# Experimentally derived interactions between TCP traffic and service quality over DOCSIS cable links

Thuy T.T. Nguyen, Grenville J. Armitage
Centre for Advanced Internet Architectures
Swinburne University of Technology
Melbourne, Australia
tnguyen@swin.edu.au, garmitage@swin.edu.au

***Abstract*— We study the effect of upstream (US) and downstream (DS) rate caps and DOCSIS media access on the end-to-end performance of broadband IP services that share a typical home-access 'cable internet' link. Our experimental study measures the performance of actual DOCSIS equipment in a typical network configuration. We observe that modestly long lived data transfers in the DS direction can create substantial latency spikes (over 100ms) in the shared DOCSIS segment, even when the DS rate cap is one or two megabits per second. Such spikes can have a big impact on delay-sensitive applications, such as voice over IP (VoIP), online games and interactive streaming video that may be sharing the DOCSIS link. We also experimentally characterize the impact of US and DS bandwidth asymmetry, MTU sizes, and TCP window sizes in achieving maximum performance over DOCSIS links.**

***Keywords- DOCSIS, data throughput, cable network, TCP performance***

## I. Introduction

One of the most popular broadband Internet access technologies nowadays is DOCSIS (Data Over Cable Service Interface Specification) cable network [1]. Uptake is being driven by a range of applications– higher speed web downloads, real-time content streaming, interactive voice and video services, and online multiplayer gaming. However, the service quality implications of mixed interactive and non-interactive applications have not been fully explored.

This paper reports on an experimental study using real-world DOCSIS equipment in a lab environment. Our IP over DOCSIS testbed [2] allowed us to explore interactions between TCP flow control behavior, upstream (US) and downstream (DS) rate limits, and end-to-end latencies experienced by applications sharing a DOCSIS link. The results of our study will be useful in motivating and guiding future work on priority queuing and packet scheduling systems in both the downstream and upstream directions of DOCSIS systems.

We demonstrate that a capped link from the Cable Modem Termination System (CMTS) to a Cable Modem (CM) can introduce substantial delay to other traffic sharing the DOCSIS segment along their end-to-end path. The additional delays, in the order 100ms extra, can significantly impact delay-sensitive applications, such as voice over IP (VoIP) or interactive online game traffic, which are sharing the DOCSIS link. We also characterise the affect of US and DS bandwidth asymmetry, MTU size, the maximum size of the TCP flow control window and the internal operation of DOCSIS itself on TCP performance and the latency-jump imposed on interactive applications sharing the link.

Section II introduces some background on DOCSIS and our test scenario. Section III outlines details of our experimental setup, analysis of the findings and theoretical verification of our results. Section IV concludes with some suggestions on optimum configuration to improve the effective throughput of a DOCSIS network, and implications for mixing interactive and non-interactive traffic.

## II. Background

A typical DOCSIS broadband cable ISP access network is illustrated in Fig.1. In this scenario, home users equipment with different activities such as Web browsing, data and movie downloading, playing interactive online game and voice chat are connected to the remote content/game servers via the DOCSIS cable network of an ISP. Conceptually, the user's traffic has to travel through the user's CM, the Hybrid Fiber Coaxial Network (HFC) and the CMTS at the ISP site, and the remote links.

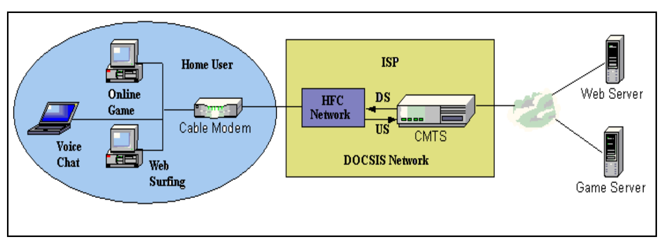


Fig.1. A typical DOCSIS cable network from ISP to home users

When a cable modem comes online (registered to the CMTS), it is assigned a Service ID (SID) that is mapped to Class of Service (CoS)/Quality of Service (QoS) parameters. These parameters specify operational limits for the cable modem, for example downstream and upstream speed limits, max-burst size, maximum number of customer's premise equipment (CPEs) per cable modem, and so on.

 

Communication between the CM and the CMTS relies on a reservation scheme, which is normally called the request/grant cycle. A CMTS periodically sends MAC allocation and management messages (MAPs) to all CMs on the network, defining the transmission availability of channels for specific periods of time. The MAP message transmission interval can be dynamic or fixed (our Cisco CMTS allowed dynamic intervals between 100usec and 2ms, or a fixed interval defaulting to 2ms). Before sending data upstream the CM must ask permission from the CMTS for a time slot to transmit. The shared request time slots in broadcast MAP messages allow the CM sending a request time for US transmission. Upon receiving the CM's request, the CMTS grants time slots according to slot availability and queues the grant message for transmission back to the CM during the next MAP transmission time. The maximum burst size of data that a CM can send upstream per MAP opportunity is limited as specified in the SID for the CM. We use the default US max-burst size of 1600 bytes. This back and forth communication mandated by the DOCSIS protocol produces additional latency into the network's performance [4][7].

We are interested in how factors such as the request/grant control of transmission cycles between the CMTS and the CM can affect the performance of a DOCSIS network - what are the impacts on different user's applications; and are there any other factors that needed to be considered in achieving optimal performance over a DOCSIS cable network.

Rather than engage in a complex theoretical modeling of the DOCSIS link access protocol, and trying to model a particular implementation, we have chosen to experimentally characterize DOCSIS performance limits using actual equipment currently used by the cable internet industry. The next section describes our test setup in detail, together with our results and analysis. It provides some insights to answer most of our questions mentioned above.

## III. TEST SETUP & FINDINGS

### A. *Impact of Downstream being a bandwidth bottleneck*

The first test that we carried out examined the impact of downstream rate caps from the CMTS to the CM, and how this affects overall TCP performance through the network.

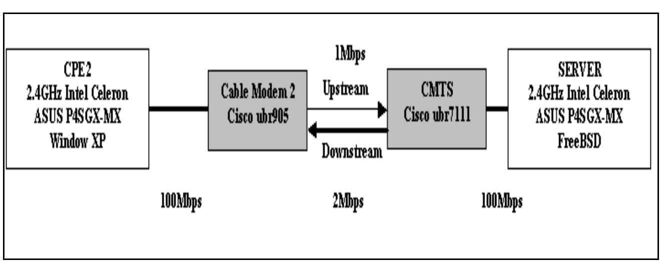


Fig.2. Test 1 setup

We used FreeBSD for our client and server hosts. To measure TCP performance we repeatedly ran nttcp [6] between a client (CPE2) and a server. We initially used the default settings to transfer 2048 buffers of 4KByte length (a total of 8 MByte) from the Server to the Client. Each nttcp trial is run repeatedly three times to ensure the accuracy of the measurements. The achieved throughput in Mbps (megabits per second) then is reported.

Our test setup is illustrated in Fig.2. We specified a DS speed limit of 2Mbps and an US speed limit of 1Mbps (to ensure the ACK rate in the US direction was essentially unrestricted). The server and client were both connected at 100Mbps to the CMTS and CM respectively.

We repeatedly ran nttcp at a number of different MTU sizes, and gathered round trip time (RTT) estimates before, during and after each run of nttcp. RTT was estimated using ICMP Echo/Response exchanges once per second from client to server (using the Unix 'ping' command). Our RTT results are presented in Fig. 3 and Fig. 4 below:

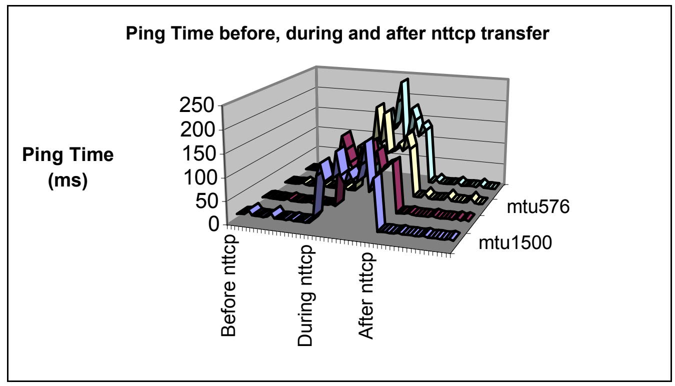


Fig. 3. Ping Time before, during and after nttcp transfer with different MTU sizes

The presence of a modest burst of traffic in the DS direction causes a dramatic increase in RTT (with the actual increase depending slightly on MTU). When the link is essentially idle the RTT is approximately 13ms. During the nttcp transfer phase it jumps to approximately 120ms (an average of 118ms with MTU 1500 bytes, 119ms with MTU 1250 bytes, 123ms with MTU 576 bytes and 137ms with MTU 512 bytes).

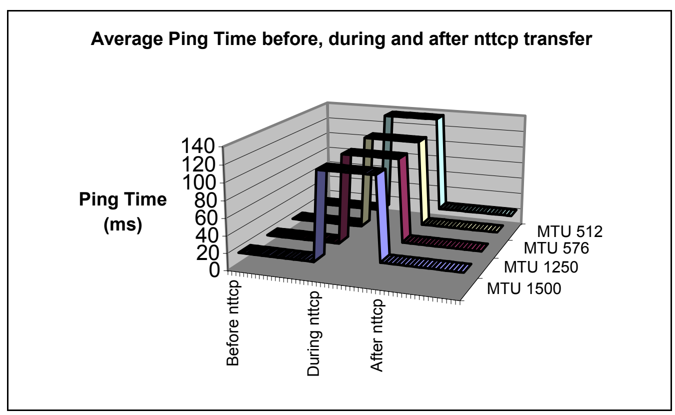


Fig. 4. Average Ping Time before, during and after nttcp transaction with different MTU sizes

We then reduced the rate at which nttcp pushed data at the

CMTS in order to characterize this increase in RTT as a function of offered load. FreeBSD's kernel-resident 'dummynet' module was used at the server side to rate limit TCP traffic associated with nttcp. We set dummynet's internal queue limit to 62Kbytes and the packet scheduler was configured for bandwidth limits between 200Kbps and 3.2Mbps.

Only server to client TCP packets were rate-limited by dummynet - upstream ACK packets and ICMP packets in each direction were unrestricted. We retained the 1Mbps US rate cap, kept the MTU fixed at 1500 bytes, and repeated our nttcp and ping tests for DS rates of 500Kbps, 1Mbps, 1.5Mbps, and 2Mbps. For each DS rate cap we also ran nttcp with different maximum TCP window sizes (from 2Kbytes to 62Kbytes) to explore the interaction between RTT growth and window size for optimal TCP performance.

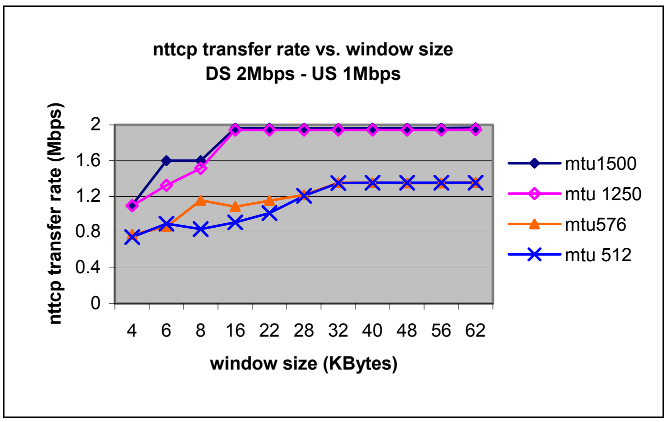


Fig.5. nttcp transfer rate vs. window size

The increased RTT under load affects the maximum TCP receive window required for optimal throughput (in theory the RTT multiplied by the bottleneck bandwidth, or 'bandwidth delay product'). Our experimental results in Fig.5 show how the nttcp transfer rates peak for maximum receive window sizes of roughly 22Kbytes with MTU sizes of 1500 bytes and 1250 bytes and roughly 32Kbytes for MTU sizes of 576 and 512 (DS and US were fixed at 2Mbps and 1Mbps respectively for these tests).

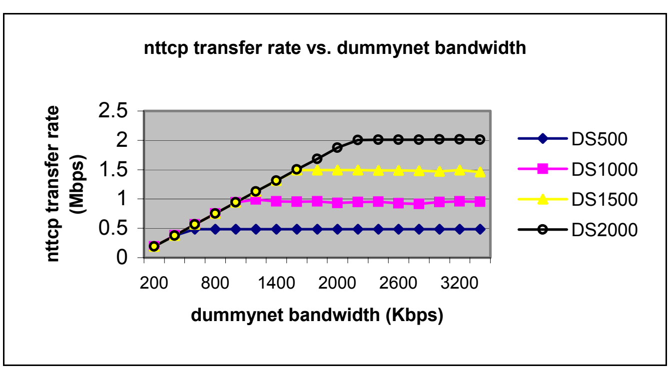


Fig.6. nttcp transfer rate vs. server rate limit and DS rate cap

All subsequent nttcp tests were run with the maximum receive window set to 48Kbytes (unless explicitly noted otherwise). Fig.6 confirms nttcp's best transfer rate is now controlled by server-side dummynet (on the diagonals, when less than the DS rate cap) and the DS rate cap (on the horizontals, when dummynet is set higher than the DS rate cap).

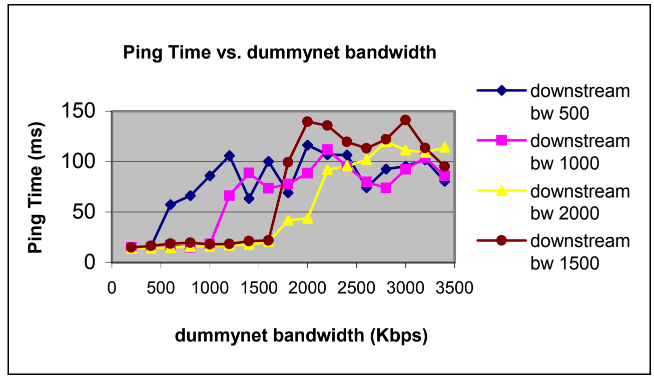


Fig.7. Ping time vs. server rate limit for various DS rate caps

Fig.7 shows the maximum ping time during an nttcp transfer for different server-side rate limits. We see that for each DS rate cap the RTT increases dramatically from approximately 13ms to over 100ms at the point where the server's offered load begins to exceed the DS rate cap (at 500Kbps, 1Mbps, 1.5 Mbps and 2Mbps respectively).

The interaction between offered DS load, RTT and maximum receive window also means that varying the client's maximum receive window varies the amount by which the RTT increases under load. Fig.8 shows the maximum RTT as the receive window is varied from 2Kbytes to 58 Kbytes, the server-side bandwidth limit ranges from 200Kbps to 3.4 Mbps and the DS rate is capped at 2Mbps. As the window size increases the effective load on the CMTS increases, thus increasing the RTT and increasing the receive window required for optimal nttcp performance.

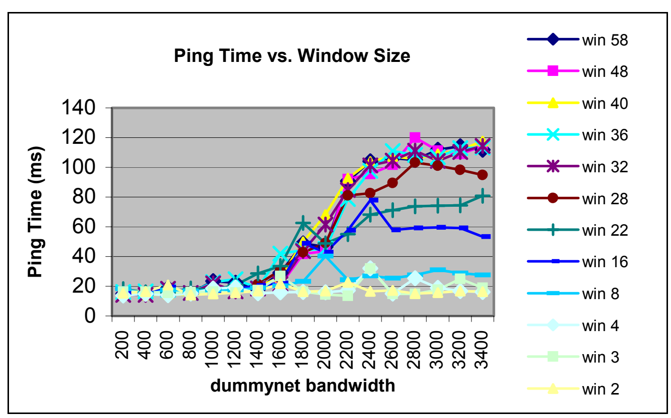


Fig.8. Ping Time vs. Window Size

Despite the relatively high downstream capacity of DOCSIS it is clear that potential exists for some rather unexpected end-to-end behaviors. The spike in RTT over the DOCSIS link affects all traffic sharing the particular CM and has significant real-world implications. For example, consider an ISP hosting local content servers on their 100Mbps or 1Gbps backbone and encouraging their directly attached 'broadband' customers to download locally rather than from distant servers. Such customers are likely to discover their RTT to other parts of the Internet jumping up by over 100ms during the local content transfer phase. Although probably not noticeable if the customer isn't doing anything else at the time, the RTT jump will be highly disruptive if the customer site was attempting concurrent interactive Voice over IP or online gaming.

It is also clear that configuring the optimal window size based on the RTT of an idle DOCSIS link would provide a highly sub-optimal result. While most customers are unlikely to be manually tweaking their operating system's TCP window sizes, those who do are quite likely to complain to their ISP's helpdesk. It is worth knowing about the RTT jump if only to help inform such customers about what they should expect and why it is normal.

### B. Impact of Asymmetric Upstream and Downstream bandwidths

The next factor that we consider is asymmetry in the DS and US rate caps. Most ISPs typically assign a DS rate cap far greater than the US rate cap, which can actually result in reduced TCP performance towards the customer site (client). When a client is downloading content from server's beyond the CMTS, the rate of data packets towards the client is limited by the rate at which ACK packets can be returned to the server. The ACK packet rate is constrained by the US rate cap set by the ISP.

If the potential ACK rate is high enough, the overall TCP performance limit will be set by the DS rate cap (or possibly other bottlenecks along the path). If the ACK rate is sufficiently restricted the overall TCP performance limit will be entirely due to the US rate cap.

The optimal DS/US ratio depends on the typical data packet size, the ACK packet size and the number of data packets acknowledged by a single ACK packet (d).

When

$$\frac{DS}{US} \leq \frac{DataPacketSize}{ACKPacketSize} * d \qquad (1)$$

the upstream channel supports enough ACKs per second that TCP behaves as though it is running over a symmetric path, and the DS rate limit will dominate overall performance.

However when

$$\frac{DS}{US} > \frac{DataPacketSize}{ACKPacketSize} * d$$

the upstream channel cannot support enough ACKs per second required to fill the downstream channel, leading to sub-optimal TCP performance in the server to client direction.

According to this analysis a DOCSIS cable network will appear to have optimal performance if the DS and the US speed limits in the DOCSIS configuration satisfy (1) above. Failing to satisfy this condition might result in the problem that customers cannot achieve download speeds consistent with the advertised DS rate cap.

To verify the effect of DS/US ratio with different MTU sizes on a DOCSIS link, we carried out a test using nttcp to transfer data from the server to the client. DS link is capped constantly at 2Mbps, while US's cap varies from 16kbps to 1Mbps for the CM. The test is repeated with different MTU sizes from 512 bytes to 1500 bytes. TCPdump is run at the client during the test.

In principle TCP should send an ACK for every one or two data packets. Output of TCPdump file shows one ACK packet per alternation of one and two data packets consistently throughout the nttcp transmission. It results in an average ratio of data packets per ACK packet (d) is consistently equal to 1.5 with all window sizes varies from 2Kbytes to 62 Kbytes with no packet loss and any signs of retransmission.

Results confirmed that the network's effective throughput is highest when the DS/US ratio exceeds optimal values

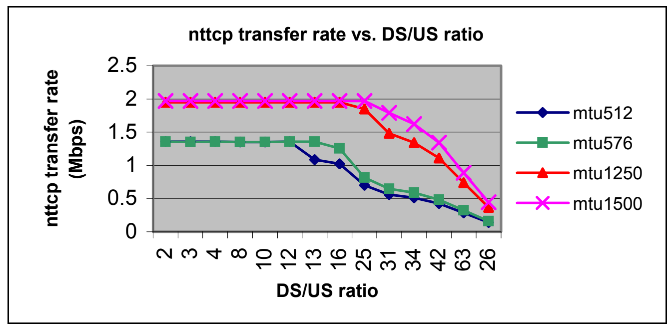


Fig. 9. nttcp transfer rate vs. DS/US ratio with different MTU sizes

calculated in (1) with d = 1.5. Fig. 9 shows that this optimal value varies depending on the MTU size used, ~ 31 for MTU 1500 and 1250 bytes, and ~ 12 for MTU 576 and 512 bytes.

### C. Impact of TCP, IP and DOCSIS overhead

The final factor that we investigate is the TCP, IP and DOCSIS protocol overheads in contribution to the overall data transfer throughput. It would be useful to understand the different between the theoretical data throughput and the actual data transfer rate, when we take into account all transmission overhead in our calculation.

We call the DS and US speed limits (in Kbps or Mbps) configured at the CMTS (and imposed on the CM) the *theoretical throughput*. An *effective data throughput* refers to only the transfer speed of user's payload. It excludes all packet headers and overhead of the network maintenance and provisioning.

Since the effective data throughput does not take into account any packet headers and monitoring traffics overhead, it is not very useful to compare the effective data throughput and the theoretical one. We therefore calculate another parameter, called *expected throughput*, which not only takes into account the actual user's payload, but also all packet headers, including Ethernet header, TCP/IP and MPEG-2 headers, and DOCSIS network mantenance and provisioning overheads.

The expected throughput is calculated as following:

*In terms of packet header overhead:*

Packet size = Payload + headers (Ethernet + MPEG-2 + TCP + IP + FEC + DOCSIS)

- TCP/IP header is 40 bytes and Ethernet header is 18 bytes. There are 6 bytes of DOCSIS overhead. That could be a total of about 4.1% (for 1500-byte packet) to 11.1% (for 512-byte packet) packet header overhead.
- DS cable physiscal layer uses ITU-J83 Annex B indicates Reed-Solomon FEC with a 128/122 code, which means 6 symbols of overhead for every 128 symbols, hence 6/128 = 4.7%. MPEG-2 is made up of 188 byte packets with four bytes of overhead, sometimes five, giving an average of 4.5 bytes overhead per 188 byte packets, giving 4.5/188 = 2.4% [4].

Hence in terms of packet header, there are from 11.2% for 1500-byte packet to 18.2% for 512-byte packet overhead.

*In terms of communication overhead of DOCSIS network maintenance and provisioning*:

*MAPS:* Downstream throughput is also reduced by the transmission of MAP messages sent from the CMTS to CMs in the request/grant cycle. A MAP message is normally sent every 2ms, which equals to 500MAPs/sec. If the MAP takes up 64 bytes, that would equals to 8 bits/byte * 500 MAPs/s = 256kbps. Therefore, with this calculation, 256kbps of the DS throughput would be used for MAPs transmission.

Upstream throughput is also limited by MAP request/grant cycle. A MAP interval of 2ms results in 500 MAPs per second. If a CM requires two MAPs (for transmission time request and grant) per upstream IP packet this equates to 250 PPS (packets per second) in the upstream direction. For upstream data transfer this represents a limit of roughly 3Mbps (assuming 1518 byte Ethernet frames) [3][4]. For downstream data transfer this represents a limit of 250 ACK packets per second. Given that modern TCP stacks transmit between one and two data packets for every ACK, we infer a limit of 250 to 500 PPS in the downstream direction - again, regardless of the actual bits/second limit assigned to the DS or US channels.

## IV. Conclusions

DOCSIS-based cable Internet services are being deployed around the world, promising higher speeds and better overall performance to consumers. However, the real-world interactions between TCP-based applications, ISP-configured upstream and downstream bandwidth limits, and end-user perceptions of overall system performance are not trivial to derive. We have experimentally characterized these relationships using a DOCSIS cable system in a test-bed designed to simulate a typical customer attachment to an ISP offering content from local servers on the ISP's network.

We have shown that, even with generous upstream, bandwidth allocations, modest, ISP-imposed rate caps on the downstream channel can results in substantial increases in latency (upwards of 100ms) over the DOCSIS link when a client downloads content from ISP-hosted local servers. All traffic heading towards the client's network experiences this additional latency, including any real-time or interactive applications such as IP telephony, conferencing, or online gaming that may be concurrently sharing the cable modem connection. ISPs should be aware of this behavior in order to better manage customer expectations. ISPs might also consider this behavior a good reason to introduce priority queuing schemes in their CMTS equipment in order to better support concurrent use of interactive and non-interactive applications by their customers.

Other factors that affect end-to-end TCP performance over the DOCSIS network were also investigated, including upstream and downstream bandwidth asymmetry, MTU sizes, the size of the maximum receive window and the overhead of communication over a DOCSIS network. For example, the CMTS works best if the upstream and downstream speeds are limited satisfied the following equation:

$$\frac{DS}{US} \le \frac{DataPacketSizeInBytes}{ACKPacketSizeInBytes} * d$$

(where the factor d is the number of data packets per ACK packet, and depends on the receiving TCP implementation.)

Bigger MTU size provides an opportunity of achiving higher data transfer throughput, and network users should calculate optimal receive window sizes using RTT of a loaded (rather than idle) DOCSIS link.

Our experiments show that all the above factors must be considered in order for ISPs and customers to predict and achieve optimal Internet application performance. Our results provide useful insights into real-world behaviour of end to end Internet paths containing DOCSIS links.

## Acknowledgment

We thank Cisco Systems Australia for supporting our work at the Centre for Advanced Internet Architectures through equipment donations and post-graduate student stipends.

## References

[1] CableLabs, “Data-Over-Cable Service Interface Specifications Radio Frequency Interface Specification SP –RFIv1.1-I01-990311”,1999

[2] Broadband Access Research Testbed http://www.caia.swin.edu.au/bart

[3] “Troubles Shooting Slow Performance in Cable Modem Networks” http://www.cisco.com/en/US/tech/tk86/tk89/technologies_tech_note09186a00800b123c.shtml

[4] “Understanding Data Throughput in a DOCSIS World” http://www.cisco.com/en/US/tech/tk86/tk168/technologies_tech_note09186a0080094545.shtml

[5] Balakrishnan, H, Padmanabhan, N. V, Katz, H. R, “The effects of asymmetry on TCP Performance”, Balzer Science Publisher BV

[6] http://www.leo.org/~elmar/nttcp/

[7] http://www.cisco.com/en/US/products/hw/cable/ps2209/products_configuration_guide_chapter09186a008018011e.html