Image Browsing and Animation over a Local ATM Testbed*

George C. Polyzos

Computer Systems Laboratory
Department of Computer Science and Engineering
University of California, San Diego
La Jolla, CA 92093-0114, U.S.A.

Email: `polyzos@cs.ucsd.edu`, Tel.: `+1-619-534-3508`, FAX: `+1-619-534-7029`

Abstract

We argue for a time-driven approach for interactive multimedia communication applications. This is in sharp contrast to the traditional data-driven style almost universally adopted. The time-driven approach can take advantage of the multiple communication channels of various quality-of-service levels that will be supported by ATM networks. Such applications can rely on and take advantage of forms of hierarchical coding for handling real-time continuous media (e.g., video, image animations, audio, etc.) effectively and efficiently. In many such cases (e.g., image database browsing), timely presentation is more important than lossless presentation. The use of hierarchical coding by applications will allow the network to intelligently discard traffic of lesser importance to the applications at times of high congestion, and will thus allow networks to operate at higher degrees of statistical multiplexing without causing problems for the integrity of the most important signal components. We present some initial performance results of a simple hierarchically encoded image animation in a local ATM environment, including some results of the image quality when loss is experienced.

1 Introduction

We are investigating the use of hierarchical coding techniques for image and video communication over packet-switching networks. Using hierarchical coding techniques applications can take advantage of statistical multiplexing gains and thus enable networks to support the required quality of service, while achieving high utilization. Image database browsing can also take advantage of the reduced perceived latency of either progressive transmission or transmission of only the most important signal components.

Hierarchical coding techniques split signals into components of varying importance [4, 7]. The aggregation of these components reconstructs the original signal, but subsets of the data can also provide various degrees of approximation to the original signal. Hierarchical coding is important for the efficient use of high-speed packet-switching networks. The main issue for such network architectures is the significant congestion control problems that can arise due to the statistical multiplexing of many very high burstiness signals [1].

A key property of any congestion control approach not based on resource reservation at the peak rate of the sources is the ability to shed load quickly and without causing an avalanche of retransmissions of dropped traffic. With hierarchical coding of continuous media, when network congestion arises it is possible to drop the less important signal components without causing service interruption, and without the need for retransmissions. Since it is expected that continuous media will constitute the bulk of the network traffic in the future, this technique can be very effective as a last resort for congestion control.

Hierarchical coding can also play an important role at the receiver because it provides the system software

*This research is sponsored by Digital Equipment Corporation, under Digital's flagship research project Sequoia 2000, other Sequoia 2000 industrial and government partners, and the UC MICRO program.

with the capability of allocating resources based on local specifications and priorities. Finally, hierarchical coding can also be used to mask network latency by displaying the most significant components of an image first; this might be very useful for image browsing where a user typically searches through many images quickly before concentrating on one, which is inspected in detail.

Image browsing, animation of scientific visualization sequences, and video conferencing for scientific collaboration are important *Sequoia 2000* applications [3, 6] that generate large amounts of traffic and have stringent delay requirements.

2 Hierarchically Coded Continuous Media over ATM

The standard ATM cell format incorporates one bit indicating the cell loss priority (CLP).[1] Therefore, a two level hierarchical coding scheme can easily be implemented in the mainstream of the ATM specification. However, there are no direct provisions for a multilevel hierarchical coding and traffic priority scheme. Such a scheme might not be necessary for simple congestion control in a network of equal capacity links, however, it might be important for multicasting to destinations reachable through a very diverse capacity network [5]. For multiple levels of priority we are exploring the following alternatives(in combination with the CLP bit):

- Using some bits from the (3-bit) payload type (PT) field. In particular, there is one bit reserved for future definition. Note that ATM switches might use the PT field for differentiating and switching preferentially some payload types. However, we are interested in differentiating among components of the same medium, e.g., video. Therefore, to use this approach we need to define payload types based on the coding hierarchy. For example, we can have high and low resolution video payload types.
- Setting up multiple virtual circuits for each information stream, for example, using separate virtual circuit identifiers (VCIs) for the high and low resolution components of a video signal. Priorities should then be associated with the VCIs. In that case switches can control congestion by selectively forwarding cells from specific (high-priority) VCs. Note that this technique would significantly increase the workload of a traditional switch (because it would multiply the number of VCs that the switch would have to handle). However, this is not a significant problem in the ATM context because the splitting of the (generic) virtual circuit identifier into a virtual path identifier (VPI) and a VCI allows for routing/switching to be done at the VPI level, if desired. This method would support practically unlimited levels of hierarchical coding. It has also the advantage that receivers that cannot use the higher quality signal do not even have to receive it, receiving instead only the VCIs they can use.
- Using payload bits, invisible to standard switches, but usable by higher layer protocols and specialized switches, e.g., at points in the path where link capacity drops abruptly.

[1] The IP specification includes a 3-bit PRECEDENCE field in the TYPE-OF-SERVICE byte, which can be used to distinguish 8 priority levels. Even though most Internet hosts and routers currently ignore this field, it has been used in the past to provide priority for some interactive applications on the NSFNET backbone, and is again under consideration.

3 The Application

Traditional image and video communications systems are designed to be data-driven, i.e., the system displays a frame at the moment it has received all data pertaining to this frame. Images and video frames are typically segmented over multiple network packets even in non-ATM environments because of packet-size constraints. Delay and loss in the network cause jitter, i.e., result in irregular display rates. Reliable communication protocols are typically used and therefore loss results in further delay in order to successfully re-transmit the lost data.

We have developed a prototype interactive image browsing and animation system which uses elementary hierarchical coding techniques. The system developed is time-driven, i.e., it displays an image at the appropriate, predetermined point in time, irrespective of whether all related data have been received. Our assumption is that hierarchical coding and system design and tuning would assure that all data are present most of the time, and the most important data are present all of the time. This system illustrates the potential benefits of hierarchical encoding of images for browsing and animation across a network in the face of delay and loss in the subnet.

The hierarchical coding method selected for this experiment is one of the simplest and more basic techniques, known as bit-plane separation, where the i-th bits of the pixels in the frame are grouped together. Thus, the 0-th bit-plane is the collection of the most significant bit of each pixel in the frame. Each frame has n bit-planes where n is the number of bits per pixel. Each packet in the $0...x$ bit-planes is assigned high priority, and packets in the $(x+1)...n$ bit-planes are set to low priority (where x is a configurable parameter). Bit-planes are always sent from the most significant bit-plane to least significant bit-plane.

The animation consists of two processes. The first process is the encoder/sender process which sends the video images in packetized form over the network. The second process is the displaying process. It receives packets and places them into a buffer according to the appropriate bit-plane and offset in the plane. Upon expiration of an interval timer, the received data is sent to the frame buffer for display. Packets that arrive for a frame that has already been displayed are ignored.

In a related project a third process receives the packets sent from the encoder/sender process and simulates loss or delay in a subnet, discarding some low priority packets. High priority packets are never discarded. We decided to implement this process that simulates network congestion so that we can easily generate congestion phenomena, and thus investigate performance under various conditions and assumptions about the network/switching fabric.

3.1 The Animation Sequence

The experimental video animation is a sequence of 550×550 8-bit color images displaying the estimated net shortwave radiation over the Earth (produced using ISCCP C1 satellite data). This image sequence was generated by members of Catherine Gautier's lab at UCSB and is available (together with other images and data contributed by Sequoia 2000 researchers) on `heel.s2k.berkeley.edu` in the directory `/data/10/showcase`.

Each image (frame) is approximately 300 KB of data. Images are being sent at regular intervals to produce an animation effect. Each frame is fragmented into 264 packets, each containing 1224 bytes of data and application layer header and 32 bytes of UDP/IP header.

Figure 1 shows one of the images in the sequence. Figure 2 shows the same image coded using the bit-plane separation technique, with only the 5 most significant bit-planes. Figure 3 shows the effect of 50% random packet loss in the 3 least significant bit-planes. Color versions of these images are available by anonymous ftp from `cs.ucsd.edu` as `/pub/csl/Papers/cs93-346-pict.ps.Z`.

4 Experimental Environment

4.1 Systems and Networks

We experimented with a Fore Systems ATM switch and Fore Systems TCA-100 Turbochannel adapters. The switch ports and the adapters are based on the 100 Mb/s TAXI chip-set. The device driver supports the standard Berkeley sockets interface, but there is also an AAL interface, as well as direct access to the native ATM layer. The workstations used were two DECStation 5000/200 workstations, running Ultrix version 4.2A, communicating over the ATM LAN. The ATM switch was generally lightly loaded and experiments were performed during non-peak hours.

4.2 Animation and Throughput Experiments

Two experiments were performed. The primary experiment intended to determine maximum throughput and acceptable loss of the hierarchically coded image animation (referred to as animation). The secondary experiment was to determine maximum achievable throughput at the application layer for the animation experiment (referred to as throughput). The animation speeds were based on (and should be compared to) the throughput results. In both cases, throughput and loss statistics were collected. The acceptable loss value for the animation experiment was derived (in a non-scientific way) by observing the animation as it proceeded.

Both experiments consisted of a constant stream of packetized video data sent from one system and received and processed on the other system. The experiments were virtually identical and differed only in the following manner. In the animation experiment received packets are reconstructed into frames and displayed. In the throughput experiment, packets are counted and discarded, yielding a much higher possible throughput, which is considered the maximum possible for the animation.

Both experiments sent groups of packets over the network in a time-paced manner governed by the packet send time, in a manner similar to the leaky bucket algorithm. Packets were sent using UDP. No error or flow control were performed in order to maintain a strict open-loop control, as proposed in [5].

Experiments consisted of sending 100 or 200 frames, which is 32 or 64 megabytes of data, respectively. More specifically, each video frame was fragmented into 264 packets. The packets were then sent in small groups, periodically (based on an interval of time, t). The packet group size, g, was generally small, 2 or 3, and an integral divisor of 264. A larger packet group size was needed some cases due to the resolution of the interval timer available. The sender sent g packets and then waited for the timer to expire. When the timer expired, the timer was reset and the transmission of the next group of packets begun immediately. This guarantees that packets will never be sent faster than g packets per time t. However, it doesn't guarantee that packets are not sent slower than this rate. In the experiments, the group size and interval were adjusted to increase the rate (thus decreasing the frame time) to the point that packets started to be dropped. This rate would then be the limit of the network facility, the host network interface, or typically in this case, the application speed.

In this experiment, no explicit priorities were assigned, since the underlying networks do not have adequate support for priorities. However, there is an implicit priority due to the order of transmission. Bit-planes are sent in order from most significant to least significant, therefore, more significant bit-planes are more likely to arrive on time than less significant bit-planes. This didn't always prove to be a benefit. Note that images are sent uncompressed, although, this form of coding could yield very high compression ratios in the higher priority bit-planes.

5 Experimental Results

Our experiments led to the results shown in the following tables. Note that the maximum throughput rates reflect our specific non-optimized applications. The experiments were performed using solely application level processes and X-window display functions available to the user program. No kernel level optimizations were made or were required. As discussed earlier, the maximum throughput experiment was simply the animation experiment without the image processing; as such, it was intended to be an upper bound on performance of the animation experiment and not necessarily for the hardware or system itself.

ATM Maximum Throughput
(DECStation 5000/200)

Transmission Time (ms)			%	Packet Loss Per Frame	
Frame	Packet	(Mb/s)	Loss	max.	avg.
515.590	1.950	5.08	0.0	0	0.0
343.728	1.300	7.65	0.0	0	0.0
257.796	0.975	10.15	0.0	0	0.0
249.744	0.946	10.34	2.05	24	5.5
234.168	0.887	9.93	13.4	85	35.5

ATM Animation Statistics
(DECStation 5000/200)

Transmission Time (ms)			%	Packet Loss Per Frame	
Frame	Packet	(Kb/s)	Loss	max.	avg.
5191.070	19.529	511	0.0	0	0.0
4671.575	17.576	568	0.0	0	0.0
4152.077	15.623	638	0.2	2	0.4
3632.579	13.670	713	2.4	11	6.2

From observations of the animations, the tolerance to packet loss was determined to be in the 2-3% range (before it became obvious and annoying). Simulations of this animation yielded a higher tolerance of loss. Those simulations consisted of three systems, a sender, a receiver and a subnet simulator. The subnet simulator artificially injected loss into the stream by not forwarding packets to the receiver. Packet loss was simulated as random loss only in the low priority component of the video stream.

There main reason for the lower loss tolerance in this experiment is that the network did not adequately support priorities. Even though the switch has two internal queues and provides better service to applications that make Quality of Service demands (through the AAL), the adapters did not support priorities. Since there was little traffic at the switch, the priority mechanism was not really exercised. In contrast, the bottleneck was at the adapter side, and the lack of support for priorities there had a significant impact on performance. In this case the loss was not random at all, but occurred in bursts, just after the receiver's buffer had overflown, typically, in the same point in the packet stream as data was being buffered while a frame was being displayed.

6 Current and Future Work

We are deploying a version of this system over the Sequoia 2000 Network, a wide area network supporting both the TCP/UDP/IP protocol suite and the UCB/Tenet protocol suite, which provides performance guarantees [2]. This effort will provide us with a networking environment that supports priorities and performance guarantees and is also highly utilized, and therefore, the proposed techniques are highly relevant.

We are also pursuing emulation and simulation based studies of the performance of the animation under various assumptions about traffic and congestion in the subnet. Finally, as discussed in [5] these techniques are central in supporting multicast interactive continuous media applications. We are developing multipoint videoconferencing applications exploiting the multicast capabilities of the Fore ATM switch.

7 Conclusions

Hierarchical coding of continuous media can help reconcile the great economic advantages of statistical multiplexing in high-speed packet-switching networks with the complexity of controlling the congestion phenomena that it produces. It promises to provide the necessary freedom to design networks that provide the needed quality of service for applications, without reverting into tight control of all network resources.

In order to take full advantage of the semantics of image browsing, in addition to the necessary application modifications, it is important that the database and the storage system is organized in a way that individual media components be directly accessible. In order to reduce latency for image browsing across wide area packet-switching networks, the components of high visual importance should be transmitted first, or as high-priority traffic. Depending on the application and the performance and cost of communication, lower importance components could be transmitted later, or as lower priority traffic, or even not transmitted at all. Operating system and hardware support for video organized in layered formats is necessary to provide good presentation performance.

Interactive continuous media applications should be written using a time-driven rather than a data driven model. This will be a departure from the traditional approach and will require dealing with error concealment to address potential cell loss or excessive cell delays. This is exactly where layered coding techniques, and more generally decomposition of information flows into hierarchically related prioritized components, can provide the solution.

Acknowledgments

The author wishes to thank Catherine Gautier and her associates at UCSB for providing the animation sequence and Kirk Taylor of UCSD for performing the experiments.

References

[1] Special Issue on Congestion Control in ATM Networks, *IEEE Network*, Vol. 6(5) (September 1992).

[2] D. Ferrari and D. C. Verma, "A Scheme for Real-Time Channel Establishment in Wide-Area Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 8(3), pp. 368-379 (April 1990).

[3] D. Ferrari, J.C. Pasquale, and G.C. Polyzos, "Network Issues for Sequoia 2000," Proc. *IEEE Compcon Spring '92*, San Francisco, CA, February 24-28, 1992.

[4] G. Karlsson and M. Vetterli, "Packet Video and Its Integration into the Network Architecture," *IEEE Journal on Selected Areas in Communications*, Vol. 7(5), pp. 739-751 (June 1989).

[5] J.C. Pasquale, G.C. Polyzos, E.W. Anderson, and V.P. Kompella, "The Multimedia Multicast Channel," *Network and Operating System Support for Digital Audio and Video*, Lecture Notes in Computer Science #712, Springer-Verlag, pp. 197-208, 1992.

[6] G.C. Polyzos, "Networking for Sequoia 2000," Proc. *4th IEEE COMSOC International Workshop on Multimedia Communications (MULTIMEDIA '92)*, Monterey, CA, April 1-4, 1992.

[7] G.C. Polyzos, "Hierarchical Coding of Images and Continuous Media for Transmission over Packet-Switching Networks," Technical Report CS93-273, Dept. of Computer Science and Engineering, University of California, San Diego, La Jolla, CA, January 1993.

[8] K. Taylor and G.C. Polyzos, "Performance Measurements of a Simple Hierarchically Coded Image Animation over Various Network Testbeds," Technical Report CS93-346, Dept. of Computer Science and Engineering, University of California, San Diego, La Jolla, CA, December 1993. (Also available as Sequoia 2000 Technical Report 93/39, U. C. Berkeley.)

Figure 1: One of the 550×550 8-bit color images in the animation sequence. This sequence was produced from satellite data by members of Catherine Gautier's lab at UCSB (available through anonymous ftp from `heel.s2k.berkeley.edu` in the `/data/10/showcase` directory).

Figure 2: The same image using only the five most significant bit-planes. Color versions of these images are available by anonymous ftp from `cs.ucsd.edu` as `/pub/csl/Papers/cs93-346-pict.ps.Z`.

Figure 3: The same image with 50% random loss in the three least significant bit-planes.