Network Working Group   C. Partridge
Request for Comments: 1257
September 1991
C. Partridge
Swedish Institute of Computer Science

Isochronous Applications Do Not Require Jitter-Controlled Networks

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard. Distribution of this memo is unlimited.

Abstract

This memo argues that jitter control is not required for networks to support isochronous applications. A network providing bandwidth and bounds delay is sufficient. The implications for gigabit internetworking protocols are briefly considered.

Introduction

An oft-stated goal of many of the ongoing gigabit networking research projects is to make it possible to support high bandwidth isochronous applications. An isochronous application is an application which must generate or process regular amounts of data at fixed intervals. Examples of such applications include telephones, which send and receive voice samples at regular intervals, and fixed rate video-codecs, which generate data at regular intervals and which must receive data at regular intervals.

One of the properties of isochronous applications like voice and video data streams is that their users may be sensitive to the variation in interarrival times between data delivered to the final output device. This interarrival time is called "jitter" for very small variances (less than 10 Hz) and "wander" if it is somewhat larger (less than one day). For convenience, this memo will use the term jitter for both jitter and wander.

A couple of examples help illustrate the sensitivity of applications to jitter. Consider a user watching a video at her workstation. If the screen is not updated regularly every 30th of a second or faster, the user will notice a flickering in the image. Similarly, if voice samples are not delivered at regular intervals, voice output may sound distorted. Thus the user is sensitive to the interarrival time of data at the output device.

Observe that if two users are conferring with each other from their workstations, then beyond sensitivity to interarrival times, the users will also be sensitive to end-to-end delay. Consider the difference between conferencing over a satellite link and a terrestrial link. Furthermore, for the data to be able to arrive in time, there must be sufficient bandwidth. Bandwidth requirements are particularly important for video: HDTV, even after compression, currently requires bandwidth in excess of 100 Mbits/second.

Because multimedia applications are sensitive to jitter, bandwidth and delay, it has been suggested that the networks that carry multimedia traffic must be able to allocate and control jitter, bandwidth and delay [1,2].

This memo argues that a network which simply controls bandwidth and delay is sufficient to support networked multimedia applications. Jitter control is not required.

Isochrony without Jitter Control

The key argument of this memo is that an isochronous service can be provided by simply bounding the maximum delay through the network.

To prove this argument, consider the following scenario.

The network is able to bound the maximum transit delay on a channel between sender and receiver and at least the receiver knows what the bound is. (These assumptions come directly from our assertion that the network can bound delay). The term "channel" is used to mean some amount of bandwidth delivered over some path between sender and receiver.

Now imagine an operating system in which applications can be scheduled to be active at regular intervals. Further assume that the receiving application has buffer space equal to the channel bandwidth times the maximum interarrival variance. (Observe that the maximum interarrival variance is always known - in the worst case, the receiver can assume the maximum variance equals the maximum delay).

Now consider a situation in which the sender of the isochronous data timestamps each piece of data when it is generated, using a universal time source, and then sends the data to the receiver. The receiver reads a piece data in as soon as it is received and and places the timestamped data into its buffer space. The receiver processes each piece of data only at the time equal to the data's timestamp plus the maximum transit delay.

I argue that the receiver is processing data isochronously and thus we have shown that a network need not be isochronous to support isochronous applications.

A few issues have to be resolved to really make this proof stick.

The first issue is whether the operating system can be expected to schedule applications to be active at regular intervals. I will argue that whether or not the network is isochronous, the operating system must be able to schedule applications at regular intervals

Consider an isochronous network which delivers data with a tight bound on jitter. If the application on the receiving system does not wake up when new data arrives, but waits until its next turn in the processor, then the isochrony of the network service would be lost due to the vagaries of operating system scheduling. Thus, we may reasonably expect that the operating system provides some mechanism for waking up the application in response to a network interrupt for a particular packet. But if the operating system can wake up an application in response to an interrupt, it can just as easily wake the application in response to a clock interrupt at a particular time. Waking up to a clock interrupt provides the regular scheduling service we wanted.

Observe that the last paragraph suggests an application of the End-To-End Principle [3]. Given that the operating system must provide a mechanism sufficient for restoring isochrony, regardless of whether the network is isochronous, it seems unreasonable to require the network to redundantly provide the same service.

Another issue is the question of whether all receiving systems will have memory for buffering. For example, the telephone network is required to deliver its data isochronously because many telephones do not have memory. However, most receiving devices do have memory, and those devices, like telephones, that do not currently have memory seem likely to have memory in the future. Many telephones have a modest amount of memory now. Furthermore, even if the end nodes require isochronous traffic it is possible that last switch before delivery to the end node could provide the necessary buffer space to restore isochrony to the data flow.

Readers may wonder if the assumption of a universal time source is reasonable. The Network Time Protocol (NTP) has been widely tested on the Internet and is capable of distributing time accurately to the millisecond [4]. Its designer is currently contemplating the possibility of distributing time accurate to the microsecond.

Some Implications

The most important observation that can be made is that jitter control is not required for networks to be able to support isochronous applications. A corollary observation is that if we are to design an internetworking protocol for isochronous applications, that internetworking protocol should probably only offer control over delay and bandwidth. (There may exist networks that simply manage delay and bandwidth. We know that's sufficient for multimedia networking so our multimedia internetworking protocol should be capable of running over those networks. But if the multimedia internetworking protocol requires control over jitter too, then jitter control must be implemented on those subnetworks that don't have it. Implementing jitter control is clearly feasible - the method for restoring jitter in the last section could be used on a single network. But if we know jitter control isn't needed, why require networks to implement it?)

Note that the argument simply says that jitter control is not required to support isochronous applications. It may be the case that jitter control is useful for other reasons. For example, work at Berkeley suggests that jitter control makes it possible to reduce the amount of buffering required in intermediate network nodes [Y]. Thus, even if applications express their requirements only in terms of bandwidth and delay, a network may find it useful to try to limit jitter and thereby reduce the amount of memory required in each node.

Acknowledgements

Thanks to the members of the End-To-End Interest mailing list who provided a number of invaluable comments on this memo.

References

[1] Leiner, B., Editor, "Critical Issues in High Bandwidth Networking", Report to DARPA, August 1988.

[2] Ferrari, D., "Client Requirements for Real-Time Communication Services", IEEE Communications Magazine, November 1990. See also RFC 1193, November, 1990.

[3] Saltzer, J., Reed D., and D. Clark, "End-To-End Arguments in System Design", ACM Transactions on Computer Systems, Vol. 2, No. 4, November 1984.

[4] Mills, D., "Measured Performance of the Network Time Protocol in the Internet System", RFC 1128, UDEL, October 1989.

[5] Verma, D., Zhang H., and D. Ferrari. "Guaranteeing Delay Jitter Bounds in Packet Switching Networks", Proceedings of TriComm '91, Chapel Hill, North Carolina, April 1991.

Security Considertaions

Security issues are not discussed in this memo.

Author's Address

Craig Partridge
Swedish Institute of Computer Science
Box 1263
164 28 Kista
SWEDEN

Phone: +46 8 752 1524

EMail: craig@SICS.SE