COMPUTER NETWORKS and ISDN SYSTEMS

The International Journal of Computer and Telecommunications Networking

Theme Issue

ITC 14 Special Sessions Presentations

Guest Editor:

J.W. Roberts

ELSEVIER Amsterdam – Lausanne – New York – Oxford – Shannon – Tokyo

ELSEVIER

Computer Networks and ISDN Systems 28 (1996) 645–651

COMPUTER NETWORKS and ISDN SYSTEMS

Cost-quality tradeoffs in the Internet

Jean-Chrysostome Bolot *

INRIA, B.P. 93, 06902 Sophia-Antipolis Cedex, France

Abstract

The data delivery service currently offered in the Internet is a point-to-point best effort service. Work is underway in various IETF (Internet Engineering Task Force) working groups to include new services for applications that require a specific quality of service. Examples of such applications include the so-called real-time applications such as audio and videoconferencing, distributed interactive simulation, etc.

The cost in terms of network resources for providing these services is higher than that for providing best effort service. Thus, the existence of new services presents a tradeoff between quality and cost. Furthermore, recent work has shown that many real-time applications can adapt their output rates (and hence their network resource requirements) depending on network conditions (and hence on the available network service). Such rate adaptive applications present a further tradeoff between cost and quality of the data sent into the network.

In this paper, we describe these tradeoffs and analyze their impact on the applications and on the network. We illustrate our points with recent results and measurements obtained with IVS, which is a rate-adaptive audio and videoconference application for the Internet developed at INRIA.

Keywords: Internet; Adaptive applications; Videoconferencing; Integrated services internet

1. Introduction

The Internet uses datagram switching as a means of dynamically allocating network resources on a demand basis. Datagram switching provides flexible resource allocation, but it provides little control over the packet delay and loss processes at the switches. Specifically, when the scheduling discipline at the switches is the FIFO discipline, the departure times of packets from a particular connection depend very strongly on the order of arrival of the packets from other connections at the switch. This makes it essentially impossible to provide the guarantees, typically expressed in terms of minimum bandwidth or maximum delay, associated with real-time applications. Two approaches have emerged to tackle this problem.

One approach is to augment the current best effort service to include new services that provide the performance guarantees desired by the applications. Examples of such services identified within the IETF include the guaranteed service for intolerant applications (i.e. applications that require hard performance guarantees), and the predictive service for tolerant applications (i.e. applications that can live with fairly reliable guarantees). Another approach is to adapt the applications to the service provided by the network. An example of this is given by the so-called rate adaptive applications, which adjust their output rate based on network conditions. These applications trade off bandwidth (and hence cost in terms of network resources)

* E-mail: bolot@sophia.inria.fr.

SSDI 0169-7552(95)00072-0

for the quality of the data delivered to the destinations.

The cost of implementing the first approach is high, since it requires that the architecture of the Internet be changed, and that admission control, policing, reservation, pricing, and/or sophisticated scheduling mechanisms be implemented in the network. The second approach does not require special support from the network, and hence it can be implemented in the current Internet. However, it is not clear whether real-time applications in general can be made to adapt to network conditions, and whether adaptation only is enough to provide the desired performance levels.

We discuss these issues next. In Section 2, we describe our experience with adaptive applications in the current Internet. In Section 3, we describe how this experience can lead to understand the need for new services, and how these services could be offered in an integrated service Internet.

2. Supporting real-time applications in the current Internet

The current Internet offers a single class best effort service. From a connection's point of view, this amounts in practice to offering a channel with time-varying capacity. This capacity is not known in advance since it depends on the (a priori unknown) behavior of other connections throughout the network. To avoid rate mismatch (and hence network congestion), it is crucial that applications adapt to the capacity available in the network at any given time. One way to do this is via a control mechanism which adjusts the rate at which packets are sent over a connection based on feedback information about the network state.

Feedback control mechanisms are already used in the Internet to control sources of non real-time traffic. The best example is the window control mechanism of TCP. There, the feedback information is packet losses detected by timeouts or multiple acknowledgements at the source, and the control scheme is Jacobson's dynamic window scheme [12]. The idea of using similar control mechanisms for sources of real-time traffic is not new. Consider for example the case of video sources. Video has conventionally been transmitted over specific networks which provide connections with constant or nearly constant capacity channels (e.g. telephone or CATV networks). However, the rate of a video sequence can vary rapidly with time due to the effects of scene complexity and motion. The problem, therefore, is to obtain from the variable rate sequence a constant rate stream that can be sent into the network. This is typically done by sending the variable rate stream into a buffer which is drained at a constant rate. The amount of data in the buffer is used as a feedback information by a controller which adapts the output rate of the coder in order to prevent buffer overflow or underflow [4]. Feedback mechanisms for video sources have also been proposed for networks with variable capacity channels. There, the goal is to adjust the output rate of video coders based on feedback information about changing network conditions, i.e. changing capacity in the network [9,25].

For both fixed and variable capacity channels, the feedback control mechanism trades off image quality for network resource (specifically bandwidth) requirements, the goal being to maximize the perceptual quality of the image received at the destinations while minimizing the bandwidth used by the video transmission. We next illustrate this tradeoff with measurements and results obtained with IVS. IVS is a software videoconference system for the Internet developed at INRIA [23]. It includes a H.261 video codec [10] and a panoply of audio codecs. IVS data is sent over the Internet using IP multicast, UDP and RTP.

A central part of the video codec is the compression/coding algorithm. In H.261, a picture is divided into blocks of 8×8 pixels. A discrete cosine transform (DCT) converts the blocks of pixels into blocks of frequency coefficients. These coefficients are quantized and then encoded using a Huffman encoding technique. In addition, images can be coded using intraframe or interframe coding. The former encodes each picture in isolation. The latter encodes the difference between successive pictures.

It turns out to be surprisingly easy to change the output rate of the coder by adjusting parameters of the coder. In IVS, we adjust three such parameters, namely the refresh rate, the quantizer, and the movement detection threshold. The refresh rate characterizes the rate at which frames are grabbed from the camera. Decreasing the refresh rate decreases the average output rate of the coder. However, it also decreases the frame rate and hence the quality of the video delivered at the receivers. The quantizer characterizes the granularity used to encode the coefficients

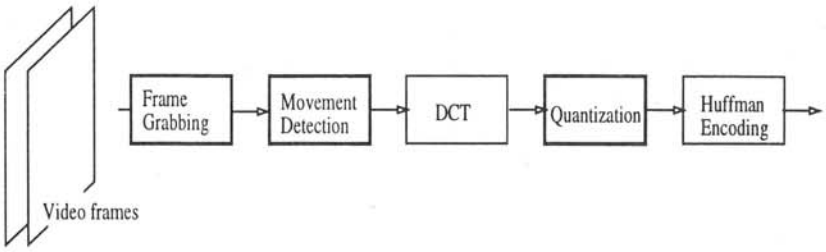

Fig. 1. Structure of the H.261 coder.

obtained from the discrete cosine transformation. Increasing the quantizer is equivalent to encoding the frequency coefficients more coarsely, and thus reducing the quality of the transmitted image. However, it is also equivalent to reducing the number of bits used to encode pixels, and thus reducing the output rate of the coder. The movement detection threshold characterizes the blocks in a frame which are "sufficiently different" from those in the previous frame. Increasing the threshold value decreases the number of blocks which are considered to have changed since the last frame. However, it also decreases the sensitivity of the coder to movement and yields lower image quality.

Thus adjusting the parameters of the video coder is an easy way, particularly in a software coder such as IVS, to trade off a lower output rate (i.e. lower bandwidth requirements) for a lower image quality. Which of the three parameters above is adjusted depends on the specific requirements of a video application. The refresh rate is adjusted if the precise rendition of individual images is important. The quantizer and the movement detection threshold are adjusted if the frame rate or the perception of movement is important. Other ways to control the output rate of video coders are described in [14,15,7].

Unfortunately, it is not so easy to modify the parameters of an audio coder to adjust its output rate. This is essentially because different compression schemes based on very different principles are used to obtain audio streams with different bandwidth requirements. One way around this problem is to use a panoply of audio codecs. IVS and other audioconference systems such as VAT [13] typically use PCM (at 64 kb/s), ADPCM (between 16 kb/s and 48 kb/s), GSM (at 11 kb/s), and LPC (below 5 kb/s) codecs. This makes it possible to choose the coding scheme most appropriate for the bandwidth available in the network at any given time.

At this point, we have shown that videoconference applications in general can adjust their bandwidth requirements. To use these applications over the Internet, however, we need a feedback mechanism to elicit information about the state of the network, and a control algorithm to adjust the audio or video output rate accordingly. The goal of the feedback mechanism is to estimate the state of the network, or rather its impact on the quality of the image received at the destinations. Since the number of destinations can be large and might even be unknown (recall that most audio and videoconference applications are expected to run in a multicast environment), the mechanism must scale well with the size of the multicast group. One such mechanism is described in [2]. Furthermore, we need to identify variables to evaluate the perceived quality of the images received at the destinations.

The Mean Opinion Score (MOS) has been used extensively as a subjective measure to design and compare video coding algorithms [11]. However, a MOS-based feedback mechanism would be impractical, since it would have to include the user in some kind of continual rating procedure. We thus have to rely on objective measures. Unfortunately, objective measures typically do not reflect the user's perception of an image [11]. The signal to noise ratio (SNR) is a measure of the spatial quality of the image. However, it is an imperfect measure because the perceptual quality in a sequence of frames depends on the quality of each frame in the sequence. Furthermore, it cannot be computed by the receivers since it requires that the original image be available. Another objective measure is the frame rate, i.e. the rate at which video frames arrive at the destinations. Yet another measure is the loss rate of the packets on the paths between the source and the destinations. We have chosen in IVS a feedback information based on measured packet loss rates at the destinations. Specifically, each receiver

measures an average packet loss rate observed during a given time interval. It then considers the network to be unloaded, reasonably loaded, or overloaded (i.e. congested) depending on the measured rate.

The control algorithm at the source gradually increases the audio or video bandwidth until the network is reasonably loaded. The source then transmits at this rate, continually polling its receivers to ensure that the network does not become congested. If the network is detected by one or more of the receivers to be congested, the source then reduces its output rate. Of course, a lower bandwidth at the source translates into a lower image quality for all destinations. However, it also translates into lower bandwidth requirements, and hence lower losses and delays in the network. There remains to quantify this bandwidth-gained/quality-lost tradeoff. The problem again is to find a way to estimate the quality of the video data delivered to the receivers. We argued earlier that the loss rate and the frame rate are good indications of this quality. Experiments carried out with colleagues at University College London (UCL) show that the control mechanism decreases the bandwidth requirements as well as the loss rate at the receiver, as long as the video traffic makes up a significant part of the total traffic on the path from INRIA to UCL.

3. Supporting real-time applications in an integrated services Internet

Our experience and that of others obtained with various audio and video transmission systems for the Internet such as VAT [13], NV [7], or NEVOT [20] indicates that the rate adjustment of audio and video coders makes it possible to maintain videoconferences of reasonable quality over the Internet. Of course, the audio and video signals (and therefore the user satisfaction) are degraded as the available capacity decreases, i.e. as the network load increases. It is not clear, however, whether this degradation is still tolerable when the available capacity is "very small". Consider for example the case of audio data. Known audio compression algorithms require a bandwidth equal to at least a few hundred bits per second [11]. There is clearly a problem if the available capacity in the network is lower than this minimum value. Ergonomic studies and anecdotal evidence from the Internet suggest (although this question would certainly benefit from further work) that users find audio and video data useful as long as the information content is above some minimum level which depends on the task at hand [26,1]. Therefore, there is a floor to the rate at which a real-time application can transmit and still send a useful stream of information. This presents the problem of who to satisfy when two applications compete for the same bandwidth, and whose combined minimum bandwidth requirements, i.e. combined floor rates, exceed the available bandwidth. The problem can be resolved either by turning off one of the applications (this is generally done by the end user who realizes that the network does not provide the desired service), or by preventing this situation from happening in the first place. This latter solution is typically implemented by means of admission control mechanisms.

However, it is important to note that the above problem is not likely to occur frequently (and hence admission control is not likely to be required or useful) if the floor rates are low and if the network is dimensioned appropriately. Unfortunately, it is not clear yet which fraction of applications will be rate adaptive, and what their floor rates will be. The answers to these questions impact the way the network architecture should be designed to provide the services required by the applications.

Scott Shenker has recently developed a simple model which brings out this impact [21]. Consider a network with a single bottleneck link with bandwidth b shared by N applications. For simplicity, all applications are assumed to be identical. The quality of an application as observed by a user of this application is captured by a function referred to as a utility function U which depends on the service provided by the network. In our case, we assume that the service is completely characterized by the available capacity. The utility for one application is $U(b/N)$, and the total network utility is $T(N) = N \times U(b/N)$. The question then is how to maximize $T(N)$. The answer to this question depends on the shape of the utility function U. Refer to Fig. 2. Shenker has shown that $T(N)$ increases monotonically if U is concave as in case (a). However, $T(N)$ first increases but then decreases as N exceeds some value N_0 if U is convex near the origin as in case (b). It is clear that in the latter case, the number of applications sharing the bandwidth

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.