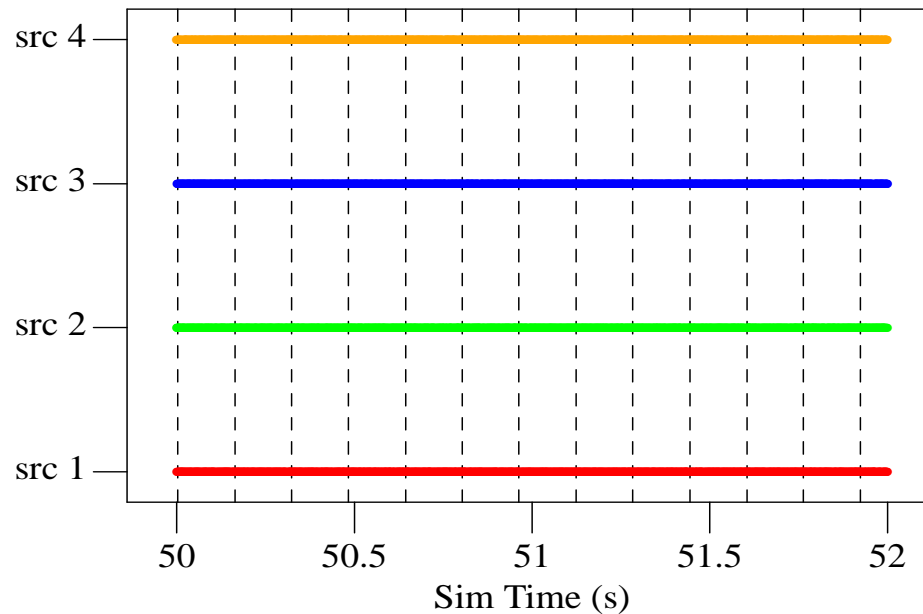


# Burstiness Disappears

---

Check the packet traces

Packet Arrival at Router



Smooth Packet flow at all RTTs

# What Next

---

Moving on...

## Better Mitigation Strategies?

- Can't outlaw burstiness
- Pacing may be expensive
- Increasing CI seems to work, but...

Understanding causes of induced burstiness

# XCP Outside the Sim

---

Can anyone use it?

## Implementations and Interoperability

- Specification
- Reference Implementation
- Experience

## Router Support

- Reference Implementation
- Complexity

## Deployment

- Incremental Deployments
- Mixed Environments

# XCP without XCP routers

---

You've got to have some...

XCP requires all routers to be XCP-capable

Guaranteeing this - or detecting the opposite - is hard

- MPLS
- Bridges

End-To-End argument argues for End-To-End control

- XCP works much better when bottleneck is XCP capable
- Traditional inferential Congestion Control other times

# When To Use Which Control?

---

If you believe you can swap

XCP to inferential is easy - ECN or packet loss

## Inferential to XCP

- World becomes worse: must detect XCP is tighter
- World becomes better: must detect that inferential is too tight

## Proposals

- Packet format that allows XCP routers to send advisory feedback
- Occasional probing with exponential backoff

# Open Issues

---

(there aren't any closed ones...)

How disruptive is inferential traffic to XCP

- fairness engine is pretty oblivious

How disruptive are probes to non-XCP bottlenecks?

- burstiness issues from earlier

Probe synchronization, frequency

# Summary

---

Watch This Space

## Implementations and Interoperability

- Specification
  - forthcoming (with implementation update)
- Reference Implementation
  - In use; update forthcoming
- Experience
  - Burstiness & large RTT variance

## Router Support

- Reference Implementation
  - IXP implementation forthcoming (not with implementation)
- Complexity
  - Division removal

# Summary

---

Watch This Space

## Deployment

- Incremental Deployments
  - PEP analysis
- Mixed Environments
  - Ongoing analysis

<http://www.isi.edu/xcp-isi>