```
and so on until we find the new operating point
```

So for ln(bandwidth*D) time we operate below the capacity we could get. In fact, each time a connection starts up that loses more than a threshold number of packets for other ABR connections, all connections lose this much.

If the number of ABR VCs is large w.r.t the number of packets in the pipeline, and the arrival rate of new flows is low, then the loss will be shared out as perhaps below the "Fast Retransmit" threshold. However, at modest numbers of ABR VCs, or high flow arrival rates, or paths with lower numbers of packets in the pipeline (which will be the case for modest haul terrestrial paths), then this will incur serious drop in throughput.

7 Summary and Conclusions

The Internet community and the ATM communities are working towards accommodating the types of traffic that were each others' main remit. In this process, the missing middle should be addressed, where a path consists of mixed Internet and ATM portions.

In this paper, we have looked at the effect on an Internet style end-to-end congestion avoidance scheme of a path that partially includes the proposed ATM based ABR service. We believe that there will be a sever performance problem, and we propose a simple change in the ABR service to improve matters.

In future work, we will look at the effect of including internet hops in paths to carry predictive or guaranteed service traffic.

References

[1] David D. Clark, Van Jacobson, John Romkey, Howard Salwen. An analysis of TCP processing overhead. *IEEE Communications Magazine*, June 1989, pp. 23-29.

[2] David D. Clark, David L. Tennenhouse. Architectural Considerations for a New Generation Protocols. *Computer Communication Review,* Vol. 20, No. 4, SIGCOMM '90, September 1990, pp. 200-208.

[3] D. Clark, S. Shenker, and L. Zhang. Supporting real-time applications in an integrated services packet network: Architecture and mechanism. ACM, pages 14-26, 1992.

[4] H.T.Kug, Robert Morris, Thomas Charuhas, Dong Lin, Use of Link-by-link Flow Control in Maximising ATM Network Performance: Simulation Results Presented to the ATM Forum, May 1994.

[5] ATM Forum Rate-Based Traffic Management Working Group Rate-Based Traffic Management Algorithm Submitted to ATM Forum, April 1994.

and many more...

Since the queues are mainly going to be outside the VCs (in routers around the edge of the cloud) we do not have an opportunity to do the optimum, which would simply to be to apply Random Early Drop over the queue (this penalise bursts in exactly the right proportion to the amount they are bursty).

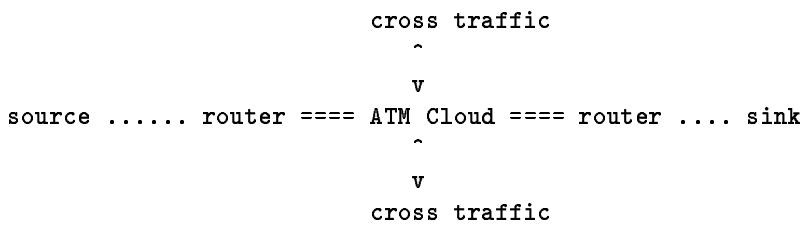
Cooperation is defined in terms of the time to respond to a change - obviously, this should be in the timeframe of 1 round trip time:

Problem: We don't know RTT...for complete path (ok, so estimate...)

Rule applied is based on Hofstadter's famous first strike analysis - basically disruption is only punished after 2 misbehaviours, and cooperation is rewarded after 1.

Why multimedia: well, as per Wakeman's analysis, you can take VBR, and divide it into required quality CBR, required quality VBR, *and* optional (ABR) CBR and VBR portions, so all of the above applies...

6 Informal Analysis



Assume that we have some number of sources, impinging on the ATM cloud, in the type "IP" paths, defined to be n.

If a new source starts, it is immediately given 1/n of the bandwidth, and the rest incur a step in delay. This step will be 1/n of the delay across the ATM portion of the path, if they all share the same path (and all have the same bandwidth allocated, as ABR will have).

Taking this as a simplifying assumption (future work will relax this), then define the delay across the entire path to be D, and the ATM portion to be d, then there will be d/D of the packets in flight outside the ATM hop.

At some point later, these packets will arrive at the prior hop and exceed its capacity, causing loss.

Currently TCP (and similar end system protocols such as the VBR+CBR algorithm in Inria's IVS video codec), have two modes of behaviour after packet loss:

- Fast Retransmit - if loss is perceived through duplicate acknowledgements at the sender, it switches to half the send window, but continues probing for extra bandwidth.
- Slow Start - if loss is greater than a small threshold of packets per window, then it backs off to 1 packet per RTT, and increases at 1 packet per acknowledgement til it reaches half the previous send window, then at 1 packet per RTT til it hits loss again.

Then all sources will go into a synchronised slow start. This will happen at every connection start up. The arrival rate of new flows that share the ATM portion of a path is l.

Assume, for the moment that connections are relatively long lived to the connection arrival rate and the RTT.

Then, each slow start costs us in lost performance:

```
1 RTT at 1 packet per round trip time
1 RTT at 2 packets per round trip time
1 RTT at 4 packets per round trip time
 ...
```

source, in the impression that a packet has not reached the destination - if the ATM path is a long delay, then the RTT estimation used by TCP and TP4 may well be very close to the ATM hops' RTT before the change.

2. The increase in buffering results in more data being queued when the end system decides that because of the delay a loss has occurred. This means that if congestion avoidance occurs in the sender, the amount of data "wasted" is increased by this scheme.

Both of these problems can be blamed o nthe same misunderstanding of connectionless services. Any hop-by-hop recovery or preventive scheme will interfere negatively with the end-to-end congestion/bandwidth-discovery scheme, by hiding information.

To understand this clearly, simply picture the TCP pushing against the upper bandwidth limit one packet per RTTs worth of packets. If there is an unobscured change in bandwidth, then TCP discovers this in one RTT. Any scheme that buffers more packets in an attempt to hide loss simply increases the time to discover the problem, but exacerbates it in doing so. In fact, the loss is merely pushed out to the edge of the ATM hop. Without explicit congestion notification (an anathema to ABR anyway) the routers at the edge of an ATM cloud/hop simply lose packets late.

In fact, there is a potential 3rd problem, with certain ATM topologies:

If the router attached to an ATM cloud is seen as an ATM source, but does not do traffic shaping (in practice, this is a common failing, even though against the standard), then it is quite possible that the bandwidth to the router outside the ATM cloud is higher than the ATM bandwidth. It is then quite impossible for the ATM cloud to use any buffering scheme whatsoever to stop loss at the first switch, since it is being provided with data faster than it can sink it.

In the long run, integrated services IP routers will need to know the underlying bandwidth available between each other to allocate resoruces properly: thus there must be an itneraction between any ATM mechanism that alters the available resources on an ATM "hop" and the IP entities either end of the hop.

5 Proposed Solution

We propose a possible solution to this problem. First divide paths across the ATM ABR service as forming two classes:

- An "I" type, where the ATM path is only part of the end-to-end path.
- A "A" type, where the path across the ATM Is the whole path.

The traffic patterns impinging on all "I" type VCs are monitored, and a single bit state variable kept to indicate whether a VC has cooperative or disruptive traffic. Cooperative traffic is indicated by a source responding to a change in network conditions.

The VC can have a number of parameters altered internally to provide the requisite implicit feedback to the source:

1. Packets can be dropped (strictly, cells can be dropped, but with the Sun proposed Early Packet Discard, this comes to the same thing).
2. The delay could increase
3. Both of the above.[1]

If the total ABR on an "I" type path changes, apportion the loss/throughput to those sources that cooperate in such a way as to maximise their chances of adapting without significant loss of throughput. In other words, given a mix of sources, both cooperative and disruptive, penalise the disruptive sources more

[1] Ideally, the traffic is marked with loss versus delay preference, via the CLP parameter for instance i nthe ATM path, but will need to pass this through the IP layer, cleanly.

- VBR traffic can be modelled as CBR + variable, and

Under congestion, best effort traffic will drive VBR down to the CBR and operate in its share.

In the long run, we need to look at the reactivity of VBR and see how it interacts with TCP schemes (and the same schemes used to adapt the variable part of video flows, c.f. INRIA/UCL IVS tool). This may particularly be true of multicast VBR video.

3 ATM and Available Bit Rate services

The service given by ATM nets to these protocols that do not fit the CBR or VBR model is called available bit rate (ABR). It is what is left after the predictive and guaranteed service traffic is served. In essence, it is simply a fair share of the remaining bandwidth amongst the VPs and VCs that have asked for this service.

A question about ABR that we may ask is how is fairness defined and implemented?

3.1 Fairness and Cooperation

If we view the ATM "cloud" as a huge multiprotocol router, with a number of flows through it, then we can picture fairness very simply, in the steady state. A simple token or rate scheme can allocate bandwidth evenly amongst all the VCs.

However, the "cloud" is subject to a number of transients:

CBR and VBR circuits come and go, leading to step functions and other changes in the total available bit rate service. Calls using ABR come and go too, leading to different changes in the bandwidth to be shared out. But most importantly, some end system protocol support bandwidth adaption. The foremost of these is the TCP Slow Start, Congestion Avoidance and Fast Retransmit algorithm. Another is that used by TP4 in DECNET, the so-called DEC Bit. Yet another is the scheme proposed by Postel et al, called Source Quench Introduced Delay. Cheriton also proposed monitoring the delay/throughput and loss/load curves to adapt the senders rate. Finally, even NFS sources adapt to some extent to perceived network conditions.

However, there remain other protocol suites that will continue to send at whatever rate they can, despite alterations in network conditions (typically these are systems developed in a LAN environment, without thought for WAN operation, where the bandwidth*delay product is very much greater, and the variation in these parameters also so much larger).

If we call the former list of protocols "cooperative", (or perfectly selfish), and the latter "disruptive" (imperfectly selfish), can we devise a scheme to permit the best of all possible worlds in the presence of transients?

4 Two or more Problems with Proposed ABR Service as part of an Internet path

There are two problems with the way in which the ABR service is envisaged, if it is to be part of an evolutionary path for sections of the Internet. They are based on two connected design decisions:

- It is proposed to be lossless on the ATM part of a path.
- It is proposed that adaptive buffering up to the bandwidth/delay product on each VC be assigned.

The problems for adaptive end system traffic that traverses a path which partially includes ABR are:

1. The introduction of new flows results in a step function increase in the delay *at the same time* as a step function decrease in the bandwidth. This results in a potential timeout at

An FTP can complete anytime before I want to print or store a file (usually), and has the capability (measured between two sparc 10/40s running IP over AAL5 on an ATM switch) at 90 Mbps), of going faster than most other applications (for instance 90 Mbps is a lot faster than the Cambridge pure ATM uncompressed video camera source, the AVA, and is 45 times as fast as SunVideo running MPEG)

1.2 Integrated Service IP, and Best Effort ATM

The CSZ[3] work is aimed at producing a scheme that predominantly enhances the Internet service to be a soft-connection. It still requires statements to the network of anticipated requirements, although it leaves scope for remaining throughput in an underloaded net to be used by old fashioned 'best effort'. ABR (Available Bit Rate) service being worked on by the ATM forum is the equivalent service for B-ISDN.

There are two current ABR proposals, one based on the Havard work by Kung[**?**], with credits and hop-by-hop flow control, and the other rate-based[5] (with an additive-increase, multiplicative-decrease feedback loop remarkably reminiscent of the TCP[1] congestion avoidance scheme). Both are workable, but achieve a lossless service at the expense of great delay variation.

2 Best Effort/Internet End to End Congestion Avoidance

A great deal of work has been done in worst case congestion avoidance by end systems given little or no network coupling for control loop algorithms (e.g. FIFO or Random Early Drop, or other gateways). This has resulted in Tahoe/Reno TCP[2] algorithms, DECBIT, Loss-Load Congestion Curve, and various 2nd order modifications (Slow Start and Search) and some suspect first order alternatives (e.g. rate based or throughput estimation based methods).

These all assume that loss or packet marking (DEC Bit) give delayed feedback that congestion is occurring. They also are predicated on everyone using them, to be fair and effective (well, fair to a first order, c.f. end-to-end flows sharing common hops, but with different bandwidth-delay paths overall).

2.1 Quarantined Traffic

It has been shown that analysis of a FIFO queuing system cannot be solved for fair and guarantees, but that a (hierarchical round robin, or layered fair queueing) Packet Generalised Processor can provide resource guarantees (bounded tail of delay distribution) while still providing statistical multiplexing, under broad, but not open assumptions about the arrival process: It is constrained by some leaky bucket or similar approach. [Actually, it is not surprising FIFO is hard to analyse - it is a very odd service - sorting different sources by an independent index, arrival time, has very little to do with any meaningful network service).

The noteworthy thing about this is

- The traffic is modelled usually as a mean, peak and burst
- Most analyses leave out loss and mean variation.

In fact, the kinds of traffic we need support include a broader range than can be considered in so simple a model:

- An FTP will vary its mean, can tolerate 100between 1 packet and more packets than fit in 1 rtt*bandwidth, without breaking.

  So what rule should be used by TCP for FTP traffic when sharing the network with PGPS, to attain maximum network power (or Best Effort traffic shares fairly with itself and with the variable portion of the VBR
- CBR traffic is simply subtracted from available transmission bandwidth.

Why Lossy Internetworking and Lossless ABR ATM Services Do Not Go Together - RN/94/21

Jon Crowcroft

June 12, 1994

Abstract

This paper is about the conflicts between two Multiservice Network architectures and to some extent, the Data Transfer Perspective for Congestion Control, in the context of an Internet with some (or most) paths provided by underlying ATM.

In the limit, we believe that networks will have excess capacity, and that this will be the model for all traffic control - hence traffic control is about finding the operating point for a fair share of the network.

The conflicts between the emerging multiservice model for the Internet, which could operate in this future, and that for ATM, which address a bandwidth limited scenario, arise because we foresee a period during which ATM available bitrate will be used as part of Internetwork paths, the rest of whose hops will either be best effort, or some new sub-service of integrated services internet, but that each is being designed with a view to being the all pervasive model, and not running recursively above the other.

1 Introduction

We envisage the Internet moving slowly towards a multiservice environment. We envisage the emergence of ATM provision, with CBR, VBR, VBR+ and ABR (Available Bit Rate) or similar, in *some* parts of the Internet.

However, the support for different traffic models is approached differently in these two endeavours, and it is clear that the mix is not a happy one.

In this paper, we look at just the problem for applications that expect best-effort service. In future work, we will extend this to examining the effects of mixing service support mechanisms with different architecture, on predictive and guaranteed service traffic.

1.1 Resource Guarantees in IP and ATM

Most of current work on multiservice nets is about resource guarantees achieved through connection setup and network support.

This paper is about the end-to-end control loop for data services with ***absolutely no*** guarantees at all (or on such a long timescale as to be outside of the scope of any packet policing algorithm), and how they interact with the aforesaid services.

We argue that most multimedia end systems are workstation based, running conventional (RTP/UDP/IP) protocol stacks and that the sources of traffic will be largely those developed to run on best-effort/ABR networks. Thus there is some importance in seeing how such traffic is affected by the introduction of more guarantees for some traffic on the net, while not for other. Ease of programming to a datagram service has proved most effective in enabling multimedia applications on the nascent Mbone. Ease of modelling with an enhanced integrated service internet is seen as easier than a de-enhanced VC service. This will lead to at least several years of coexistence of best effort, integrated services IP, and IP over ABR on ATM.