Performance of Paced and Non-paced Transmission Control Algorithms in Small Buffered Networks

Onur Alparslan, Shin'ichi Arakawa, Masayuki Murata

Osaka University

27 June 2006

Outline

- **Objective**
- **Small Buffering**
- **eXplicit Control Protocol (XCP)**
- **Simulations**
- **Conclusions**

Objective

- Compare the performance of paced and non-paced versions of TCP and XCP with small buffering
- Show stability of XCP

Advantages

- If we can reduce the buffer requirements, hardware cost and power consumption of high speed routers greatly decrease

Buffer Size Requirement

- According to a rule-of-thumb, output link buffer size requirement is *B = RTT x BW*
- Appenzeller et al. showed that *B = RTT x BW/sqrt(n)* is enough when there are *n* TCP flows on the link
  - Requires a large number of flows for a drastic decrease in buffer requirements.
- Recently, Enachescu et al. showed that when TCP pacing is used, *O(logW)* buffers are sufficient where *W* is the maximum congestion window size of each flow.

eXplicit Control Protocol (XCP)

- **XCP is a transmission control protocol proposed by Katabi et al., with a new control theoretical framework**
  - XCP uses explicit feedback coming from core routers
  - Tries not to lose any packets.
  - Achieves **fair bandwidth allocation**, **high utilization**, **small standing queue size**, and **near-zero packet drops** with both steady and highly varying traffic.
  - Does not maintain any per-flow state in core routers

Minimizing Buffer Usage with XCP

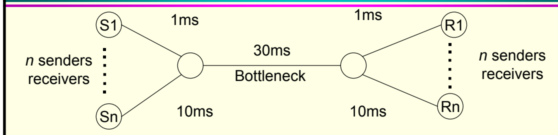
- **Preventing link over-utilization**
  - Carefully select XCP parameters
  - Control maximum link utilization ratio
    - At each XCP core node, link speed must be explicitly given to XCP control algorithm
    - By giving a false capacity value to XCP lower than actual link speed:
      - Possible to control maximum link utilization
      - XCP algorithm limits the average link throughput to the given false speed
      - Prevents buffer buildups
      - Can be explicitly given as a target utilization
- **Burstiness**
  - Apply pacing to XCP flows

Stability and Buffer Requirements of XCP

- **We compare,**
  - Non-paced XCP with original parameters
  - Non-paced XCP with conservative parameters
  - Paced XCP with original parameters
  - Paced XCP with conservative parameters
- **Original Parameters**
  - α=0.4
  - γ=0.1
  - β=0.226
  - Target utilization=100%
- **Conservative Parameters**
  - α=0.2
  - γ=0.05
  - β=0.056
  - Target utilization=90%

7

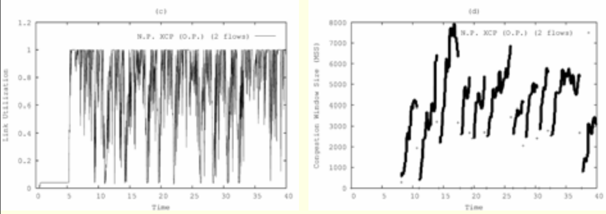
Dumbbell Topology Simulation Settings



- ***n*** **is ranging between 2 - 800 nodes**
- **622Mbps (OC-12) Bottleneck, 2.4Gbps (OC-48) extension links**
- **Two-way traffic**
- **FTP flows start randomly in (0,10) sec.**
- **100 sec. simulation**

8

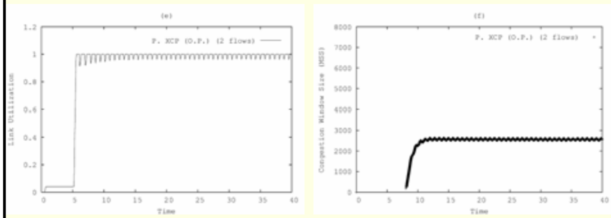
Non-paced XCP with Original Parameters



- **2 flows (4 flows including the reverse flows)**
- **Buffer size is set to Rule-of-Thumb**
- **Original XCP is oscillatory when there is two way traffic and few number of flows**
  - XCP can increase its window size in bigger steps than TCP, so it is burstier than TCP
  - High burstiness causes ACK starvation problem
  - ACK starvation further increases the burstiness and thus XCP becomes too much oscillatory and loses many packets

9

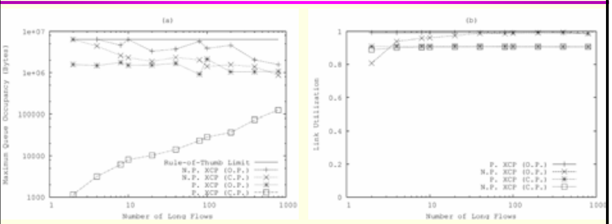
Paced XCP with Original Parameters



- **Buffer size is set to Rule-of-Thumb**
- **Pacing solves the burstiness problem and XCP becomes stable**
- **No packet loss**

10

Buffer Requirements



- **Buffer size is set to Rule-of-Thumb**
- **Using conservative parameter set decreases the buffer requirements, but still packet losses occur**
- **Pacing solves the stability problem, so there is no packet loss**
- **Using pacing and conservative parameter set together drastically decreases buffer requirements**

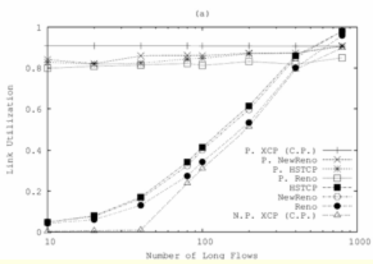
11

Utilization Comparison

- **Set the buffer size to maximum buffer requirements of paced XCP with conservative parameter set with corresponding number of flows**
- **We compare both Paced and non-Paced versions of**
  - Reno
  - NewReno
  - HSTCP
  - XCP

12

Utilization Comparison



- Paced XCP and TCP versions have higher utilization in general
- Non-paced TCP and XCP have high utilization with small buffering only when the number of flows is high enough according to Appenzeller's *B = RTT x BW/sqrt(n)* formula

Conclusions

- **Even rule-of-thumb sized buffers are not enough for XCP in some cases due to its high burstiness.**
- **XCP can be adapted to small buffered networks by pacing and a careful selection of parameters.**
- **Paced TCP and XCP versions have much lower buffer requirements than non-paced versions.**
- **A big disadvantage of XCP based algorithms is that they require deployment of XCP capable senders, receivers and routers.**
- **Possible to use P. TCP algorithms by updating only senders.**