ONLINE SMOOTHING OF VBR VIDEO STREAMS IN SYSTEMS WITH VARIABLE AVAILABLE BANDWIDTH

Pietro Camarda, Antonio De Gioia, Domenico Striccoli
Politecnico di Bari – Dip. di Elettrotecnica ed Elettronica, Via E. Orabona, 4 – 70125 Bari (Italy)

Keywords: Video Distribution, VBR traffic, Online Smoothing, Available Bandwidth

Abstract: Compressed multimedia transmission is assuming a growing importance in the telecommunication world. However, the high data rate variability of compressed video over multiple time scales makes an efficient bandwidth resource utilization difficult to obtain. Smoothing techniques is one of the approaches exploited to face this problem. Various smoothing algorithms have been proposed, that reduce the peak rate and high rate variability of video streams by efficiently prefetching video data to be transmitted over the network. However, all previous algorithms consider a constant available bandwidth. Such a constraint can be hardly verified in modern telecommunication networks. In this paper a novel online smoothing algorithm is proposed, that performs data scheduling by taking into account the residual available bandwidth, and at the same time minimizing rate variability changes. This algorithm can be fully exploited for online smoothing of video applications that want to tolerate very short playback delays. Numerical results show that the proposed algorithm is very effective for online smoothing purposes in a link sharing environment.

1 INTRODUCTION

The increasing computational capacity of modern computers together with the sustained growth of telecommunication networks bandwidth allow multimedia streaming through bursty Variable Bit Rate (VBR) stream transmission. As it can be seen from Figure 1, the VBR source behavior makes the optimization of network utilization more difficult while providing at the same time Quality of Service (QoS) guarantees, i.e., low delays and jitters, low data losses, and so on (Kurose and Ross, 2000) (Zhang et al., 1997).

To reduce the total amount of bandwidth assigned to video streams, work-ahead smoothing techniques can be exploited (Salehi et al., 1998) (Feng and Rexford, 1997). These techniques are based on the reduction of the peak rate and the bit rate variability of network streams; they consist in transmitting, ahead of playback time, pieces of the same film with a constant bit rate that varies from piece to piece according to a scheduling algorithm that smoothes the bursty behaviour of video streams. On the transmission side a buffer regularizes data transmission, while on the receiving side the frames are temporarily stored in a client buffer and extracted during the decoding process. Obviously, the bit rate must be chosen appropriately in order to avoid buffer overflow and underflow, ensuring a continuous playback at the client side. The client smoothing buffer size determines number and duration of the Constant Bit Rate (CBR) pieces that characterize the smoothed video stream. An increase of the smoothing buffer size generally produces a smaller number of bandwidth changes among CBR segments and a peak rate and rate variability reduction of smoothed video streams (Zhang et al., 1997) (Salehi et al., 1998).

As described in (Feng and Rexford, 1997), a VBR video stream is composed by N video frames, each of them of size d_i bytes $(1 \le i \le N)$. On the server side, the stream data enter a buffer whose capacity is b bytes, and the buffer output gives the smoothed video stream data. At the client side, the smoothed video data enter the buffer and the original unsmoothed video frame sequence leaves the buffer. Let us now consider the client buffer model in the k^{th} discrete frame time, that is, the time interval in

Striccoli D., Camarda P. and De Gioia A. (2004).
ONLINE SMOOTHING OF VBR VIDEO STREAMS IN SYSTEMS WITH VARIABLE AVAILABLE BANDWIDTH.
In *Proceedings of the First International Conference on E-Business and Telecommunication Networks*, pages 369-374
DOI: 10.5220/0001388103690374

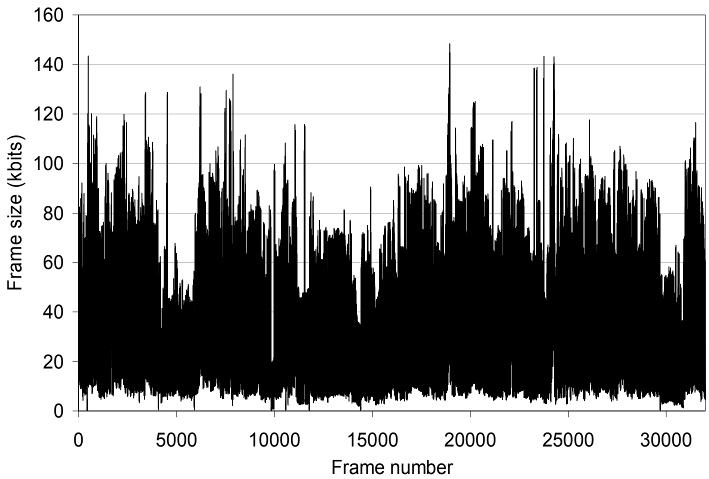

Figure 1: 32.000 video frames of the "Simpson's" cartoon, codified with the MPEG-1 algorithm.

which a video frame is transmitted. To guarantee a feasible transmission, the cumulative amount of data consumed by the client buffer at discrete time *k*:

$$D(k) = \sum_{i=1}^{k} d_i \qquad (1)$$

should arrive quickly enough to avoid buffer underflow At the same time, to avoid buffer overflow, at time *k* the client buffer should not receive more data than:

$$B(k) = b + \sum_{i=1}^{k} d_i \qquad (2)$$

The cumulative smoothed data have to respect the following bounds:

$$D(k) \le \sum_{i=1}^{k} s(i) \le B(k) \qquad (3)$$

where $s(i)$ represents the smoothed stream bit rate in the discrete frame time *i*, while $S(k) = \sum_{i=1}^{k} s(i)$ are the cumulative smoothed data arrived to the client buffer until frame time *k*. The smoothed stream transmission plan will result in a number of CBR segments, and the correspondent stream evolution is given by a monotonically increasing and piecewise-linear path that lies between the $D(k)$ and $B(k)$ curves, as can be shown in Figure 2a. According to the definition given in (Feng and Rexford, 1997), each CBR segment defines a run that can be considered as a frontier of possible starting points for the next run.

As described in (Feng and Rexford, 1997), different types of smoothing algorithms can be implemented; all of them transform the highly bursty video stream bit rate behaviour into a series of CBR pieces. The scheduling algorithm regulates each of the CBR bit rate values in such a way to respect the buffer constraints $D(k)$ and $B(k)$. Now let us examine more in detail some of the most common smoothing algorithms proposed in literature.

The Critical Bandwidth Allocation (CBA) algorithm minimizes the number of bandwidth increases as follows. For bandwidth decreases, the rate decrease starts in the earliest point in time, when the previous run hits the lower bound curve. For

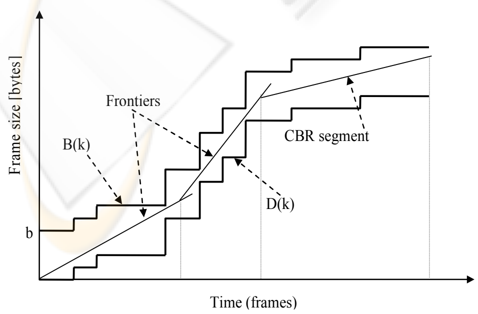

Figure 2a: An example of smoothed video stream transmission plan.

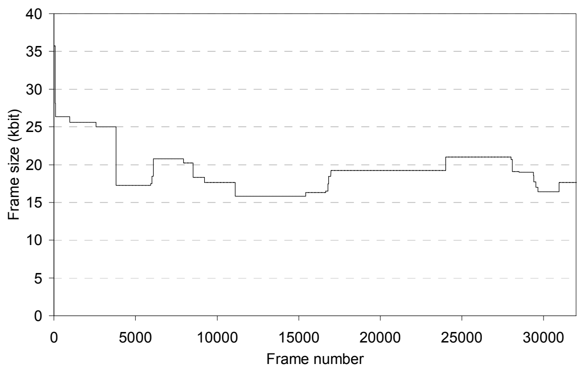

Figure 2b: The "Simpson's" cartoon, MVBA smoothed (buffer size 1024 kbytes).

bandwidth increases, the starting point of the next run is chosen in such a way that it extends as far as possible. In this way, the transmission plan has the smallest peak bandwidth requirement and the minimum number of bandwidth increases (Feng and Sechrest, 1995).

The Minimum Changes Bandwidth Allocation (MCBA) algorithm minimizes both the number of bandwidth increases and decreases by performing the same research of the next run starting points made in the CBA algorithm, for bandwidth increases. This results in a transmission plan that minimizes the number of bandwidth increases, bandwidth decreases, and also the peak bandwidth requirement (Feng et al., 1996).

The Minimum Variability Bandwidth Allocation (MVBA) algorithm reduces the bandwidth change variability by searching, for each CBR piece, the earliest point in time in which a bandwidth increase or decrease can happen, obviously respecting at the same time the lower and upper constraints. The corresponding transmission plan gradually performs rate changes, assuring in this way the smallest variability of rate changes, at the expense of a greater number of CBR pieces if compared with CBA and MCBA algorithms (Salehi et al., 1998) (Feng and Rexford, 1997).

The Piecewise Constant Rate Transmission and Transport (PCRTT) algorithm divides the video flow into time intervals with fixed temporal dimension in which the assigned bandwidth changes. The transmission plan is obtained by creating a CBR piece in each time interval; the bit rate is obtained by connecting the extreme points of the lower bound curve. The segment slope represents the bit rate of the CBR piece. Finally, the segment is raised in such a way to be included among the two bound curves, avoiding buffer underflow (Feng and Rexford, 1997) (McManus and Ross, 1996). An enhanced version of the PCRTT algorithm, called e-PCRTT, can be found in (Hadar and Cohen, 2001). It behaves like the PCRTT algorithm, but it is capable to reach the same transmission plans of the original PCRTT algorithm with smaller smoothing buffers, or alternatively, given the same buffer sizes, it reduces the number of bandwidth changes. Furthermore, it reduces also the playback delay if compared with the PCRTT algorithm.

The choice of each of the mentioned algorithms depends on what aspect of data transmission has to be optimized among the peak rate, number, variability and periodicity of bandwidth changes.

An example of the application of the smoothing algorithm can be observed in Figure 2b.

All the mentioned smoothing techniques mainly apply to stored video traffic, where all source video data are a priori known and can be optimally scheduled in an "off-line" manner. The optimality of the offline algorithms derive from the a priori knowledge of the entire video data to be scheduled. Nevertheless, the algorithms can also be applied in an "on-line" manner, in limited temporal windows. In this case, smoothing algorithms generally have a limited a priori knowledge of frame sizes in short consecutive temporal observation windows, thus reducing video burstiness in a smaller time scale and with a less efficient transmission plans (Rexford et al., 1997); nevertheless, they remain. effective to reduce peak rate and rate variability in the temporal window of interest. In this context of online video smoothing the proposed algorithm can be introduced and exploited.

2 THE ABSA SMOOTHING ALGORITHM

In this section the Available Bandwidth Smoothing Algorithm (ABSA) is proposed and implemented, taking into account not only the parameters of the smoothing algorithms proposed in literature, i.e., buffer size and unsmoothed data, but also the available residual bandwidth, that fluctuates in time due to the presence of other traffic running into the network. The ABSA algorithm represents a substantial novelty if compared with the other classical smoothing algorithms already analyzed; it can be efficiently exploited in an online smoothing context. The a priori knowledge of available bandwidth resources in the considered time window is an important requirement; thus available bandwidth necessary for implementing the stream bandwidth plan, is supposed to be a priori known through bandwidth estimation techniques.

Let us suppose to analyze the video data transmission in a temporal window of length N video frames., at the same time knowing the temporal evolution of available bandwidth $w(k)$ in the frame time *k*. The two bounds (1) and (2) change as follows. First of all:

$s(k) \le w(k)$, $0 \le k \le N$

that is

$S(k) - S(k-1) \le w(k) \Rightarrow S(k) \le w(k) + S(k-1)$,(4)

Furthermore, according to (3), it has to be:

$S(k) \le B(k)$

Thus, calling:

$U_{S,B,w}(k) = \min[B(k), w(k) + S(k-1)]$

it follows:

$S(k) \le U_{S,B,w}(k)$, $0 \le k \le N$

It has to be pointed out that the bound $U_{S,B,w}(k)$ depends on the available bandwidth w, the transmission plan $S(k)$ and the $B(k)$ curve..

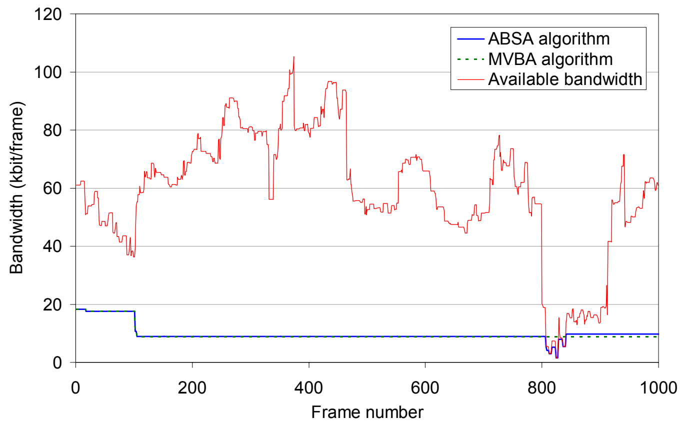

Figure 3: A comparison between the ABSA and MVBA smoothing algorithms. The window size is 1000 video frames.

Similarly, exploiting again(3) and (4): $S(k-1) \geq S(k) - w(k)$; $S(k) \geq D(k)$.

Defining the function:

$L_{S,D,w}(k) = \max[D(k), S(k+1) - w(k+1)]$

with the obvious further condition:

$L_{S,D,w}(N) = D(N)$

It is satisfied that:

$S(k) \geq L_{S,D,w}(k)$, $0 \leq k \leq N$

We have:

$$L_{S,D,w}(k) \leq S(k) \leq U_{S,B,w}(k) \quad 0 \leq k \leq N \qquad (5)$$

The main problem of the $L_{S,D,w}(k)$ and $U_{S,B,w}(k)$ calculation is that they depend on the scheduled data in the previous and following steps. The dependence of $L_{S,D,w}(k)$ and $U_{S,B,w}(k)$ from the scheduled data $S(k)$ is thus eliminated by introducing the functions

$U'_{B,w}(k) = \min[B(k), U'_{B,w}(k-1) + w(k)]$,

with the initial condition $U'_{B,w}(0) = B(0)$, and

$L'_{D,w}(k) = \max[D(k), L'_{D,w}(k+1) - w(k+1)]$

with the initial condition $L'_{D,w}(N) = D(N)$.

It can be demonstrated that, if $S(k)$ is a feasible transmission (that is, $L_{S,D,w}(k) \leq S(k) \leq U_{S,B,w}(k)$ $\forall k$),then:

$$U'_{B,w} \geq U_{S,B,w}\,;\; L'_{D,w} \leq L_{S,D,w} \qquad (6)$$

Proof:

It is valid that

$U_{S,B,w}(0) = U'_{B,w}(0) = B(0)$

Let us proceed by induction and suppose that (6) is valid in *k*. We have to show that (6) is valid in *k+1*. It will be:

$U'_{B,w}(k+1) = \min[B(k+1), U'_{B,w}(k) + w(k+1)] \geq$
$\geq \min[B(k+1), U_{S,B,w}(k) + w(k+1)] \geq$
$\geq \min[B(k+1), S(k) + w(k+1)] = U_{S,B,w}(k+1)$.

This demonstrates the first of (6). Similarly, the second of (6) can easily be demonstrated.

Thus is valid that:

$$L'_{D,w} \leq S \leq U'_{B,w} \qquad (7)$$

with the further advantage that $L'_{D,w}$ and $U'_{B,w}$ are two bounds for S, that are independent from S itself. For this reason, a first approach to find the transmission plan S is to apply the MVBA smoothing algorithm as described in (Salehi at al., 1998) with the two boundaries expressed by the $L'_{D,w}$ and $U'_{B,w}$ curves. If a frame time is found in which $L'_{D,w} > U'_{B,w}$, the corresponding transmission plan S will not be feasible and the smoothing algorithm can not be applied due to the strong

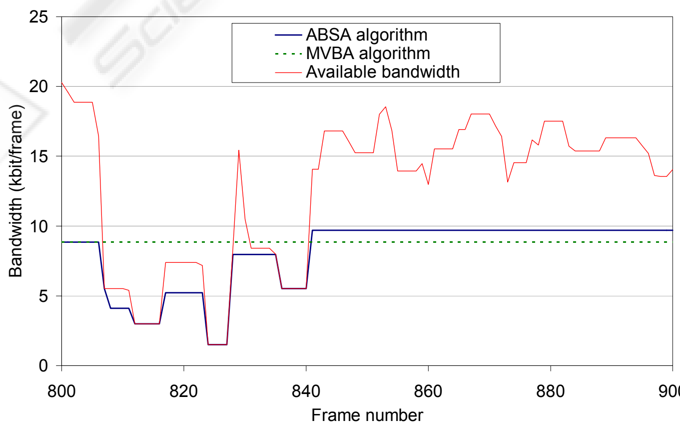

Figure 4: ABSA and MVBA in the critical time zone, where the available bandwidth consistently lowers.

limitation in available bandwidth.

Let us now suppose to have verified (7) and consequently (5), where the transmission plan $S(k)$ has been obtained through the optimal MVBA smoothing algorithm (Salehi at al., 1998). In this case, the ABSA smoothing algorithm behaves exactly like the MVBA optimal smoothing algorithm. If instead (5) is not effectively verified for each $0 \le k \le N$ after have calculated (7) the ABSA algorithm adjusts the CBR segment slopes of S (and consequently the constraint curves $L_{S,D,w}$ and $U_{S,B,w}$) in such a way that (5) is verified for each k, at the same time maintaining the scheduling S the closest possible to the optimal MVBA smoothing algorithm curve. To better explain this concept, let us suppose to have verified (5) for $0 \le k \le k_1 - 1$, and that in frame time k_1, (5) is not verified. The ABSA algorithm progressively increases, in an iterative way, the value of $S(k_1)$ since it verifies (5) in $k = k_1$. This final step will surely be reached, since if we assign:

$$S(k_1) = U'_{B,w}(k_1) . \qquad (8)$$

it can be easily verified that:

$$U_{U'_{B,w},B,w}(k_1) \le U'_{B,w}(k_1) \le L_{U'_{B,w},B,w}(k_1) \qquad (9)$$

After the calculation of $S(k_1)$, the optimal MVBA algorithm is applied again starting from $k = k_1$ and ending at $k = N$ and then verifying (5) in $[k_1 + 1, N]$ through the same procedure previously illustrated.

3 NUMERICAL RESULTS

In this section some numerical results are provided, to testify the effectiveness of the proposed algorithm. Different simulation scenarios have been considered, and the algorithm performance has been tested for different video stream types, different smoothing buffers and temporal observation window sizes. The available bandwidth information, in this specific case, has been derived in the hypothesis that other smoothed video streams form the background traffic, consisting of 12 MVBA-smoothed video streams. Flow aggregation has been performed randomly choosing the video stream starting points and deriving the total bandwidth occupied by stream aggregation simply by adding the number of bits contained in each video streams frame, in each discrete time unit given by a frame transmission time. In this case, the so obtained bandwidth is expressed in bit/frame. Supposed a channel capacity C, the available bandwidth has been derived simply subtracting the bandwidth exploited by stream aggregation previously calculated to C, in each frame time and supposing to know in advance all the flow aggregation information in each frame time.

Established the temporal observation window size (in frame number), the ABSA algorithm is then applied. A first comparison among the ABSA and MVBA smoothing techniques is illustrated in Figure 3. A temporal window of size 1000 video frame has been chosen; taking into account a constant frame rate of 25 frames/s, the temporal window size is 40 s. In this window a piece of the "James Bond: Goldfinger" video stream, MPEG-1 codified, has been smoothed with both the MVBA and the ABSA smoothing algorithms, highlighting the main differences between them. From Figure 3 it can be noted that there is a strong available bandwidth reduction, beginning from the 807th frame until the 843th frame, due to high bandwidth requirements by flow aggregation already present in the network link. During this period the ABSA smoothing algorithm, represented through a continuous blue line, follows perfectly the available bandwidth curve (depicted as a continuous red line), while the MVBA algorithm crosses the red line, testifying a frame loss that occurs until the available bandwidth curve raises again. This important particular can be better

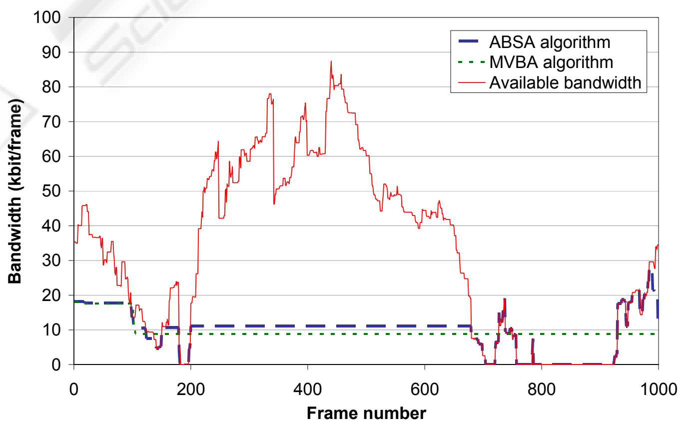

Figure 5: The ABSA and MVBA smoothing algorithms, for stronger available bandwidth reduction.

appreciated by observing Figure 4, in which an enlargement of the critical time interval, in which the available bandwidth falls down quickly, is reported.

As can be noted from Figure 4, the ABSA algorithm transmission plan raises again after have passed the strong bandwidth reduction zone (after the 843th video frame), continuing with a long CBR segment, according with the ABSA algorithm purpose of performing the MVBA smoothing technique whenever possible. Anyway, in the last time period the bandwidth level reached by the ABSA algorithm is higher than the corresponding MBA offline smoothing transmission plan. This is obvious, since the ABSA algorithm has to compensate in some way the lower bit rates scheduled during the strong bandwidth reduction zone. In Figure 5 another comparison among the two proposed algorithms is depicted, in more critical available bandwidth conditions.

In Figure 5 two “critical zones” , in which the available bandwidth strongly falls down, can be observed; the first is clearly visible on the left of the figure, beginning in the 105th video frame and ending at the 200th video frame. In this first critical zone, the available bandwidth is sometimes null. The second critical zone starts from the 680th video frame and ends at the 930th video frame. In this second critical zone a major lacking of available bandwidth can be noted, and the time interval in which available bandwidth reaches zero is longer. The utilization of the MVBA algorithm would result in very consistent frame losses, while the ABSA algorithm produces no losses all the time, perfectly following the available bandwidth curve. In the second critical zone on the right, it can be noted that the ABSA algorithm continues following the available bandwidth curve long after the critical zone is finished, since there is no other way to recover from the heavy resource lacking previously occurred. It can be easily verified that the ABSA algorithm behaviour appears very effective also if applied to other types of films, with different smoothing buffer and/or temporal window sizes.

4 CONCLUSIONS AND FUTURE WORK

In this paper, a novel smoothing algorithm, called ABSA algorithm, has been developed and analyzed. The main novelty of this algorithm is that it is able to take into account residual available bandwidth fluctuations, trying to adapt the smoothing transmission plan to available bandwidth resources, at the same time trying to keep, whenever possible, the main advantages of the MVBA smoothing algorithm. Numerical results show that the ABSA algorithm performs better than the MVBA algorithm in all cases of reduced available bandwidth resources, avoiding packet losses also in critical free bandwidth conditions. This makes the ABSA algorithm suitable for a more efficient packet transmission planning. Nevertheless, some other aspects of the ABSA algorithm have to be investigated, like more efficient ways to modify the ABSA transmission plan to minimize losses, or the ABSA algorithm enhancement for a flow aggregation. This last aspect would be of a great use to avoid scalability problems, at the same time optimizing bandwidth resource saving.

REFERENCES

Kurose, J. & Ross, K., 2000. *Computer Networking: A Top-Down Approach Featuring the Internet*. Addison Wesley Longman.

Zhang, Z.-L., Kurose, J., Salehi, J. D. & Towsley, D., 1997. Smoothing, Statistical Multiplexing and Call Admission Control for Stored Video. *IEEE Journal on Selected Areas in Communications*, 15(6), 1148-1166.

Salehi, J.D., Zhang, Z.-L., Kurose, J. & Towsley, D., 1998. Supporting Stored Video: Reducing Rate Variability and End-to-End Resource Requirements Through Optimal Smoothing. *IEEE/ACM Transactions On Networking*, 6(4), 397-410.

Feng, W.-C. & Rexford, J., 1997. A Comparison of Bandwidth Smoothing Techniques for the Transmission of Prerecorded Compressed Video. In *IEEE INFOCOM*.

Feng, W. & Sechrest, S., 1995. Critical Bandwidth Allocation for Delivery of Compressed Video. *Computer Communication*, 18, 709-717.

Feng, W., Jahanian, F. & Sechrest, S., 1996. Optimal Buffering for the Delivery of Compressed Prerecorded Video. In *ACM SIGMETRICS*.

McManus, J.M. & Ross, K.W., 1996. Video on Demand over ATM: Constant-rate Transmission and Transport. *IEEE Journal on Selected Areas in Communications*, 4(6), 1087-1098.

Hadar, O. & Cohen, R., 2001. PCRTT Enhancement for Off-Line Video Smoothing. *Journal of Real-Time Imaging*, 7(3), 301-314.

Rexford, J., Sen, S., Dey, J., Feng, W., Kurose, J., Stankovic, J. & Towsley, D., 1997. Online Smoothing of Live Variable-Bit-Rate Video. In *7th Workshop Network and Op. Systems Support for Digital Audio and Video*.