On the Scalability of RTCP-Based Network Tomography for IPTV Services

Ali C. Begen
Cisco Systems
San Jose, CA 95134 USA
abegen@cisco.com

Colin Perkins
University of Glasgow
Glasgow, G12 8QQ UK
csp@csperkins.org

Jörg Ott
Helsinki University of Technology
02150 Espoo, Finland
jo@netlab.tkk.fi

***Abstract*—Quality of experience (QoE) is an important, and admittedly overloaded, concept for the emerging IPTV services. Service providers are continuously working towards delivering a better TV experience. To this effect, cost-effective and scalable tools are highly desirable for QoE monitoring, diagnostics and reporting. In this paper, we demonstrate that the RTP Control Protocol and its network tomography extensions can satisfy the needs of the providers in collecting and reporting both detailed and summarized information using several numerical examples based on real-life scenarios.**

I. Introduction

IPTV is a collection of systems that deliver entertainment-quality video programs through an IP network that can run over, but are not limited to, a wireline, cable or wireless infrastructure. While the same programs can be delivered through the conventional radio frequencies, airways, cable or satellite distribution, IPTV offers more flexibility and richer features. Through IP, networked devices can connect to the IPTV distribution to retrieve available content whenever they demand, and integrate other IP-enabled services such as VoIP, chat and social networking with their TV services.

As much as offering more and richer features is important and attractive in converting existing cable and satellite TV subscribers, IPTV service providers may face additional hurdles in maintaining high customer satisfaction due to the uncertainties of IP networking. IP does not make any guarantees for timely delivery. Packet loss and jitter, and loss of connectivity due to malfunctioning, buggy or broken network hardware and software may occur anywhere in the network at any time. To keep the subscriber churn at a low rate, providers must deploy intelligent solutions (*e.g.*, [1, 2]) to minimize the occurrences of such events as well as their impact when they occur. When problems arise, they must be detected early enough so that they can be fixed quickly without causing long service interruptions. The detection as well as isolation of the faults in the network require the providers continuously monitor the packet transport and the quality of experience (QoE) of their subscribers. Monitoring is also needed to ensure that the action(s) taken against a problem indeed worked and resolved the outstanding issue(s).

A major role in this effort is to deploy cost-effective tools for efficient QoE monitoring, diagnostics and reporting. With the proliferation of Real-time Transport Protocol (RTP)-based IPTV architectures, a straightforward choice for this purpose is to use the RTP Control Protocol (RTCP) [3]. In this paper, we show that RTCP is suitable for use as a scalable, low-overhead, reporting channel for IPTV systems. We demonstrate that RTCP, with network tomography extensions, can provide a wealth of data for QoE monitoring, network management and fault diagnosis purposes. Our contributions are (i) the development of an architecture for RTCP monitoring in large multicast IPTV systems with hierarchical feedback, and (ii) demonstration that the monitoring overheads in terms of network capacity are small, and the architecture is scalable.

To the best of our knowledge, ours is the first analysis of the overheads of RTCP in a realistic IPTV deployment scenario. Previous work has shown that RTCP is scalable to large any-source multicast groups, and when used in source-specific multicast groups using hierarchical unicast feedback [4]. This latter scenario is similar to IPTV deployments, but reports on a multi-level hierarchical aggregation scheme, rather than the single-level hierarchy. It also neglects the costs of RTCP use for error reporting and network tomography, and the complexity introduced by asymmetric bandwidth in residential access links. By way of contrast, [5] considers scaling RTCP extended network tomography reports, but is focussed on VoIP sessions, and does not consider single-level hierarchical feedback as used in IPTV systems. We consider these points in our study.

In the remainder of this paper, we give an overview of the IPTV deployment models, RTP transport and RTCP-based monitoring architectures. We then derive expressions that relate to these architectures and present numerical examples.

II. QoE and Its Monitoring for IPTV Systems

In IPTV, measuring the network and transport-level metrics is essential, however, it is not solely sufficient in quantifying subscribers' QoE, since the whole experience is dependent on several factors including picture quality, audio clarity, usability and friendliness of the user interface, and responsiveness and interactivity of the auxiliary services. While RTCP has proven itself to be a good solution for conveying QoS metrics for VoIP and conferencing applications, and we demonstrate its suitability for IPTV in this paper, research is still underway to converge on universally accepted metrics and models that accurately quantify QoE.

A. Deployment Models

For linear content (*e.g.*, broadcast channels), IPTV uses IP multicast for efficient distribution to a very large number of IP set-top boxes (STB). Multicast IPTV cannot use TCP and almost always runs over UDP. For on-demand content, a unicast session is established between the IP STB and a content server often over UDP as well. For both service models, reliability must be provided by higher-layer protocols.

In Fig. 1, an example IPTV distribution topology is presented where content is injected from content servers at one edge, transported over several transit and distributor networks and finally consumed by receivers at other edges. Multiple content owners, transit and service providers may co-exist, and serve IPTV subscribers in different locations.

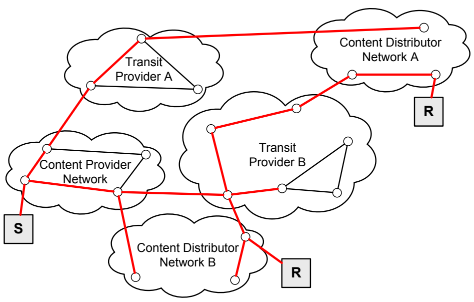

Fig. 1. Transit and distributor networks connecting the content servers and receivers.

B. RTP Transport and RTCP Feedback

The most widely available transport format for digital broadcast and on-demand media content has been the MPEG2 Transport Stream (MPEG2-TS) [6]. MPEG2-TS is an encapsulation method that can carry content produced by both the conventional MPEG2 and emerging MPEG4/AVC (also called H.264) codecs. A single MPEG2-TS can multiplex digital audio, video and metadata, and provides an internal synchronization mechanism. In the IPTV world, MPEG2-TS has also been quite popular so far and each transport stream is often directly mapped onto a sequence of UDP datagrams. Newer standard-compliant deployments use RTP encapsulation for these UDP datagrams as specified in [7].

The RTP Framework is an Internet Standard, published by the IETF in July 2003 [3]. RTP comprises a data transfer protocol and an associated control protocol (RTCP), augmented by various extensions, profiles and media payload formats. RTP can run over any transport protocol, but it typically runs on top of UDP and benefits from UDP's checksum support to identify the corrupted packets. RTP provides the following main services: (i) *payload type identification* that allows applications to identify the format of the payload carried in the RTP stream, (ii) *sequence numbering* that allows RTP receivers to detect missing packets and put the packets in the right order, and (iii) *timestamping* that allows RTP senders and receivers to synchronize their clocks and calculate delay jitter.

The main purpose of RTCP is to provide RTP applications a minimal control and identification functionality as well as a scalable quality monitoring service for RTP transport. RTCP achieves the latter via sender reports (SR), receiver reports (RR) and extended reports (XR) [8]. Sender and receiver reports convey the basic transmission and reception statistics from the active sender(s) and receiver(s) in an RTP session, respectively. Extended reports provide supplementary information for the applications that desire more detailed statistics such as loss and duplicate run-length encoded (RLE) reports, packet-receipt and round-trip time reports as well as summary statistics reports. These reports collectively enable network monitoring and fault isolation using network tomography methods in multicast applications. Further extensions are currently being developed to convey various application-level metrics.

The RTCP reporting interval is computed in a way that the control traffic is limited to typically 5% of the media bandwidth, resulting in reporting intervals from several seconds to several minutes depending on the group size. With $\pm 50\%$ randomization, this approach has been shown to scale to sessions with tens of thousands of participants, and to cope well with flash crowds [9]. However, there has been no analysis of how its use for network tomography scales in single-level hierarchical IPTV deployments, as the sort we discuss below.

III. An RTCP-Based Monitoring Architecture

The source-specific multicast (SSM) perfectly fits the one-to-many nature of IPTV distribution for broadcast channels, where each channel is offered in an SSM session. Content servers (S) send the content to distribution sources (D), which act as SSM senders, generating RTP and RTCP traffic for each channel. Content is distributed to downstream receivers (R) and other network elements such as local caches and repair servers using SSM. Feedback from receivers is collected at feedback targets (FT) located at the network edges [10]. A simplified distribution architecture is sketched in Fig. 2.

Network QoS mechanisms can provide reliable, low-latency, delivery service to network elements within the core and aggregation networks. This is important, because packet loss that occurs early in the multicast distribution tree will, if not repaired, be seen by all receivers in that SSM session. On the other hand, it is often a challenge to achieve the same reliability for receivers beyond the access networks, largely because home networks are managed by unqualified end-users. Incorrect network configurations, lack of carrier-grade software and hardware in home-networking devices as well as poor wiring and incompatibilities lead to serious complications that adversely impact the QoE. Yet, a packet loss on a poor access link or home gateway device (H) is only observed by the IP STBs that are connected to this link or the home-gateway device, confining the impact to a small area. Network abnormalities occur with different probabilities, and impact different areas, depending on their location.

Referring to Fig. 2, the receivers and other network elements that join the SSM session to receive the RTP stream(s) provide RTCP feedback on their reception. Comparing reports

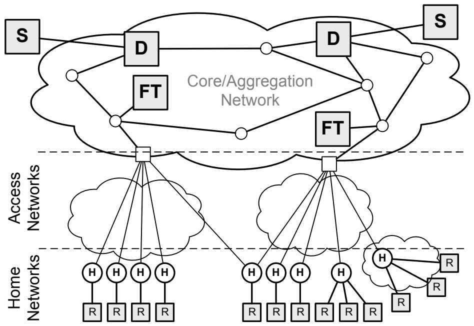

Fig. 2. Intradomain IPTV distribution and monitoring architecture.

coming from multiple nodes and paths, current conditions in a specific subnet or on a particular path segment can be determined, a process known as *network tomography*. Having more RTCP-capable network elements along the distribution tree(s) naturally provides more detailed information. The combination of the RTP and RTCP protocols, and their extensions for unicast feedback in SSM sessions, provides a powerful architecture for robust IPTV delivery.

IV. Analysis of Overheads and Scalability

Considering the intradomain IPTV distribution and monitoring architecture shown in Fig. 2, we identify four classes of RTCP monitoring and reporting flows:

- forward control information from the sender to receivers (S $\rightarrow$ R) for lip-synchronization, to indicate liveness, and to identify the sender;
- forward control information from the feedback targets to their receivers (FT $\rightarrow$ R) to manage feedback rates;
- reception quality reports from feedback targets to the sender (FT $\rightarrow$ S), reporting quality of the path from sender to feedback target, and summarizing reception quality for downstream receivers; and
- reception quality reports from the receivers to the feedback target (R $\rightarrow$ FT).

We observe that two RTCP feedback loops are operating: (i) between the sender and the feedback targets, and (ii) between the feedback targets and their downstream receivers.

A. The RTCP Feedback Loop FT $\leftrightarrow$ R

The main purpose of the RTCP feedback loop between the feedback targets and their downstream receivers is to provide regular reports on QoS and QoE. Feedback must be sent rapidly enough to provide the possibility of adapting the transmission to match the characteristics of the path, yet not so rapidly as to congest the subscriber upstream access link or overload the feedback target (which may be receiving feedback from many receivers, potentially watching different channels).

Fig. 3 shows a typical compound RTCP packet containing RR, source description (SDES) block and XR network tomography extensions to report detailed packet loss and reception time statistics, as might be sent from a receiver to feedback target in an IPTV system. The size of the RTCP packet depends on the number of RTP data packets lost (the loss RLE block) and received (the receipt times block) during

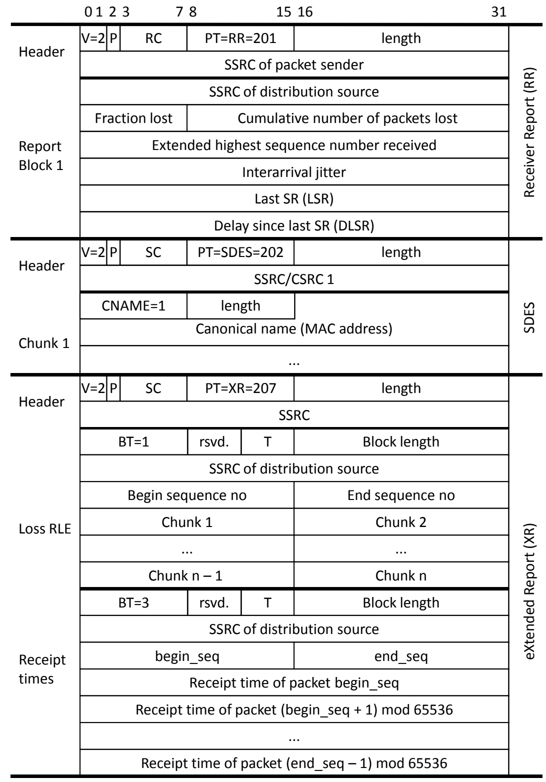

Fig. 3. RTCP reception report with network tomography metrics.

the reporting interval [5]. If the receiver uses the MAC address of the interface where it receives RTP traffic as its canonical name, the size of this compound RTCP packet in octets, $S_{RTCP,R}$, can be calculated as:

$$S_{RTCP,R} = 68 + S_{LossRLE} + S_{Receipt}. \tag{1}$$

The size of the receipt times RTCP XR block in octets is:

$$S_{Receipt} = 12 + \frac{4D_i}{2^{T_r}}, \tag{2}$$

where D_i is the number of data packets received during the RTCP reporting interval (for a media rate of P_i packets per second, the number of RTP data packets received during the reporting interval, I_r, is $D_i = P_i \times I_r$), and T_r is the thinning factor for receipt time reports (the receiver reports on the reception of only one in every 2^{T_r} packets).

The size of the loss RLE block is determined by both the number of packets lost and the loss pattern. In the best case, there is no loss. Thus, the loss RLE block can be omitted. In the worst case, packets are lost unpredictably, requiring a complete loss bitmap to be sent covering every packet. Finally, an important intermediate case is that a single loss event occurs between two good runs. These three possibilities give the following loss RLE block sizes:

$$S_{LossRLE} = \begin{cases} 12 + \frac{D_i}{15 \times 2^{T_l}}, & \text{worst case;} \\ 12 + 4, & \text{single loss;} \\ 0, & \text{no loss,} \end{cases} \tag{3}$$

where T_l is the thinning factor for loss RLE reports.

The maximum size of a compound RTCP packet is limited by the path MTU. Assuming report thinning is not used, and combining (1)–(3), the number of events that may be reported is constrained as follows:

$$\text{MTU} \geq \begin{cases} 80 + 4D_i + 12 + \frac{D_i}{15}, & \text{worst case;} \\ 80 + 4D_i + 16, & \text{single loss;} \\ 80 + 4D_i, & \text{no loss.} \end{cases} \quad (4)$$

With the common 1500-octet path MTU, this allows for a maximum of 355 reports to be included per compound RTCP packet in the no-loss case, reducing to 346 reports if the worst-case loss pattern occurs. For a typical 4.2 Mbps MPEG-2 IPTV channel (not counting the IP and UDP headers), sending $P_i = 400$ packets per second with payload size $S_{RTP} = 1316$ octets, this requires RTCP reports to be sent more than once per second per receiver if they are to report on all packet loss and reception timing events. This is clearly infeasible, since the overhead would be large and grow without bound as the receiver population increases, so we turn instead to the question of whether a meaningful subset of such metrics can be reported while retaining a reasonable RTCP reporting overhead, constraining $S_{RTCP,R} < S_{RTP}$ and assuming the RTCP bandwidth allocation is the standard fraction, f = 5%, of the RTP media bandwidth.

The RTCP reporting interval then depends on the average RTCP packet size, the media bandwidth, M, and the number of receivers. The media sender will generate RTCP reports comprising an SR and an SDES block, of size $S_{RTCP,S} =$ 56 octets (assuming the sender uses its MAC address as the canonical name). The size of a downstream receiver summary information (RSI) report sent as a non-compound RTCP packet [11] from the feedback target to its receivers is $S_{RTCP,FT} = 28$ octets when it contains a single RTCP bandwidth indication sub-block. For a population of N receivers, one sender, and one feedback target, the average RTCP packet size, S_{RTCP}, is therefore

$$S_{RTCP} = \frac{S_{RTCP,S} + S_{RTCP,FT} + N \times S_{RTCP,R}}{N + 2}. \quad (5)$$

The RTCP reporting interval for receivers, which receive 75% of the RTCP bandwidth allocation is (See Section 6.3 of [3])

$$I_r = \frac{N \times S_{RTCP}}{0.75 \times f \times M}. \quad (6)$$

Fig. 4 shows the increase in the RTCP reporting interval with increasing group sizes, for various values of S_{RTCP}, for a 4.2 Mbps media bandwidth. Once the 5-second minimum is exceeded, the reporting interval grows linearly as the group size increases, with the rate of increase depending linearly on the receiver RTCP packet size.

The receiver RTCP packet size constrains the amount of network tomography data that may be included in each packet. Assuming for the moment that $S_{LossRLE} = 0$, we may derive an expression for the thinning factor for receipt time reports, T_r, such that the resulting RTCP XR block fits within the available RTCP packet size. The expression for a single loss event is similar, except that the constant 80 in the denominator changes to 96 to account for the size of the Loss RLE block.

$$T_r = \left\lceil \log_2 \frac{4 \times P_i \times I_r}{S_{RTCP,R} - 80} \right\rceil \quad (7)$$

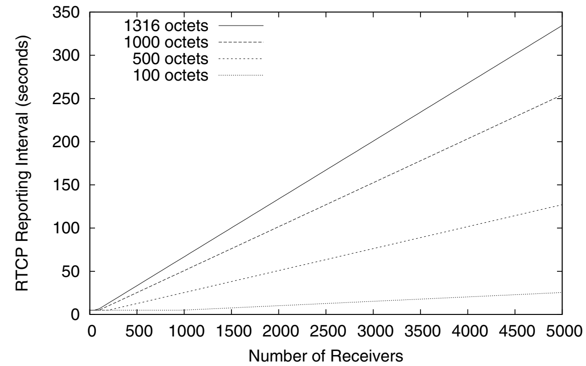

Fig. 4. Increase in RTCP reporting interval with increasing packet size (from 100 to 1316 octets) and number of receivers. Media bandwidth is 4.2 Mbps.

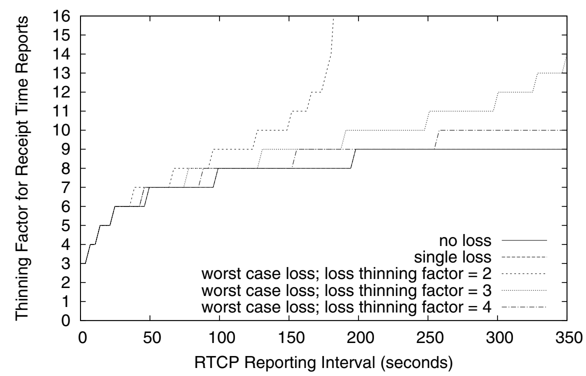

Fig. 5. Required T_r with increasing group size for the FT $\leftrightarrow$ R RTCP session. RTCP packet size is 1316 octets.

This is illustrated in Fig. 5, which shows that the required thinning factor rapidly increases, reaching 7 – report on one in every 2^7 packets – once the reporting interval reaches 50 seconds (a group size of around 500, in this example). This is considerably more detail than provided by the single interarrival jitter value included in the RR block, but still provides only a limited view of the timing characteristics.

Given that packet loss is rare in a well-managed network, the inclusion of a single-loss report will likely be sufficient to report on loss as well as timing variation for typical deployments. For generality, however, we consider worst-case loss patterns, where a report must be sent for every packet. Adding worst case $S_{LossRLE}$ to (7), we get a required thinning factor for receipt time reports of:

$$T_r = \left\lceil \log_2 \frac{4 \times P_i \times I_r}{S_{RTCP,R} - 92 - \frac{P_i \times I_r}{15 \times 2^{T_l}}} \right\rceil \quad (8)$$

With loss thinning factor, $T_l = 2$ (*i.e.*, report on every 4^{th} loss), we observe in Fig. 5 that reports can be sent with a thinning factor $T_r < 10$ (*i.e.*, report timing of every 1024^{th} packet) at RTCP reporting intervals of up to 150 seconds (corresponding to a group size of approximately 2000 receivers). Reporting on fewer loss events (increasing T_r) reduces the RTCP XR packet size, allowing more frequent reports to be sent. Alternatively, since the size of the XR block generated per unit time is smaller, this allows a receiver to continue reporting on packet receipt times at much larger

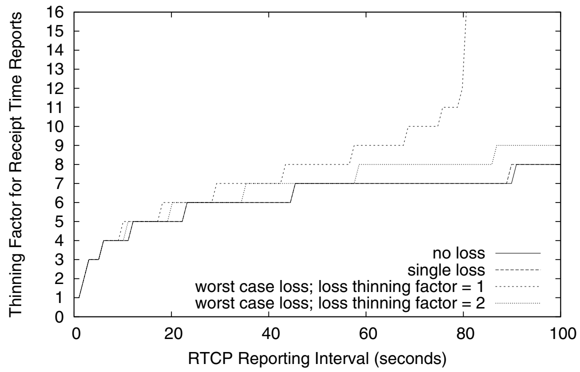

Fig. 6. Required T_r with increasing group size for the S $\leftrightarrow$ FT RTCP session. RTCP packet size is 1316 octets.

reporting intervals while remaining within the packet size constraint (although the usefulness of such sparse data for network tomography rapidly degrades).

B. The RTCP Feedback Loop S $\leftrightarrow$ FT

The second RTCP feedback loop is between the media sender and the feedback targets. Each feedback target will generate compound RTCP packets containing RR and SDES blocks (32 and 28 octets, respectively), an RTCP XR network tomography packet for the flow it receives from the sender, and an RSI packet sent on behalf of its downstream receivers. It will also analyze the network tomography data it gets from the receivers, and potentially raise management alerts if problems are detected. Detailed network tomography reports from all receivers are not forwarded to the media sender, since they would rapidly overload the system, and because of potential privacy concerns in allowing the sender to identify viewers.

The RSI packet sent by the feedback target can contain various statistics on loss and jitter as seen by the receivers. We suggest that appropriate contents might be a loss distribution block, reporting on the distribution of packet loss rates up to 25%[1], and a general statistics block reporting on the median interarrival jitter seen at the receivers; this will comprise 108 octets (more detailed jitter statistics can be reported, but will increase the size of the report). The remainder of the compound RTCP packet, constrained to $R_{RTCP} < S_{RTP}$, is filled with RTCP XR data.

For the 4.2 Mbps, 1318-octet media payload example we consider, this leaves 1142 octets for the RTCP XR data (after accounting for the XR header). Repeating the analysis in (7) and (8), and accounting for the size of the RSI block, we derive report thinning parameters as shown in Fig. 6. Reporting intervals up to around 60 seconds allow worst-case loss reporting with $T_l = 1$ (every second packet) and $T_r = 8$ (every 256^{th} packet); sufficient for detailed performance analysis [12] of the backbone. The reporting interval from the feedback target towards the sender follows (6) since the feedback targets take the entire receiver bandwidth fraction for their reports. From Fig. 4, it can be seen that a 60-second reporting interval corresponds to a feedback target population of around 500 (around 250,000 receivers if each reports in the same interval; up to one million if receivers report every 150 seconds).

[1]A count of receivers seeing a loss rate above 25% is included in the highest bucket; there seems little value in reporting details of higher loss rates, since a loss rate of 25% already indicates a serious problem in the system.

V. Conclusions

We have demonstrated that RTCP can provide a timely, low-overhead, channel for IPTV QoS reporting, including detailed reports for network tomography. For a typical 4.2 Mbps IPTV flow, this allows a hierarchy comprising approximately 500 feedback targets to provide detailed QoS reports on the critical flows in the core network once per minute, each summarizing reception quality reports from around 2000 receivers reporting every 150 seconds, for a total scale of approximately one million receivers, with only 5% reporting overhead. This provides a powerful tool for network management and ensuring subscriber QoE. Introducing multi-level feedback targets could even allow further scaling and extend the tuning range for the reporting frequency and details, thus giving service providers more flexibility without increasing overhead.

This was a preliminary theoretical analysis. In our future work we plan to confirm these results with simulations and experimentation with inputs from actual deployments.

Acknowledgments: Thanks to Martin Ellis for his feedback on an early version of this paper.

References

[1] C. Filsfils and J. Evans, "Deploying diffserv in backbone networks for tight SLA control," *IEEE Internet Comput.*, vol. 9, no. 1, pp. 66–74, Jan. 2005.

[2] R. Doverspike, G. Li, K. N. Oikonomou, K. Ramakrishnan, R. K. Sinha, D. Wang, and C. Chase, "Designing a reliable IPTV network," *IEEE Internet Comput.*, vol. 13, no. 3, pp. 15–22, May 2009.

[3] RFC 3550, "RTP: A transport protocol for real-time applications." [Online]. Available: http://www.ietf.org/rfc/rfc3550.txt

[4] V. Novotný and D. Komosný, "Large-scale RTCP feedback optimization," *Jour. of Networks*, vol. 3, no. 3, pp. 1–10, Mar. 2008.

[5] R. Cáceres, N. Duffield, and T. Friedman, "Impromptu measurement infrastructures using RTP," in *IEEE Int. Conf. Computer Communications (INFOCOM)*, 2002.

[6] "Generic coding of moving pictures and associated audio information: Systems," *ITU-T H.222.0*, 2006.

[7] RFC 2250, "RTP payload format for MPEG1/MPEG2 video." [Online]. Available: http://www.ietf.org/rfc/rfc2250.txt

[8] RFC 3611, "RTP control protocol extended reports (RTCP XR)." [Online]. Available: http://www.ietf.org/rfc/rfc3611.txt

[9] J. Rosenberg and H. Schulzrinne, "Timer reconsideration for enhanced RTP scalability," in *IEEE Int. Conf. Computer Communications (INFOCOM)*, 1998.

[10] Internet Draft: draft-ietf-avt-rtcpssm-18, "RTCP extensions for single-source multicast sessions with unicast feedback." [Online]. Available: http://tools.ietf.org/html/draft-ietf-avt-rtcpssm

[11] RFC 5506, "Support for reduced-size real-time transport control protocol (RTCP): Opportunities and consequences." [Online]. Available: http://www.ietf.org/rfc/rfc5506.txt

[12] N. Duffield, J. Horowitz, D. Towsley, and W. Wei, "Multicast-based loss inference with missing data," *IEEE J. Select. Areas Commun.*, vol. 20, no. 4, pp. 700–713, May 2002.