A Receiver-driven Adaptive Mechanism Based on the Popularity of Scalable Sessions

Paulo Mendes[1,2], Henning Schulzrinne[1], and Edmundo Monteiro[2]

[1] Department of Computer Science, Columbia University
New York NY 10027-7003. Phone: +1 212 939-7000. Fax: +1 212 666-0140
{mendes,schulzrinne}@cs.columbia.edu
[2] CISUC, Department of Informatics Engineering, University of Coimbra
3030 Coimbra, Portugal,
{pmendes,edmundo}@dei.uc.pt

Abstract. Receiver-driven adaptation allows streaming of multimedia content to different receivers across heterogeneous networks. However, receivers are only encouraged to adapt if network providers guarantee a fair distribution of bandwidth and also the punishment of receivers that do not adjust their rate in case of congestion. We define a receiver-driven adaptive mechanism based on a new fairness protocol that provides the required guarantees for adaptation. We use simulations to evaluate the proposed mechanism and to compare its performance with other receiver-driven mechanisms.[3]

Keywords: quality adaptation, scalable sessions, SAPRA, fairness, TCP

1 Introduction

Source-based rate adaptation performs poorly in a heterogeneous multicast environment because there is no single target rate: the different bandwidth requirements of receivers cannot be simultaneously satisfied with one transmission rate. This problem can be solved if sources use scalable encoding and the adaptation task is performed by receivers. Scalable encoding [21,7] divides a video stream into cumulative layers with different rates and importance. Layers are then sent to different multicast groups. All layers belonging to the same stream form a *session*, as in the Real-Time Protocol (RTP) [20]. The rate of each session is obtained by adding the rates of all its layers.

When a receiver-driven approach is combined with scalable encoding, receivers can adapt to the best quality the network offers, by selecting subsets of layers for their session. But, to motivate receivers to adapt, the network has to have three fairness properties. The first is *inter-session* fairness, the ability to guarantee a fair distribution of bandwidth between sessions sharing a service. The second is *intra-session* fairness, the ability to respect the importance of

[3] This work is supported by POSI-Programa Operacional Sociedade de Informação of Portuguese Fundação para a Ciência e Tecnologia and European Union FEDER

each layer of a session. The third is the ability to punish *high-rate* sessions, i.e., sessions with a rate higher than their fair share of bandwidth, due to the fact that their receivers do not reduce the reception rate when packets are lost.

The current Differentiated Services (DS) model [2] aggregates traffic into services with different priorities at the boundaries of each network domain. Among the services DS can provide, the Assured Forwarding PHB (AF) [4] is ideal for transporting scalable sessions, since flows are assigned different drop precedences. Although AF services provide intra-session fairness, the DS model lacks the other two properties. Therefore, we proposed [14,15] a protocol named *Session-Aware Popularity-based Resource Allocation* (SAPRA) that allows a fair allocation of resources in each DS service. SAPRA provides inter-session fairness by assigning more bandwidth to sessions with higher audience size, and intra-session fairness by assigning to each layer a drop precedence that matches its importance. SAPRA has a punishment function and a resource utilization maximization function. The former increases the drop percentage of high-rate sessions during periods of congestion. The latter avoids waste of resources when sessions are not using their whole fair share: the remaining bandwidth is equally distributed among other sessions. SAPRA is implemented in edge routers to handle individual traffic aggregated in each service: interior routers are not changed.

In this paper, we propose a simple receiver-driven adaptive mechanism named *SAPRA Adaptive Mechanism* (SAM). Simulation results show that when network resources are fairly distributed using SAPRA, a simple mechanism such as SAM has a good performance.

The remainder of this paper is organized as follows. In Section 2, we describe related adaptive mechanisms. Section 3 describes SAPRA support to receiver-driven adaptive mechanisms and characterizes SAM operation. In Section 4 we evaluate SAM using simulations. Section 5 presents some conclusions.

2 Related Work

McCanne et al. [12] developed the Receiver-driven Layered Multicast (RLM) mechanism, the first receiver-driven adaptive mechanism for scalable sessions. However, RLM has high instability with bursty traffic such as Variable Bit Rate (VBR), poor fairness and low bandwidth utilization [17]. Vicisano et al. [22] described a protocol called Receiver-driven Layered Congestion control (RLC) that complements RLM with a TCP-friendly functionality. However, RLC does not solve issues such as slow convergence and losses provoked by the adaptive mechanism on other flows.

An analysis of the pathological behavior of RLM and RLC [9] showed that their bandwidth inference mechanism is responsible for transient periods of congestion, instability and periodic losses.

Legout et al. [11] developed a receiver-driven protocol for scalable sessions named packet Pair receiver-driven cumulative Layered Multicast (PLM) that uses Packet-pair Probing (PP) [8] and Packet-level Generalized Processor Sharing (PGPS) scheduling [16] to infer the available bandwidth in the path. However, PLM has four major disadvantages. First, PP measurements can have large oscillations, since PP depends on packet size and the burstiness of traffic. Second, PLM requires all packets from all layers to be sent back-to-back and adds an extra bit to the packet header to identify the start of a PP burst. Third, PLM is not suitable for DS scenarios, since all routers have to implement PGPS. Fourth, PGPS uses the max-min fairness definition [1]. However, this fairness definition cannot be applied to discrete sets of rates [19], does not take into account the audience size of sessions, not increasing the number of receivers with good reception quality, and does not punish high-rate sessions.

3 SAM Description

In this section we describe how receivers use SAM to adapt to different network conditions. We also explain briefly how SAPRA support simple receiver-driven adaptive mechanisms, such as SAM. SAPRA is described and evaluated in [15] and a detailed study of its fairness policy can be found in [14].

3.1 SAPRA Support for Receiver-driven Adaptation

SAPRA computes the *fair rate* of a session for each outgoing link of an edge router. The fair rate represents a percentage of the link bandwidth, given by the ratio between the session audience size and the total population of the link. Eq. 1 gives the fair rate F_{ui} of a session S_u in a link i, where n_{ui} is the audience size of S_u and C_i is the bandwidth of the service shared by m_i sessions.

$$F_{ui} = \left(\frac{n_{ui}}{\sum_{x=1}^{m_i} n_{xi}}\right) * C_i \tag{1}$$

The *sustainable rate* of a session is also computed for each outgoing link. The sustainable rate is the larger of the fair rate of the session and the sum of the session rate plus the bandwidth not being used in the link. Eq. 2 gives the sustainable rate U_{ui} of a session S_u in a link i, where r_{ui} is the rate of S_u, and b is the bandwidth not being used in that link.

$$U_{ui} = max(F_{ui}, (r_{ui} + b)) \tag{2}$$

Traffic is marked in each edge router using the fair rate of sessions. SAPRA also provides receivers with periodic reports about the minimum sustainable rate in the path from their session source. Reports are updated with a minimum interval of 1 s. In case the sustainable rate does not change significantly (25% or more in our experiments), reports are suppressed.

Receivers use SAM to add layers when notified of an increase in the sustainable rate. The sustainable rate can increase if: the session has a higher audience; other sessions have a lower audience; there is bandwidth not being used in the path of the session.

Receivers that join later reach an optimal quality level quickly, since they always get a report immediately after joining their session, even if the sustainable rate does not change significantly due to their arrival.

3.2 Overview of SAM Operation

Receivers can join sessions by, for instance, listening to Session Announcement Protocol (SAP) [3] messages, which may include information about the address and rate of each layer. Receivers join first the multicast group for the most important (lowest) layer and then the SAM algorithm controls the reception quality by joining and dropping additional (higher) layers.

The decision to join or drop layers depends on the session sustainable rate and on the existence of congestion in the session path. The sustainable rate, provided by SAPRA, gives SAM an indication of the maximum number of layers that receivers can join. Packet loss is a sign of penalization for high-rate sessions in a congested path. Packets start to be dropped from the less important layer, since SAPRA protects the most important ones. When losses happen in any layer, SAM is triggered to leave layers. In this paper we assume a loss limit of 2.5% as the maximum quality degradation allowed by receivers. We chose this value based on the study made by Kimura et al. [5], which shows that in MPEG-2 layering with Signal to Noise Ratio scalability, 5% of losses in the most important layer in addition to 100% of losses in all other layers, lead to a decrease of the quality of sessions from good to bad accordingly to the ITU-500-R rating [6].

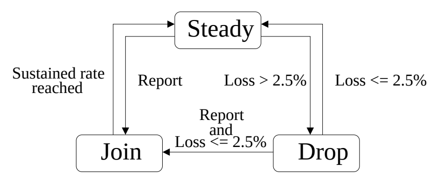

Fig. 1. SAM States

SAM operation is divided into three states as shown in Fig. 1: steady state, join state and drop state. Receivers remain in the steady state as long as they do not receive a report and while losses are lower than 2.5%. Upon receiving a report, receivers enter the join state. If the new sustainable rate is higher than the previous one, receivers increase their reception quality adding layers. Since receivers know in advance the average rate of each layer, they immediately join as much layers as possible. The number of layers they can join is upper bounded by the sustainable rate of their session. After this, receivers return to the steady state. Receivers react to a lower sustainable rate considering the percentage of lost packets and not the report. If losses rise above 2.5%, receivers enter the drop state. In the drop state, receivers drop a layer every 500 ms, the *non-reacting* period, while losses are higher than 2.5%. The non-reacting period avoids over-reacting to losses. With losses equal or below 2.5%, a receiver enters the join state if it receives a new report while in the drop state. Otherwise, it returns to the steady state.

Since we compare SAM with PLM in section 4, we use the same non-reacting period as PLM. Although we use a loss limit of 2.5% and a non-reacting period of 500 ms, these values can be changed to suit other configurations.

4 SAM Evaluation

In this section we present simulations (using NS) that aim to show that SAM has small convergence time, remains stable in the optimal quality level, is fair towards TCP, and allows receivers to use a rate proportional to the audience size of their session and to the amount of bandwidth not being used. We use the three scenarios shown in Fig. 2. A complete set of results can be found in [13].

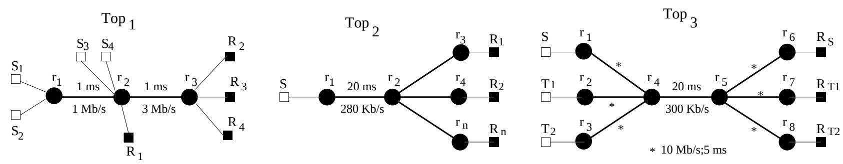

Fig. 2. Topologies

As metrics, we use the rate measured at receivers, presented with a precision of 10 kb/s. As system parameters, we use the type of traffic and the granularity of layers, since these factors affect the adaptation performance. We use layers with exponential rates, which are common in scalable codecs [18], and thin layers as diagnostic tool, since they can identify pathological behaviors.

The first simulation uses the topology *Top1* of Fig. 2. We aim to show how the sustainable rate computed by SAPRA allows SAM to reach an optimal quality level. We use four sessions: S_1 spans seconds 30 through 130, and S_2, S_3 and S_4 span seconds 10 to 240. Sessions S_1, S_2 and S_3 have one receiver each, and S_4 has one receiver until second 170 and five receivers after that. Each session has six layers: the most important layer, l_1, has 32 kb/s and each layer l_i has a rate equal to twice the rate of l_{i-1}. Sessions S_1 and S_2 share the link between routers r_1 and r_2, (r_1, r_2), and S_2, S_3 and S_4 share (r_2, r_3). R_i represents receivers of S_i. We assume that queues have a size of 64 packets, which is the default value in Cisco IOS 12.2, and data packets have 1,000 bytes, a middle value between the 576 bytes MTU of dial-up connections and the 1,500 bytes MTU of ethernet and high speed connections.

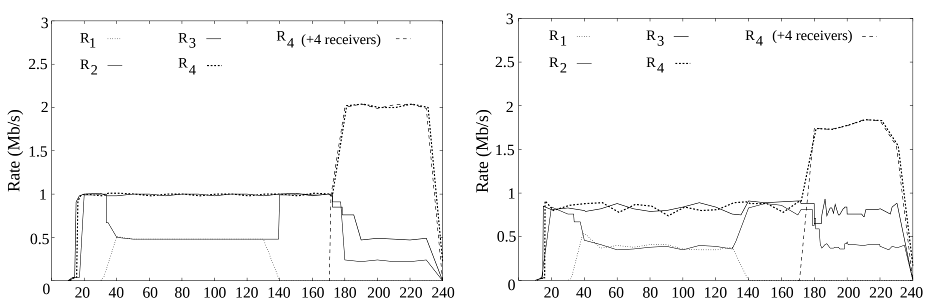

Fig. 3. Rate that receivers get when sessions have CBR (left) and VBR sources (right)

Fig. 3 (left) shows the rate that receivers get when sessions have Constant Bit Rate (CBR) sources. Until $t = 30\,\mathrm{s}$, S_2 is the only session in (r_1, r_2), so it has a sustainable rate of 1 Mb/s in that link. In (r_2, r_3), there are three sessions, S_2, S_3 and S_4. Since these three sessions have one receiver each, SAPRA distributes the link bandwidth equally between them, giving each a sustainable rate of 1 Mb/s. Therefore, R_2, R_3 and R_4 receive a sustainable rate of 1 Mb/s in the first report. This allows them to join four more layers, reaching a rate of 992 kb/s, as show in Fig. 3 (left). At $t = 30\,\mathrm{s}$, S_2 starts sharing (r_1, r_2) with S_1. Therefore, the sustainable rate of S_2 is diminished by half, becoming 500 kb/s, the same value of the sustainable rate of S_1. As a consequence, R_1 joins four layers, reaching 480 kb/s, and R_2 leaves one layer, decreasing its rate from 992 kb/s to 480 kb/s. R_3 and R_4 get a new report since the sustainable rate of their sessions increase more than 25% due to the decrease of the sustainable rate of S_2. However, R_3 and R_4 do not join l_6, maintaining their rate of 992 kb/s. This happens because the new sustainable rate is lower than 2.016 Mb/s, the total rate of the six layers. At $t = 130\,\mathrm{s}$, R_1 leaves and the sustainable rate of S_2 increases from 500 kb/s to 1 Mb/s. Therefore, R_2 gets a new report and grabs the bandwidth not being used by S_1 in (r_1, r_2), reaching again a rate of 992 kb/s. At $t = 170\,\mathrm{s}$, four more receivers join S_4, increasing the sustainable rate of the session from 1 Mb/s to 2.142 Mb/s. Therefore, the five receivers of S_4 receive a new report, which allows the four new receivers to join six layers and the previous receiver to join one more layer, reaching each of them a rate of 2.016 Mb/s. This shows that SAM allows late-join receivers to get the same quality as previous members of an existing session. Due to the higher audience size of S_4, the sustainable rates of S_2 and S_3 decrease to 428 kb/s each, which is insufficient to maintain their quality level. R_2 is the first to react to losses leaving l_5 with 2.54% of losses and leaving l_4 500 ms after that, with 9.68% of losses. Due to the reaction of R_2, the sustainable rate of S_3 increases to 760 kb/s, which is sufficient to maintain l_4. This shows that with SAM, competition for bandwidth does not increase quality oscillations.

Fig. 3 (right) shows SAM behavior when sessions have a VBR source. Each layer has a mean rate equal to its rate with CBR, a maximum and minimum rate 1.5 times higher and lower than the mean rate, respectively, and a burst time of 2 s with a deviation of 0.5 s. Results show that SAM is able to adjust fairly the reception quality even when sessions have oscillatory rates. We can observe that S_2 and S_3 have a higher rate with VBR than with CBR after $t = 170\,\mathrm{s}$. This happens because they have a higher sustainable rate, since S_4 has a rate lower with VBR than with CBR.

The simulation with topology *Top1* of Fig. 2 evaluates the operation of SAM. Results shows that reports provided by SAPRA allow a simple adaptive mechanism, such as SAM, to keep receivers rate close to the sustainable rate of their sessions, guaranteeing inter-session fairness and increasing bandwidth utilization. Results also show that a loss threshold of 2.5% does not lead to quality oscillations.

In what concerns the convergence time, stability and fairness with TCP, Legout et al. show that PLM performs better than RLM and RLC. Hence, we compare SAM with the results presented by Legout et al. for PLM. For that, we use the same scenarios used by Legout. et al.. These simulations use the topologies *Top2* and *Top3,* shown in Fig. 2.

We use the topology *Top2* shown in Fig. 2 to evaluate the time SAM takes to convergence to an optimal quality level, its accuracy and stability. We also show the performance that SAM would have, if receivers didn't know in advance the rate of each layer. Links (r_2, r_n) have a bandwidth uniformly chosen between [500,1000] kb/s and a delay uniformly chosen between [5,150] ms. We use one session, S, and layers with a thin granularity of 50 kb/s. At t = 5 s, twenty receivers join S. From t = 30 s to t = 50 s, a receiver joins S every 5 s. At t = 80 s, five more receivers join this session. Each receiver is positioned in a different leaf router. We use the packet and queue size used for PLM, i.e., packets with 500 bytes and queues with 20 packets.

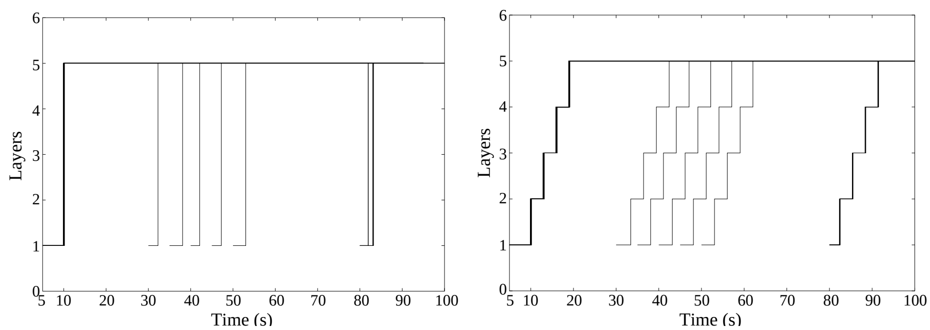

Fig. 4. Convergence time with (left) and without (right) knowledge of layers rates

Fig. 4 (left) shows that the first twenty receivers start to converge to their optimal rate at t = 10 s, 5 s after joining S, while late-join receivers wait a little less (3 s) to converge. This happens because receivers only start to converge after receiving the first report. First receivers wait longer, since the fair rate of S has to be computed for the entire path, and so the first report is originated by the node nearest to the source. Nevertheless, all receivers converge immediately to an optimal rate, which is maintained without losses.

Fig. 4 (right) shows that the convergence time would be slower if receivers did not know in advance the average rate of each layer. This happens because receivers would have to wait before joining each layer, in order to estimate the current rate of the session and predict the rate of the next higher layer. In these experiments, receivers use an exponential equation to estimate the average rate of each received layer, and predict that the next layer has a rate equal to the last joined layer.

This simulation shows that neither the audience size or late-joins influence the convergence time and stability of SAM. The results shown in Fig. 4 (left) are similar to the ones presented by Legout et al. for PLM [10], except that with

PLM receivers start to converge 2 s after joining a session. This happens because with PLM receivers are notified only about the available bandwidth in the path and not about the sustainable rate of their session.

We use the topology *Top3* of Fig. 2 to evaluate the behavior of SAM in the presence of TCP flows, which are handled by SAPRA as scalable sessions with one layer and one receiver. This scenario has one scalable session, S, and two TCP flows, T_1 and T_2. S has one receiver, R_S, and layers with granularity of 20 kb/s. T_1 starts at $t = 0$ s, S_1 at $t = 20$ s and T_2 at $t = 60$ s.

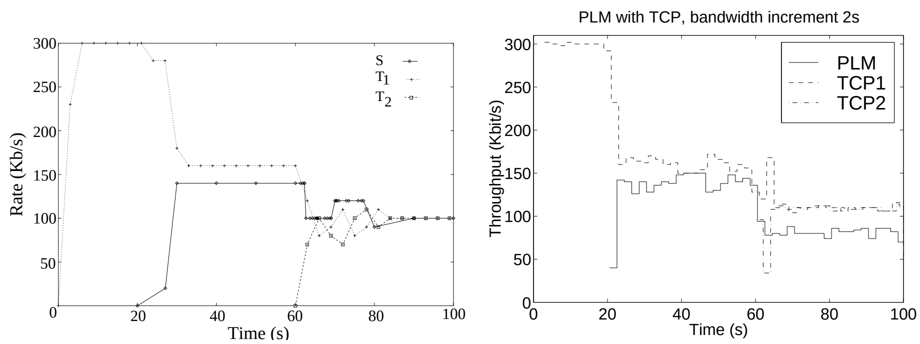

Fig. 5. Fairness of SAM (left) and PLM (right) with TCP

Fig. 5 (left)[4] shows that SAM is fair in the presence of TCP. R_S joins the lowest layer at $t = 20$ s, reaching a rate of 32 kb/s. At $t = 27$ s, it increases its rate after receiving the first report. Since S has only one receiver, the bandwidth of (r_4, r_5) is equally divided between S and T_1. Therefore, after $t = 20$ s, S and T_1 have a fair rate of 150 kb/s, and so R_S joins seven layers, reaching a rate of 140 kb/s. Since S does not use 10 kb/s of its fair share, the rate of T_1 reaches 160 kb/s. When T_2 starts, the fair rates of S, T_1 and T_2 reach 100 kb/s. R_S starts to experience losses and decreases its rate to 100 kb/s, but it maintains seven layers since losses are lower than 2.5%.

	60 s	63 s	66 s	69 s	72 s	75 s	78 s	81 s	84 s
S	140	100	100	120	120	120	120	90	100
T_1	160	120	80	90	110	80	70	110	100
T_2	0	70	100	80	70	100	110	90	100
Total	300	290	280	290	300	300	300	290	300

Fig. 6. Link (r_2, r_4) utilization in kb/s

Due to T_1 and T_2 oscillations until $t = 84$ s, R_S grabs the bandwidth not being used by the TCP flows, increasing its rate to 120 kb/s. From then until the end of the simulation, the rate of S, T_1 and T_2 stabilizes at 100 kb/s. In the meantime, R_S leaves layer seven with losses of 2.91%, but maintains the rate of 120 kb/s, which is the maximum possible rate with six layers. At $t = 80$ s, R_S leaves layer six, since losses reach 3.28%. Fig. 6 shows that the bandwidth of (r_4, r_5) is completely used, except for the interval from $t = 63$ s to $t = 81$ s, where the utilization rate decreases to 98.1% due to T_1 and T_2 oscillations.

[4] Fig. 5 (right) was taken from [10] with the author's permission.

The results shown in Fig. 5 show that SAM and PLM are fair in the presence of TCP, but SAM has smaller quality oscillations and a fairer distribution of bandwidth: after $t = 60\,\mathrm{s}$, S, T_1 and T_2 get the same share of bandwidth, while in the case of PLM, the two TCP flows get a higher share.

5 Conclusion

Receiver-driven adaptive mechanisms can accommodate heterogeneity when combined with scalable encoding. However, receivers are only motivated to adapt if the network guarantees a fair distribution of bandwidth and also punishes receivers that do not adjust their rate in case of congestion.

This paper describes and evaluates SAM, a receiver-driven adaptive mechanism based upon SAPRA. SAPRA is a signaling protocol that has the required punishment and fairness properties. SAM controls the reception quality by joining and dropping layers. The sustainable rate, provided by SAPRA, indicates to SAM the maximum number of layers that receivers can join, while the measured packet losses triggers SAM to drop layers.

Simulation results show that with SAM, receivers get always a rate near the sustainable rate of their session, independently of the number of sessions and their audience size. Results also show that SAM has small convergence time and remains stable even in the presence of bursty traffic, such as VBR. Compared to PLM, SAM has less quality oscillations, is fairer in the presence of TCP flows and requires few changes in the network structure.

As major improvement, SAM motivates receivers to adapt, since SAPRA guarantees a fair distribution of bandwidth and the punishment of high-rate sessions.

References

1. Dimitri Bertsekas and Robert Gallager. *Data Networks*. Prentice-Hall, Englewood Cliffs, New Jersey, 1987.
2. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated service. Request for Comments 2475, Internet Engineering Task Force, December 1998.
3. M. Handley, C. Perkins, and E. Whelan. Session announcement protocol. Request for Comments 2974, Internet Engineering Task Force, October 2000.
4. J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski. Assured forwarding PHB group. Request for Comments 2597, Internet Engineering Task Force, June 1999.
5. Jun ichi Kimura, Fouad A. Tobagi, Jose-Miguel Pulido, and Peder J. Emstad. Perceived quality and bandwidth characterization of layered MPEG-2 video encoding. In *Proc. of SPIE International Symposium*, Boston, Massachussetes, USA, September 1999.
6. ITU-500-R. "Methodology for the subjective assessment of quality of television pictures". Itu-500-r recommendation bt.500-8, ITU, 1998.

7. Mathias Johanson. Scalable video conferencing using subband transform coding and layered multicast transmission. In *Proc. of International Conference on Signal Processing Applications and Technology (ICSPAT)*, Orlando, Florida, USA, November 1999.
8. Srinivasan Keshav. A control-theoretic approach to flow control. In *Proc. of SIGCOMM Symposium on Communications Architectures and Protocols*, pages 3–15, Zürich, Switzerland, September 1991. ACM. also in *Computer Communication Review* 21 (4), Sep. 1991.
9. Arnaud Legout and Ernst Biersack. Pathological behaviors for RLM and RLC. In *Proc. of the International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, Chapel Hill, North Carolina, USA, June 2000.
10. Arnaud Legout and Ernst W. Biersack. PLM: Fast convergence for cumulative layered multicast transmission schemes. technical report, Institut Eurécom, Sophia-Antipolis, France, November 1999.
11. Arnaud Legout and Ernst W. Biersack. PLM: Fast convergence for cumulative layered multicast transmission schemes. In *Proc. of ACM SIGMETRICS performance evaluation review*, Santa Clara, California, USA, June 2000.
12. Steven McCanne, Van Jacobson, and Martin Vetterli. Receiver-driven layered multicast. In *Proc. of SIGCOMM Symposium on Communications Architectures and Protocols*, pages 117–130, Palo Alto, California, USA, August 1996.
13. Paulo Mendes. "SAPRA: Session-Aware Popurality-based Resource Allocation fairness protocol". http://www.cs.columbia.edu/~mendes/sapra.html.
14. Paulo Mendes, Henning Schulzrinne, and Edmundo Monteiro. Session-aware popularity resource allocation for assured differentiated services. In *Proc. of the Second IFIP-TC6 Networking Conference*, Pisa, Italy, May 2002.
15. Paulo Mendes, Henning Schulzrinne, and Edmundo Monteiro. Signaling protocol for session-aware popularity-based resource allocation. In *To appear in Proc. of the IFIP/IEEE International Conference on Management of Multimedia Networks and Services*, Santa Barbara, California, USA, October 2002.
16. Abhay K. Parekh and Robert G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: The single node case. *Journal of IEEE/ACM Transactions on Networking*, 1(3):344–357, June 1993.
17. Gopalakrishnan Raman, James Griffioen, Gisli Hjalmtysson, and Cormac Sreenan. Stability and fairness issues in layered multicast. In *Proc. of the International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, Basking Ridge, New Jersey, USA, June 1999.
18. Kenneth Rose and Shankar L. Regunathan. Toward optimality in scalable predictive coding. *Journal of IEEE Transactions on Image Processing*, 10(7):965–976, July 2001.
19. Dan Rubenstein, Jim Kurose, and Don Towsley. The impact of multicast layering on network fairness. In *Proc. of SIGCOMM Symposium on Communications Architectures and Protocols*, Cambridge, Massachussetes, USA, September 1999.
20. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: a transport protocol for real-time applications. Request for Comments 1889, Internet Engineering Task Force, January 1996.
21. David Taubman and Avideh Zakhor. Multirate 3-D subband coding of video. *Journal of IEEE Transactions on Image Processing*, 3(5):572–588, September 1994.
22. Lorenzo Vicisano, Luigi Rizzo, and Jon Crowcroft. TCP-Like congestion control for layered multicast data transfer. In *Proc. of the Conference on Computer Communications (IEEE Infocom)*, San Francisco, California, USA, March/April 1998.