Simulation of an advanced QoS protocol for mass content

Thomas Dreibholz, Avril IJsselmuiden
Institute of Experimental Mathematics, University of Duisburg-Essen, 45326 Essen, Germany
***E-mail**: {dreibh,avril}@exp-math.uni-essen.de*

John Adams
British Telecom, Martlesham Heath, Suffolk, UK
***E-mail:** john.l.adams@bt.com*

Abstract

This paper describes a new network device to be located in network edge nodes. The device can deal with congestion conditions that may arise when, for example, a home or SME customer requests too many simultaneous flows to be forwarded down a DSL link or other access technology. It provides a solution to guaranteeing certain flows that are forwarded along one or more congested links, by making others (typically the latest flow, or another flow selected because of policy reasons), the subject of focused packet discards. The functionality of the device is described, and results from a fast-track simulation model implementing a lightweight version of the device, developed in LISP, are presented here.

1. Introduction

Broadband services delivered over DSL to residential customers have been the focus of much interest recently. Opportunities exist for services such as TV distribution, combined with voice and data services, which allow customers to select from a number of differently priced packages. Some of these packages may rely on a QoS function controlling the aggregate mix of services forwarded to each customer; the function would protect certain high priority flows, which could be pre-selected by the customer.

We have developed a device which implements this functionality[1], and it is currently under consideration by standards bodies ITU-T and ETSI[2][3][4].

After initial discussion of the proposal, the standards bodies requested simulation results. In order to fulfill this requirement, the device was implemented in LISP, as a lightweight version, and in Opnet Modeller, with the full functionality. The reason for this was that as the full functionality was fairly complex, and would take careful construction in Opnet, we decided to use LISP to build a fast track model to show that even using a lightweight protocol was preferable to using no protocol at all in congestion conditions. Additionally, by using two models, we can make an initial cross-validation. After cross-validation, the Opnet model can be used to provide simulation results for the standards meetings [6], whilst the LISP model can be used to investigate security solutions[7].

In Section 2, we give the background to the problem; in Section 3 we describe some Quality of Service (QoS) issues. The device itself is presented in Section 4, and its LISP implementation is described in Section 5. In Section 6 we give details of the experimental testbed, with results in Section 7. Finally, Section 8 gives the conclusions.

2. Edge Nodes

Edge nodes exist in a network and channel all content from service providers towards customers (Figure 1). While this could be achieved using a separate ATM VC for each service type (TV, voice, and data), this is very complex if it is extended so that, for instance, the data VC is no longer a single VC but separate VCs for several types of data. In particular, web streaming would need a separate VC if its QoS is to be treated differently to other data types. Therefore, it is advantageous if all flows are aggregated onto one, or at most two ATM VCs. For example, one VC may carry best effort traffic, and the other VC is for all QoS content. Such a separation allows the ATM layer to manage cell discards (favouring QoS content over best effort) if there is a cost advantage in doing this. The existence of a group of QoS content VCs implies that the IP layer has to handle any contention that may occur among the QoS content flows.

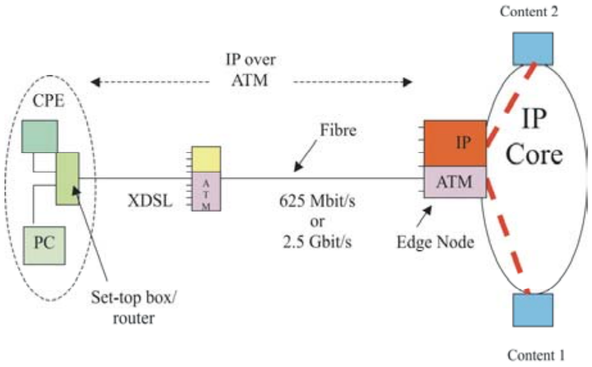

Figure 1: Broadband service components

Several vendors have developed equipment, in the form of Edge Nodes, that channel services using separate ATM VCs and several vendors are now considering how they can move to IP-based multiservice aggregation. The device proposed in this document relates to an improved Edge Node that operates in conjunction with essentially functionally similar equipment to that currently existing, except for a modification to the set top box. The modification enables the box to recognise certain new alarm signals created by the device described here.

3. Quality of Service (QoS)

The scenario of an end user that can connect to, potentially, many thousands of content sites and purchase QoS-sensitive content causes a reconsideration of the QoS set-up and clear-down procedures.

Users may get QoS-sensitive content in different ways:

1. From an ISP product where the end user can see Internet content and, possibly, a content portal managed by the ISP. The ISP may provide QoS guarantees only on content purchased from within the portal and the content suppliers would settle directly with the ISP using such information as the number of hits, duration, etc.
2. From the network access service provider, if they offer direct access to the Internet (i.e. the access provider assigns an IP address to the end user from its pool of addresses and the user selects services directly from different sites). It may be the case that the access provider also has content portal and only offers QoS guarantees on content selected from this portal.
3. From all content sites, looking for niche content where a site has established a reputation or general content where a site is offering a highly competitive price. In this model the user is not just looking for QoS guarantees on specific portal content but, more generally, on any QoS-sensitive content.

The protocol developed and described in this paper addresses the third point above. If the end user has a direct internet access product from the network access provider, then there has to be a realistic commercial model that underpins the QoS guarantees sought by the content sites, so that its content is viewed favourably. Duration charges truly reflect the actual duration of the content. This implies that the network access provider (who is the source of bills to the end user) takes steps with an untrusted content site to ensure that QoS guarantees are not charged for after the content flow has ceased.

It is this which has prompted QoS control procedures that could be applied by an access provider given that:

- It is dealing with untrusted content sites (who may "forget" to send a call cessation signal or may be incapable of keeping dynamic call record information on calls in progress)

- It is nevertheless the source of bills to the end user and needs to ensure that bills are fair and truly reflect what was consumed.

We are proposing that the answer to these issues is a signalling protocol that puts minimum demands on the content sites and more controls in the access provider network. It actually applies to the ISP-centred model as well as the direct access model, since the ISP is also in a dilemma about how to pass on charges to the user given that the content site is an untrusted element.

4. The QoS Device

The device described in this paper would modify and improve the above proposed shaping function in an Edge Node, but could operate equally well other places in the network. Currently, when flows consist of different priority information, such as video and data, shapers use schemes such as Type of Service marking, to distinguish flow content, and discard packets based upon the content[8]. Our device addresses the problem of *equal priority* flows causing congestion and unable to slow down through the control of, say, TCP.

An Overview

The device works on the principle that if congestion occurs, and packet discard is necessary, it is better to use focused discard than arbitrary discard. This principle, applied at the ATM cell level, was first suggested in [9] and its value has been shown in other publications since [10][11]. This principle is applied in our device by making the latest flow(s) the subject of discard. In order to perform this, the device has to know when a new flow begins, and has to maintain a record of it.

The device recognises that a new flow has started, because the flow is (in normal conditions) always preceded by a Start Packet. We propose a very simple “signalling” protocol consisting of a *start packet* appeneded at the head of a flow of packets. This acts to eliminate SPAM by indicating that the receiving user has added a security key (to the start packet) that can be read by the network, and verified that the user wants this flow. This requires support from IP6.

The start packet would be recognisable to the QoS device, and so the device would know that a new flow had started. Having sent its start packet, a flow may immediately start transmitting actual data packets. The basic principle is that the Start Packet contains flow ID information (such as the IP header fields of the subsequent data packets) that is extracted and recorded by the device. Subsequent data packets are examined and are able to be identified as belonging to that flow.

The device maintains a vulnerable flows window, where flow IDs are stored. When a new flow starts, it enters this window, and whilst there, it is regarded as being the target of packet loss if congestion occurs. As new flows start up, the flow moves through the window, until eventually it is removed from the window (by being overwritten), when the following conditions are satisfied:

- The sum of the rates of the flows in the window, minus rate of the oldest flow, is greater than *N,* where *N* is a percentage of the total bandwidth.
- The flow has been in the window for at least time *T*, OR,
- it has received at least *n* packets after the start packet.

When a packet's ID is removed from the vulnerable flows window, it becomes a guaranteed flow, except under certain extreme traffic conditions to be discussed below. This means that, normally, there are no packets discarded from such a flow when the buffer starts to experience congestion.

If, over an interval of time, a sequence of flows start with their corresponding start packets, then the normal behaviour of the QoS device allows some of the earlier flows to move to the guaranteed area, while always retaining at least one flow ID, whose packets will be the subject of focused discard if the buffer becomes too full.

When a packet is deleted, the QoS device may send another new control packet forward towards the customer, an Alarm message. This advises the application resident in the customer's receiving equipment that a network congestion condition has occurred. An application may choose to continue receiving such data packets that are not deleted, or may close down and indicate network busy to the user.

5. LISP Implementation

Our device has four functionally separate blocks, as shown in Figure 2. Flows are classed as either being in the drop window, or being in the guaranteed area.

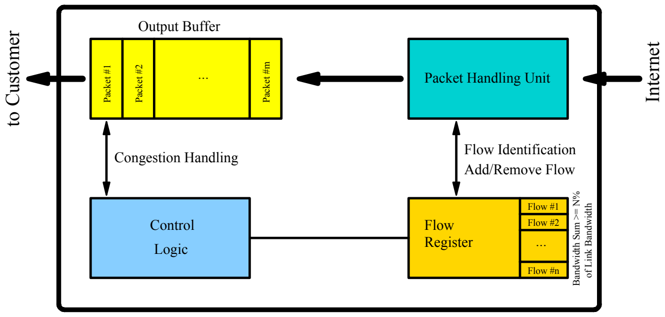

Figure 2: The functionality of the edge device

Packet Handling Unit

When a packet arrives at the packet handling unit, the flow register is queried to identify the flow ID. If a flow ID is found, a reference to this flow identity is mapped to the packet. This also identifies the packet as belonging to a drop window flow. If the flow identity is not found, then the packet belongs to the guaranteed area traffic. After identification, the packet is moved to the output buffer.

Output Buffer

The output buffer is mainly a leaky bucket buffer of limited size that stores incoming packets before moving them onto a link to a traffic sink (the customer requesting the traffic) with a given maximum output bandwidth.

Control Logic

When a new packet has to be appended to the output buffer, a check is first made to determine whether there is sufficient space in the buffer. If so, the packet is appended; else, a configurable *dropper procedure* (explanation follows below) is invoked. If this procedure is able to gain sufficient space within the buffer, the packet can be inserted, otherwise, it is dropped. If the packet dropped belongs to a drop window flow, we record this as a *drop window drop*, otherwise, it is recorded as a *guaranteed area drop*.

The dropper procedure we use is quite simple: if the packet for which space has to be gained is a drop window packet, it is simply discarded (as it belongs to a drop window flow, and so is targeted for discard in the case of congestion). However, if the packet belongs to a guaranteed area flow, the dropper mechanism traverses the output queue and drops every drop window packet, until sufficient space is gained or it reaches the end of the buffer. Then, the new guaranteed area packet can be inserted or dropped accordingly.

Flow Register

The flow register is responsible for maintaining the drop window. This is where the flow identity of a new flow is stored, initiated by a start packet arrival at the beginning of the flow. When a new flow arrives (that is, its start packet is processed), its identity is added to the window, and the register must calculate whether any existing flows are able to leave the window. It does this by checking whether the total sum of rate advisories of the flows in the window is greater than a configured window size N (which is a configured percentage of the output link bandwidth). If so, it calculates whether this condition is still met when the rate advisory of the earliest arrived flow is subtracted. In this case, it is allowed to leave the drop window and becomes a guaranteed area flow. This procedure is repeated until the sum of the rate advisories, minus the rate advisory of the earliest arrival, is less than N.

6 Experimental Testbed

Our simulation model has been implemented using an Open Source simulation package under BSD licence written in Common LISP [14]. This package has been used to get a model up and running in a relatively short time, due to its simplicity when compared to tools like Opnet and NS2. The model was designed to run with the protocol turned on, and turned off, by setting a runtime parameter. When the protocol is off, no dropper strategy is used, and packets arriving are discarded indiscriminately in the case of congestion.

The device was set up with 5 traffic generators attached, as shown in Figure 3;

- the reference flow: a multimedia source (flow 1) of 256 Kbit/s using 30 frames per second of 1060 bytes,
- 2 voice sources (flows 2,3) of 64 Kbit/s each (one using 80 frames per second of size 100 bytes, the other using 8 frames per second of 1000 bytes),
- a multimedia source (flow 4) having the same parameters as flow 1 and
- a multimedia source (flow 5) of 512 Kbit/s using 50 frames per second of 1280 bytes.

For each source, the deviation of the frame interarrival time is +/- 25% (randomly chosen, using uniform distribution). Each frame is segmented to packets having sizes of at most 1000 bytes. The link bandwidth is 1 Mbit/s. The edge device uses an output buffer of 100000 bits (therefore storing at most 100ms at 1 Mbit/s output speed) and a drop window size of 25% of the link bandwidth (that is: 250 Kbit/s).

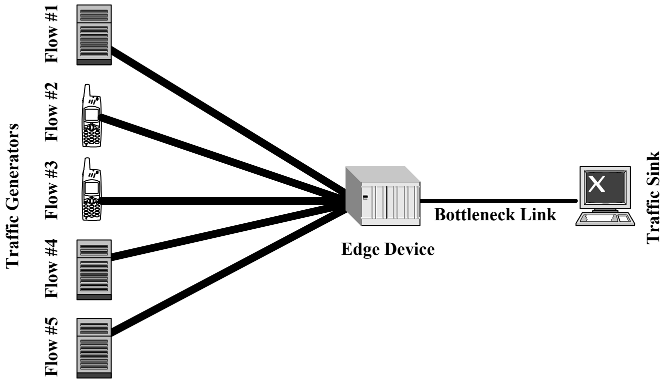

Figure 3: Traffic generators attached to the Edge Device

7 Results

A number of experiments were performed, using confidence intervals of 95%. Our first experiment was performed with the protocol turned off, in order to verify that packet loss would be indiscriminate, and spread across all 5 streams.

Figure 4 shows the percentage loss rate of the total packets transmitted, for each stream; it is indiscriminate in that all 5 streams have lost packets. Except for flow 2, the loss rates have values between 13% and 22%. The reason for the significantly lower loss rate of flow 2 is that its frame size is only 100 bytes, while the other flows use frame sizes between 1000 and 1280 bytes (frames are segmented to packets of at most 1000 bytes). Clearly, inserting a small packet into a partially filled output buffer is usually more successful than inserting a large one. However, the most important result is that all five flows experience some reduction in service quality, due to significant packet loss.

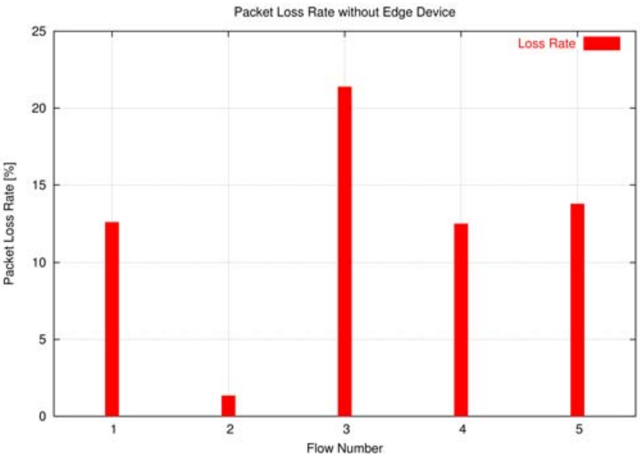

Figure 4: Packet loss with the protocol turned off

The next step was to run simulations with the protocol turned on. Flow 1 is the reference stream (in the drop window) and Flows 2–5 were in the guaranteed area. Figure 5 shows the resulting percentage packet loss rate using a $\log_{10}$ scale.

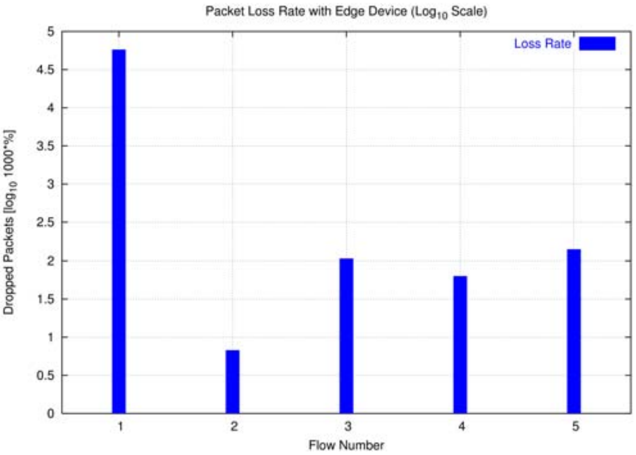

Figure 5: Packet loss with the protocol turned on (% *1000 converted to log scale)

Obviously, the loss rate for the reference stream is highest: about 58%. For the other flows, the loss rate is about 0.1% or less. This clearly shows that even using a very simple edge device protocol, it has been possible to achieve a quite low loss rate (about only 1 of 1000 packets has been dropped) for four of five flows, with the heaviest loss targeted against one flow. We can translate this to say that four of five streams may notice no significant reduction in service quality, and only one stream has a significant reduction. Clearly, in terms of customer satisfaction, this is far better than all streams being unsatisfactory!

However, it should be noted that flows 2–5 do not receive the same QoS as in the case of a congestion-free network: in the case that there is currently a packet of the reference flow being transmitted, and the output queue becomes filled, there is no possibility of suspending this transmission (non - pre-emptive transmission). So, if the dropper algorithm is not able to gain space for a guaranteed area packet by dropping other packets of the reference flow in the output buffer, this results in packet loss from guaranteed flows.

8. Conclusions

In this paper, an advanced QoS protocol for mass content has been described. The protocol is suitable for controlling congestion in network edge devices. A lightweight version of the protocol has been implemented in a LISP simulation model, and this was described, along with some results. The initial results show that even a lightweight protocol is better than using no protocol at all. The research now splits into 3 tracks: this model is further used to investigate security mechanisms; an Opnet model is used to implement the full device functionality and provide simulation results to the Standards Bodies, ITU-T and ETSI; and a third model is currently being developed as a student project, in Omnet++.

References

[1] J.L. Adams & A.J. Smith**.** European Patent Application No EP 01 30 5209. *Packet discard control for broadband services* Issue June 2001

[2] British Telecom Contribution (A. IJsselmuiden & J.L. Adams). *Proposal for a new IP transfer capability and future QoS studies.* ITU-T Contribution. D-, Q4, 6,10,11,16, SG13, Geneva, Feb. 2004.

[3] British Telecom Contribution (A. IJsselmuiden & J.L. Adams). *Delivery of assured QoS content in NGNs.* ITU-T Contribution. D-, Q4, 6,10,11,16, SG13, Geneva, Feb. 2004.

[4] British Telecom Contribution (A. IJsselmuiden & J.L. Adams). *Delivering QoS from remote content providers.* ETSI contribution. TISPAN#01(03)TD132, Sophia Antipolis, Sept. 2003.

[5] A. IJsselmuiden. "*From SDLs to Opnet Model*", Opnetwork 2004, Washington DC, Aug.-Sept. 2004.

[6] A. IJsselmuiden & J.L. Adams. *Description of the QoS device.* ITU-T contribution scheduled for December 2004.

[7] T. Dreibholz, A. Smith & J. Adams. *Realising a scalable edge device to meet QoS requirements for real-time content delivered to IP broadband customers.* 10th Int. Conf. On Telecoms, (ICT 2003), Feb 2003, Tahiti, Papeete.

[8] S. Wright, T. Anschutz, & E. Shrum. *Opnet simulation of DSL framework.* Opnetwork 2003.

[9] A.J. Smith, J.L. Adams, C.J. Adams, & A.G. Tagg. *Use of the Cell Loss Priority Tagging Mechanism in ATM switches.* Proc. ICIE '91, Singapore, Dec. 1991.

[10] S. Floyd & V. Jacobsen. *Random Early Detection gateways for congestion avoidance.* V.1, N.4, August 1993, p. 397-413. http://ftp.ee.lbl.gov/floyd/red.html

[11] K.Kawahara, K.Kitajima, T.Takine & Y.Oie. "*Performance evaluation of selective cell discarding schemes in ATM networks".* Infocom'96, pp. 1054-1061, 1996.

[12] T. Anschutz (Ed). *DSL evolution – architecture requirements for the support of QoS-enabled IP services.* DSL Forum, Working Text 81, rev. 7., May 2003.

[13] S. Wright, T. Anschutz. *QoS requirements in DSL networks.* Globecom 2003.

[14] http://sctp.fh-muenster.de/sim