Extensions to RTP to support Mobile Networking

Kevin Brown

Department of Computer Science
University of South Carolina
Columbia, SC 29208

Suresh Singh

Department of Computer Science
Department of South Carolina
Columbia, SC 29208

Abstract

In this paper, we identify limitations of the real-time protocol (RTP) regarding mobile networking and low-speed links and propose solutions to these problems. In particular, we propose schemes to limit the bandwidth used on the wireless link by RTP data messages and RTCP control messages.

1 Introduction

Developments in cellular communications and public desire for increasing personal mobility are combining to create a potentially huge market for *mobile computing* services. Mobile computing refers to an environment incorporating both wireless and high-speed networking technologies wherein users equipped with palmtops or laptops receive services over the wireless medium. The key issue we consider here, involved in providing services to the mobile user, is the limited wireless bandwidth.

Users accustomed to high-bandwidth fixed network connections expect high quality audio and video communication and other multimedia services. As they move to wireless devices, they will expect similar services even when connected via a low-bandwidth wireless connection. Hence, these services must scaleable over a wide range of bandwidths in order to support mobile users.

2 Real Time Protocol (RTP)

The Real Time Protocol (RTP) is a transport protocol used to support multimedia traffic on the existing internet. It does not require changes to existing routers or gateways and may be implemented on top of UDP/IP or ATM. As such, it may gain wide-spread acceptance and is worth further study. In RTP, each media source creates its own connection, and sends packetized data which is marked with a sequence number and timestamp. These fields allow receivers to recreate the media stream and to synchronize it with other streams from that same source (e.g. audio and video). Each media source also creates a Real Time Control Protocol (RTCP) connection on which it periodically sends *sender reports* (SR) which include the total number of packets and octets sent as well as more timestamps. *Receiver reports* (RR) are generated by the receiver to indicate their current loss ratios, jitter, and highest sequence number received. These reports allow the sender to detect congestion in the network and possibly alter its data rate to compensate.

Each sender and receiver report is sent to *every* participant of the RTP connection. However, the RTCP bandwidth is constrained to some percentage of the total session (data plus control) bandwidth; 5% is suggested in [1]. This control bandwidth is split up amongst all the participants and determines the rate at which they generate sender and receiver reports.

3 Mobile Networks

In order to provide high bandwidths to mobile users, geographic areas are split up into *cells*. The mobile hosts (MH) communicate over the wireless medium with a mobile support station (MSS) within the cell. In our proposed architecture [3], we add a third level to this hierarchy. A supervisor host (SH) controls several MSSs and provides connectivity to the fixed or wired network. The SH provides services to the MH including routing, flow control, and maintenance of quality of service (QOS) guarantees. The SH, along with the MSSs and MHs which it controls, make up a *mobile subnetwork*. We have proposed that protocols optimized for mobility be used on the mobile subnetwork to insulate the fixed network from the high error rates and frequent disconnections that characterize the wireless environment [4]. In particular, the SH maintains bandwidth allocation information about each of the MH's connections. Since the bandwidth available to mobile users can change quickly and drastically (from a 2 Mbps wireless LAN to a 19.2 Kbps CDPD connection), the presence of this information allows the SH to make intelligent and dynamic decisions regarding scheduling and congestion control within the mobile subnetwork.

4 Limitations of RTP

When we attempt to add mobile participants to a videoconference using RTP, several problems arise. If there are currently two members of the conference, both transmitting MPEG-1 video (1.2 Mbps) and CD quality audio (192 Kbps), then the session bandwidth is 2.784 Mbps, (we assume no more than 2 active senders, although there could be more). If a mobile host joins this conference, in order to receive the data in a timely manner, it would have to be the sole user of a high-end wireless LAN. A mobile user connected via CDPD would see enormous and unacceptable delays.

[1] suggests that a RTP *mixer* might be used to reduce the data bandwidth for slow connections. We propose the design for such a mixer below. However, reducing the data bandwidth is not enough. The bandwidth used for RTCP control messages is defined to be a percentage of the session bandwidth. RTP participants generate sender and receiver reports at this control bandwidth and send the reports to *all* other participants. Assuming a mixer recodes the MPEG video to teleconferencing quality (128 Kbps) and the audio to telephone quality (16 Kbps), this gives 288 Kbps used on the wireless link. But the sender and receiver reports will be arriving at 5% of 2.784 Mbps (the session bandwidth on the fixed network), giving 139 Kbps on the wireless link. This amounts to 32% of the total wireless bandwidth used just for control messages. Clearly, this is unacceptable. We propose mixer functions which scale down the control bandwidth on slow links but do not affect RTCP functions on the fixed network.

5 Data Bandwidth Reduction

RTP data bandwidth reduction will be accomplished by a mixer that is situated at the border of the fixed network and the mobile subnets, i.e the supervisor host in our architecture. The SH operates at the transport layer and hence may inspect RTP packets that are destined for MHs in its domain. We propose two data reduction techniques:

- Recoding data to a lower rate
- Intelligently discarding data

The SH will intercept RTP data packets on a per-connection basis, recreate the input stream, and apply a new coding technique which will produce an output bandwidth which is close to the current bandwidth allocated to the MH. For example, MPEG-1 (1.2 Mbps 30 fps) may be recoded to teleconferencing quality (128 Kbps 5-10 fps). Due to the difference in frame rate, the SH will serve as a synchronization source (SSRC) for this output stream, generating its own timestamps and sequence numbers. The bandwidth allocated to each MH in a SH's domain changes dynamically due to reception quality, and there may be many MHs in a SH's domain. As a result, the SH may choose to do a hierarchical recoding [6] which would allow it to do one recoding but to transmit output of varying quality and bandwidth requirements to different MHs.

Recoding alone will not be enough, however, because it must be done at discrete levels generating discrete output bandwidth requirements. The bandwidth available to a MH, however, ranges continuously over some interval. Also, the bandwidth can change very quickly, meaning that the recoding will not be able to provide the right bandwidth at all times. Hence recoding will be done to get a data bandwidth which is *close* to the bandwidth available to the MH. Then the SH will use a variation of the Loss Profile Transport Sub-Layer (LPTSL) [5] to judiciously discard data and produce an output data throughput which can feasibly be sent to the MH. Since random loss of data can cause complete failure in some application protocols, LPTSL throws out *logical* segments to ensure an x% reduction in bandwidth. For example, in MPEG-1, LPTSL might throw away the P or B frames and leave the I frames untouched.

LPTSL determines what constitutes a logical segment by inspecting the RTP packets. The packet type specifies the scheme used to packetize the data, and this may be used to find the boundaries of logical segments within the data stream. In the common case, one application data unit (ADU) corresponds to one RTP packet making discarding fast and simple. The MPEG-1 profile [2], however, allows a large ADU to span up to 3 RTP packets. The boundaries of the packet are marked in a standard way though, which will allow LPTSL to discard appropriately. LPTSL also supports different discard functions. To insulate the fixed network from the effects of mobility, we propose that the *MH* specify the discard function in a new RTP SDES (Source Description) item, PREFERENCES. For example, audio streams sound better under uniform loss whereas video applications would prefer to lose whole frames.

SDES items are periodically sent in RTCP packets by participants of the RTP connection. Other preferences a MH might specify in relation to a data stream are:

- Black and white video vs. color
- 16 colors vs. 256 colors
- audioconferencing speech quality vs. CD quality, or
- No data from this stream at all (refuse connection)

The SH would intercept this SDES item and attempt to provide the service requested by the MH within the limits of the supplied RTP data stream and the available wireless bandwidth. These preferences may be changed dynamically during the connection if the MH chooses (e.g. if it moves from a CDPD link to a wireless LAN).

Mobile devices will constitute a wide range of hardware platforms. As such, they may or may not have sound cards, hardware MPEG decoders, cameras, microphones, or even displays! To simplify the specification of preferences over many RTP connections, we propose another new SDES item, HARDWARE, whereby the MH informs the SH of the hardware support it has available. The SH maintains this state for each MH and sends RTP data streams which are constrained by the the mobile device's display media. In this way, the MH need only specify its PREFERENCES if the default based on its HARDWARE is unsatisfactory.

Hence, to reduce the data bandwidth, the SH recodes and intelligently discards parts of the input data stream based on the user's preferences, available hardware, and the available bandwidth on the wireless link.

6 Control Bandwidth Reduction

Once RTP data bandwidth on the wireless link has been reduced to some x bps, it is desirable to reduce the RTCP control bandwidth to a small percentage of the data bandwidth (e.g. 5%). Again the SH maintains bandwidth allocation information about each MH. It uses the standard RTCP algorithms to determine the frequency that sender and receiver reports should be delivered to the MH, but using the *reduced* data bandwidth as a parameter. The SH buffers the most recent report received from each other participant, discarding old ones, and sends the buffered report to the MH when the calculated time arrives. RTP sender and receiver reports are cumulative to guard against losses in the internet, so the discarded reports will not affect RTP functions at the MH. Note that most data inputs will be recoded by the SH, implying that the SH will generate its own sender reports for the MH rather than pass on sender reports from the fixed network. Clearly, the sequence numbers and octet counts in the original reports would have no meaning to the MH.

In this same way, receiver reports originating from the MH would have no meaning to the other participants. However, receiver reports must be received from all participants at the rate agreed upon or the participant will be marked inactive. Hence, the SH generates receiver reports for each MH in its domain based on the maximum sequence number received, loss ratio, etc. as perceived by the *SH*. These reports are generated at the fixed network rate. This allows the other RTP participants to do congestion control appropriately.

If a MH is the source of an RTP stream, the reduced control bandwidth may mean that it reacts slowly to congestion in the network signalled by the receiver reports. To alleviate this effect, we propose that some percentage of the control bandwidth (e.g. 10%) on the wireless link be reserved for special *overview* receiver reports. These reports would specify the worst and average case for each of the fields included in a standard receiver report, (cumulative packets lost, highest sequence number, jitter) and the number of senders and receivers. This information is available to the SH as it is already buffering the last report from each participant. These periodic overview reports would allow the MH to react quickly to network congestion but would not unduly interfere with the reception of the participant-specific information in the standard receiver reports. The overview report may be implemented as a profile specific header extension to a standard receiver report.

7 Future Work and Conclusions

The method for calculating the session bandwidth is not specified in [1], but the assumption appears to be that it is some bandwidth times the number of participants. To properly include mobile users, the specific bandwidth used by each participant should be part of the calculation. Since the bandwidth allocated to a mobile user changes dynamically, we propose that the maximum bandwidth a mobile user might receive be used in the calculation. In keeping with the design philosophy cited in [4], an explicit division of the RTP connection at the SH may be warranted. This split would allow an asymmetric protocol, which is optimized to support mobility, to be used on the mobile subnet. Lastly, although we have concentrated on mobile users, these solutions may be used for any RTP participant connected via a low-bandwidth link, given a mixer which performs the functions of the SH.

Acknowledgments

This work was supported by the NSF under grant number NCR-9410357.

References

[1] Internet Engineering Task Force, "RTP: A Transport Protocol for Real-Time Applications", *RFC 1889*, January 1996.

[2] Internet Engineering Task Force, "RTP Payload Format for MPEG1/MPEG2 Video", *Internet Draft draft-ietf-avt-mpeg-01.txt*, November 1995.

[3] Rohit Ghai and Suresh Singh, "An Architecture and Communication Protocol for Picocellular Networks", *IEEE Personal Communications Magazine*, Third Quarter 1994, pp. 36-46.

[4] Kevin Brown and Suresh Singh, "A Network Architecture for Mobile Computing", *IEEE Infocom 96*, S.F. CA, March 1996.

[5] Krishanu Seal and Suresh Singh, "Loss Profiles: A Quality of Service Measure in Mobile Computing", *Journal of Wireless Computing*, Vol. 2(1), 1996.

[6] Jose Moura et. al., "Video Over Wireless", *IEEE Personal Communications Magazine*, February 1996, pp. 44-54.