Evaluation of FAST TCP in Low-Speed DOCSIS-based Access Networks

by
L.H. Andrew[†], I. Atov[*], D. Kennedy [*]
and B. Wydrowski[×]

[†] University of Melbourne
[*] Swinburne University of Technology
[×] Caltech, USA

Presentation Outline

- Motivation
- Overview of FAST and DOCSIS protocols
- Experimental setup
- Results and analysis
- Conclusions and future work

Motivation

- A replacement for TCP Reno, called FAST TCP, is being designed at Caltech to improve performance in high-speed networks
- FAST has been tested by Caltech and independent groups such as SLAC and CERN only in ***high speed environments***
  - Our aim - to experimentally evaluate the performance of FAST in ***low speed environments*** (0.5 – 3 Mbps) i.e:
    - links using DOCSIS cable modem MAC protocol
    - simple low rate links
  - The understanding gained will allow optimal parameter settings to be determined for a range of conditions

FAST overview

- TCP regulates source sending rate by adapting window size according to some congestion signal:
  - Packet-loss-based: Most TCP algorithms since TCP Tahoe regulate source sending rate by adapting window size according to the packet loss rate ← Reno, HSTCP, BIC
  - Delay-based: Flow rates are adjusted in response to the measured delay. These algorithms attempt to maintain for a flow a constant number of packets, α, queued in nodes along its path ← FAST, Vegas
- FAST updates the window size according to:

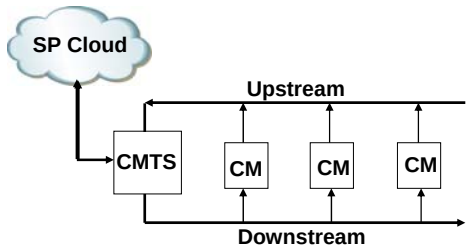
$$w(t+1) \le \left\lfloor \frac{1}{2}\left(w(t) + \frac{d}{D} w(t) + \alpha \right) \right\rfloor$$

D – average RTT
d – base RTT

Focus of our work

- The focus of our work is how to set the alpha parameter in two different low-speed environments:
  - links using DOCSIS cable modem protocol
  - simple low rate links
- High-speed: For high-speed links, it has been recommended that α be set to cause a given small queueing delay, such as ~2ms i.e., $\alpha = 2C$, $q = 2C/C$
- Low-speed: In this work we show that this rule of thumb gives insufficient queueing for low-speed networks, especially when DOCSIS is used

Overview of DOCSIS

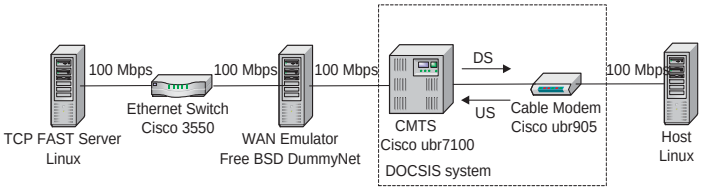


Logical topology of DOCSIS network

DOCSIS networks operate on a reservation scheme (known as *Request-and-Grant* cycle) where the modems request a time to transmit and the CMTS grants time slots based on availability.

- DOCSIS system introduces latency fluctuations based on the offered load in the system (T. T. T., Nguyen and G. J. Armitage – Globecom 2004)
- Our goal is to investigate how the Request-and-Grant control of transmission cycles between the CMTS and the CM can affect the performance of FAST TCP and to identify factors that need to be considered in achieving optimal performance over a DOCSIS cable network

Experimental Setup

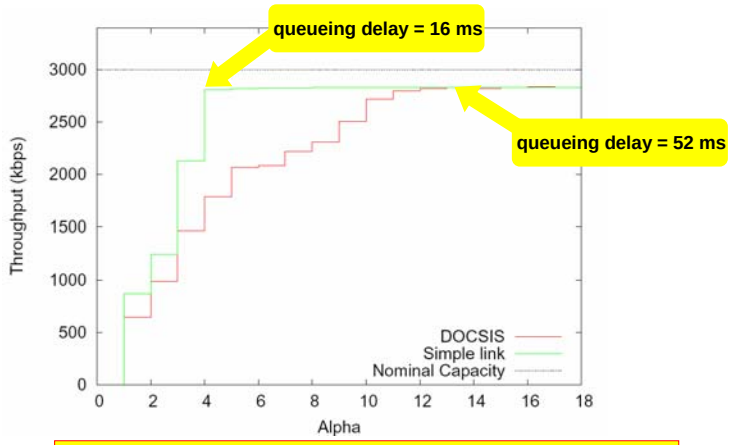


DS	US
256 Kbps	128 Kbps
512 Kbps	256 Kbps
1.54Mbps	512 Kbps
2 Mbps	512 Kbps
2.5 Mbps	512 Kbps
3 Mbps	512 Kbps

- DOCSIS link:
  - Dummynet: RTT=100ms, no bandwidth limitations, buffer size of 2048 Kbytes
  - Maximum buffer size at CMTS set to the default Cisco value of 512ms
- Simple rate-limited link:
  - DOCSIS system was bypassed. Dummynet emulated system with equivalent US and DS capacities and buffering. Also, RTT=100ms.
- The bottleneck link carried one or two FAST flows DS, and no other traffic

Single Flow Results

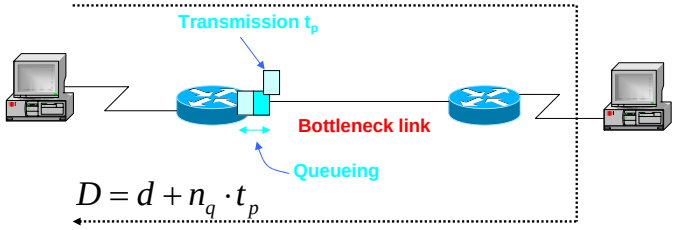
Throughput vs. α parameter for DS = 3 Mbps and US = 512 Kbps
(results are averaged over 100 runs)



Much larger queueing is needed than what is necessary to obtain accurate timing estimates!

FAST on a simple slow link

- There are two major effects at work causing utilisation to be low for $\alpha \leq 3$:
  - Burstiness – causes the average queue size observed by the packets to be greater than the true mean queue size. FAST attempts to minimise burstiness, but is hindered by TCP delayed acknowledgements (RFC 2581)
  - The integer arithmetic of FAST control mechanism



- Even for very low utilisation, delayed ACKs alone can allow D to be up to $d+t_p$ (t_p is the packet delay). Thus, FAST requires $\alpha \geq 1$ to achieve full utilisation

FAST on a simple slow link – cont.

- The second reason for requiring large α is the floor operation in (1)

$$w(t+1) = \left\lfloor \frac{1}{2}\left(w(t) + \frac{d}{D} w(t) + \alpha \right) \right\rfloor \qquad (1)$$

- Without the $\lfloor \cdot \rfloor$ operation, the update rule satisfies the equilibrium relationship

$$w(t+1) = \frac{1}{2}\left(w(t) + \frac{d}{D} w(t) + \alpha \right) \quad (2) \Rightarrow \quad w = \alpha \frac{D}{D-d}$$

- However, at equilibrium of (1) we know:

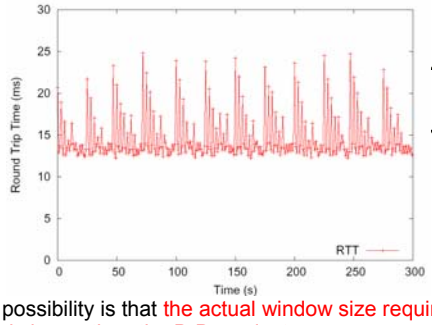
$$w \leq \frac{1}{2}\left(w + \frac{d}{D} w + \alpha \right) \leq w + 1 \Rightarrow \quad (\alpha - 2)\frac{D}{D-d} \leq w \leq \alpha \frac{D}{D-d}$$

- So that equilibrium window size can be as small as that predicted by (2) under the substitution $\alpha' = \alpha - 2$

Thus, we expect that $\alpha \geq 3$ for full utilisation

FAST on a DOCSIS link

- There are several possible reasons for this discrepancy:
  - The latency fluctuations introduced by the DOCSIS system could possible interfere with FAST estimates of the queueing in the network, resulting in the congestion window being too low



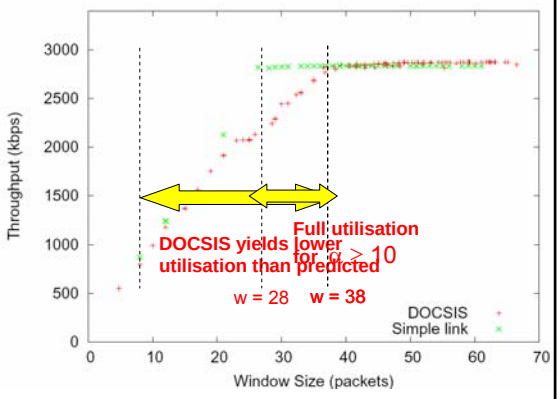
Selected packets experience up to 13ms jitter

  - Another possibility is that the actual window size required for full utilisation in DOCSIS is larger than the B-D product

FAST on a DOCSIS link – cont.

Throughput vs. window size for DS = 3 Mbps and US = 512 Kbps for 140 experiments using α values from 1 to 35

- A bottleneck link carrying a single flow in a purely deterministic network will be fully utilised if the flow's window size is at least the ``bandwidth delay product''
- For a 100 ms (or 115 ms) path with a bottleneck link of 3 Mbps, this is 25 (or 28) packets of 1500 bytes
- For smaller windows, the throughput reduces in proportion to the window size



DOCSIS yields lower utilisation than predicted

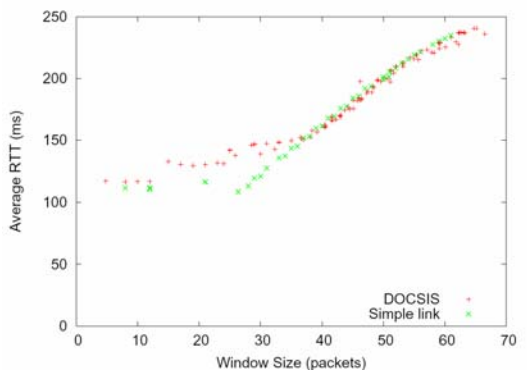
Full utilisation for $\alpha > 10$

w = 28 w = 38

This suggests that FAST window size is not adversely affected by the randomness of the delay

FAST on a DOCSIS link – cont.

Average RTT vs. window size for DS = 3 Mbps and US = 512 Kbps DOCSIS and simple link

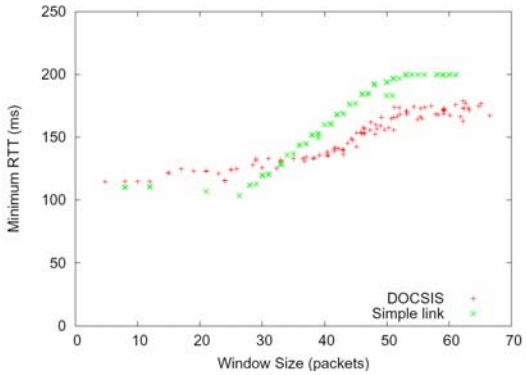


- For underutilised link, the queueing delay should be zero
- For fully utilised link the queueing delay should be proportional to the difference between the window size and the bandwidth delay product

FAST on a DOCSIS link – cont.

Minimum RTT vs. window size for DS = 3 Mbps and US = 512 Kbps DOCSIS and simple link

- The "phantom delay" cannot be attributed to burstiness, as it would be expected that at least some packets every round trip time would observe approximately the true propagation delay.
- Thus, it is not simply FAST's estimate of the RTT that has increased, but rather the actual RTT

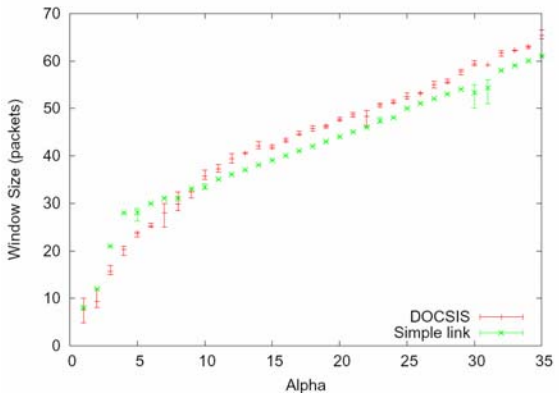


This indicates that DOCSIS may buffer packets even when the link is idle (due to the Request-and-Grant scheme it employs)

FAST on a DOCSIS link – cont.

Window size calculated by FAST as a function of α for DS = 3 Mbps and US = 512 Kbps, DOCSIS and simple link (error bars indicating the maximum and minimum)

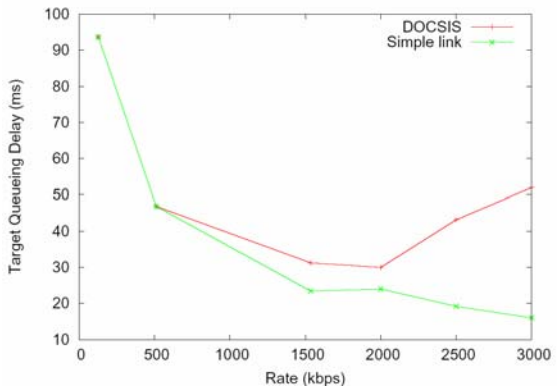
- The reason for requiring large α is that the DOCSIS delay causes FAST to set the window size too small
- When the link is underutilised, FAST underestimates w using DOCSIS, reflecting the phantom delay
- At full utilisation, the phantom delay is ``absorbed" into the queueing delay, and the window size is no longer too small



FAST on a DOCSIS link – cont.

Summary of results for a single flow - Total target queueing delay required by FAST as a function of link capacity

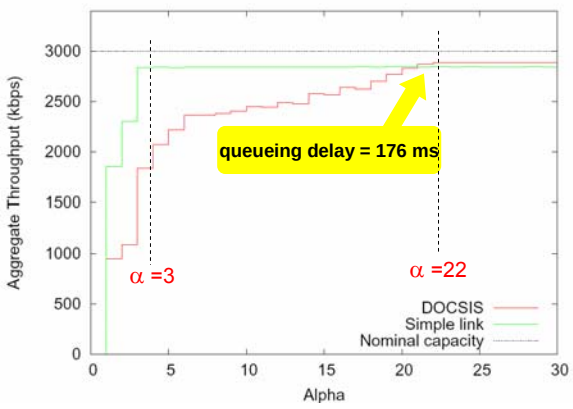
- For very low rates, full utilisation is achieved with α =1 or 2
- As the rate increases, the queueing required on a simple link decreases monotonically
- In contrast, the queueing delay required over DOCSIS increases again as the rate increases



Two Flow Results

Aggregate throughput vs. α parameter for DS = 3 Mbps and US = 512 Kbps for 2 flows with equal α (α = 1, 2…,35). Results averaged over 10 runs

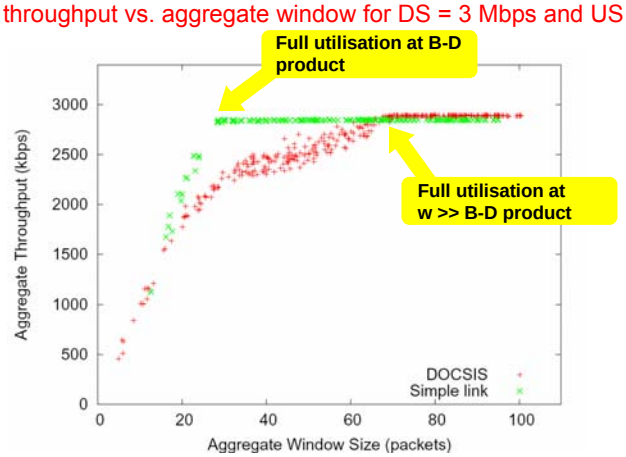
- We would expect the required α value to scale inversely with n i.e, for 2 two flows, we would expect α to halve
- E.g., for DOCSIS link and 3M/512K (we needed α=13 and q=52ms for single flow) and we expect each flow to need α=7, which would again give a total queuing delay of 52ms



For DOCSIS, the required α increased to 22! That corresponds to a total target queue size of 44 packets or a delay of 176 ms.

Two Flow Results – cont.

Aggregate throughput vs. aggregate window for DS = 3 Mbps and US = 512 Kbps



- The trend of superlinear buffer requirements is concerning, as the CMTS had a default ``traffic shaping'' buffer with maximum delay 512 ms, which can be increased to at most 1028 ms
  - the default buffer could support fewer than 512/(176/2) ≈ 6 flows at full utilisation!

Conclusions and Future work

- The performance of FAST TCP was investigated over low speed links, and in particular links running DOCSIS
- Main findings:
  - FAST is able to achieve almost full utilisation over a low speed link if its target queue size (alpha) is at least three packets
  - In contrast, DOCSIS introduces additional delay, which results in the need for a w >> B-D product i.e., requires alpha to increase as the link capacity increases beyond 1 Mbps, up to alpha=13 for 3Mbps
  - Although DOCSIS also introduces unpredictable delays, these do not appear to interfere with FAST's ability to estimate the queueing delay
- Future work:
  - Study the mechanisms by which DOCSIS introduces the additional delays
  - Derivation of an analytic model of the interaction between DOCSIS and FAST

SWIN BUR NE

CENTRE FOR ADVANCED INTERNET ARCHITECTURES

Thank You

...any questions?

SWIN BUR NE

CENTRE FOR ADVANCED INTERNET ARCHITECTURES