University of Würzburg
Institute of Computer Science
Research Report Series

Discrete-time Analysis of a Finite Buffer with VBR MPEG Video Traffic Input

O. Rose

Report No. 150 September 1996

Institute of Computer Science, University of Würzburg
Am Hubland, 97074 Würzburg, Germany
Tel.: +49-931-8885508, Fax: +49-931-8884601
e-mail: rose@informatik.uni-wuerzburg.de

Abstract

Currently, a lot of interest is paid to multimedia traffic in general and to video traffic in particular. Most of the studies deal with appropriate models for this complex type of traffic and use trace-driven simulations to assess the performance of high-speed networks carrying this type of traffic. In contrast, we develop two models of different correlation complexity and apply them in a discrete-time analysis of a buffer with variable bit rate (VBR) MPEG video traffic input. The algorithms are easy to implement and provide good cell loss estimates over a wide range of buffer sizes.

1 Introduction

In recent years, the modeling of VBR video traffic attracted a lot of interest. A large variety of models based on different approaches can be found in the current teletraffic literature: histogram models (e.g., [13]), Markovian models (e.g., [9]), autoregressive models (e.g., [1]), TES models (e.g., [10]), and even self-similar models (e.g., [4]).

In most cases, however, the papers focus on the modeling itself or on the application of the models in simulations. Due to the complex correlation structure of video traffic [11], there is a lack of studies which analytically assess the performance of buffers with video input. In [12], the authors determine Usage Parameter Control (UPC) parameters of MPEG video streams based on a discrete-time analysis. Since only a part of the frame-by-frame correlations of the video traffic is covered by their model, the usage of the analysis is limited to rather small systems.

In this paper, we present a discrete-time analysis of a buffer with VBR MPEG video traffic input. To that extend, two video source models are developed: the *basic model* reflects the correlations within a *Group of Pictures (GOP)*, whereas the *extended model* also includes GOP-by-GOP correlations. To keep the analysis numerically tractable, we apply ideas from *fluid flow simulation* in the development of both the system and the video traffic models.

The paper is organized as follows. Section 2 gives an outline of statistical properties of MPEG video traffic and introduces the basic video model. The system model is presented in Section 3. In Section 4, the discrete-time analysis algorithm for the system with basic video model input is developed. This approach is extended in Section 5 to a video model including GOP-by-GOP correlations. The numerical results are presented in Section 6.

2 Basic MPEG video source model

Due to the high bandwidth needs of uncompressed video streams, several coding algorithms for the compression of video streams are in discussion. Since the MPEG (ISO - Moving Picture Expert Group) suite of coding schemes [5, 6, 8] is currently used in a large variety of applications for the compression of video data, we will dedicate our interest to the modeling of the frame size process at the output of a MPEG-1 video coder.

A video sequence consists of a series of frames, each containing a two-dimensional array of pixels. The number of frames per second as well as the number of lines per frame and pixels per line depend on national standards. For each pixel, both luminance and chrominance information is stored. The compression algorithm is used to reduce the data rate before transmitting the video stream over communication networks.

In MPEG coded streams, there are three types of frames, each using a slightly different coding scheme:

I-frames use only intra-frame coding, based on the discrete cosine transform and entropy coding;

P-frames use a similar coding algorithm to I-frames, but with the addition of motion compensation with respect to the previous I- or P-frame;

B-frames are similar to P-frames, except that the motion compensation can be with respect to the previous I- or P-frame, the next I- or P-frame, or an interpolation between them.

Typically, I-frames require more bits than P-frames. B-frames have the lowest bandwidth requirement. The different ways in coding frames result in different statistical properties of each frame type.

After coding, the frames are arranged in a deterministic periodic sequence, e.g. "IBBPBB" or "IBBPBBPBBPBB", which we called GOP pattern throughout the rest of the paper. These frames are then packetized into ATM cells.

As experimental video data, we use the *Star Wars* movie sequence [3]. The sequence consists of 174126 frames, which corresponds to about 2 hours of movie. The GOP pattern of this video is "IBBPBBPBBPBB". The size of the decoded video frames is 504 x 480 Pixels.

Frame type	**Number**	**Average** $[cells]$	**Min** $[cells]$	**Max** $[cells]$	**CoV**
all	**174126**	**41.12**	**2**	**483**	**1.15**
I	**14511**	**157.74**	**31**	**483**	**0.33**
P	**43531**	**60.58**	**6**	**454**	**0.63**
B	**116084**	**19.25**	**2**	**169**	**0.65**

Table 1: Statistical data of Star Wars sequence.

Table 1 shows some statistical data of the video stream assuming a payload of 48 octets per cell; CoV denotes the coefficient of variation. In addition to the distributions of the frame sizes, there are the following correlation properties of MPEG coded video streams [11]:

- Periodical dependences introduced by the coding algorithm due to the use of a certain GOP pattern,

- Strong positive correlations of the GOP size for the first tens of lags, where the GOP size denotes the sum of frame sizes of a GOP.
- Long-range dependence, i.e., the Hurst parameter of the GOP size sequence is larger than 0.5 .

The GOP pattern plays the most important role concerning autocorrelation effects of an MPEG video stream coded with different frame types, because it fixes the periodic nature of the stream. This unique property of MPEG coded videos prevents us from using video models which are based on statistical data from video sequences which have only one frame type or ignore the GOP structure.

Therefore, the idea behind our *basic model* is to describe the coder output process by a cyclic array of frame size distributions of the specific GOP pattern of the considered video sequence. From the *Star Wars* sequence, we therefore extract a sequence of 12 different distributions. The only frame-by-frame correlation information which is used in the *basic model* is the order of the frame size distributions fixed by the GOP pattern. In Section 5, we will extend this model to reflect GOP-by-GOP correlations. The effects of long-range dependence are not considered in this paper since our discrete-time approach is limited to cover correlations over a finite time range.

3 System model

For our analysis, we consider a one-stage queuing system with a single video traffic source. The buffer is of finite length and the service time is constant. Figure 1 depicts the system and provides the most important system parameters. For system analysis, we make the

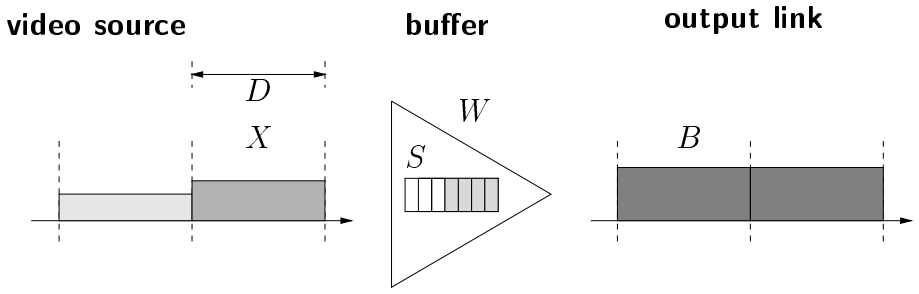

Figure 1: Buffer with a single video traffic source

following assumptions: The time is discretized into frame durations D, i.e., the reciprocal of the frame rate $r = 25s^{-1}$, and synchronized to frame starts.
All data is discretized into ATM cells carrying 48 octets of payload, i.e., the frame sizes,

the buffer content, and the amount of data transmitted during a frame duration.
Cells of a single frame are regarded as a fluid according to the *fluid simulation approach* [2]. Instead of individual cell arrivals we consider frame data as a fluid, which flows into the buffer at a constant rate. There are two important benefits from the fluid approach. It is conceptually simple and it leads to a reduced algorithmic complexity while the accuracy of the results is comparable to cell-oriented approaches. This method is applicable if the cell rate stays constant for a period of time which is considerably longer than the cell inter-arrival times. In our paper, this is clearly the case since the rate stays constant for one frame duration. Due to the constant inter-arrival time of the frames, this bit rate is equal to *frame size* · *frame rate*. Loss probabilities can be calculated in terms of overflow volumes. In case of video traffic, the fluid approach is also useful from practical view point. In [13], the authors report that smoothing the data cells of a video source to a constant cell rate during one frame duration outperforms all other cell emission schemes in terms of multiplexer performance.

4 Analysis without GOP correlations

First, we analyze the system with *basic model* input. The following notation is used to characterize the system:

X_n	:	random variable for the number of cells in the nth frame,
B	:	random variable for the number of cells transmitted during one frame duration D; in our case B is deterministic and we denote the constant number of cells transmitted by b,
W_n	:	random variable for the number of cells waiting in the buffer upon the arrival of the nth frame,
S	:	buffer size.

The random variables X_n, B, and W_n follow discrete distributions $x_n(k)$, $b(k)$ and $w_n(k)$, where, for instance, $x(i) = \Pr\{X_n = i\}$.

Due to our fluid-flow assumption, we are able to simplify the computation of the distribution of the buffer content $w_n(k)$ considerably as compared to standard unfinished work approaches such as, e.g., in [14]. The system evolution is determined by the following equation:

$$W_{n+1} \quad = \quad \min\left\{\max\{W_n + X_n - b, 0\}, S\right\}. \tag{1}$$

At first glance, our approach looks like a typical batch arrival process with batch size of X_n. In contrast to the original method, however, we already subtract the maximum amount of data which can be transmitted during a frame duration b upon arrival of frame

n. This has to be done due to the fluid-flow assumption and the discretization of the system time into frame durations. After the arrival of one frame as a batch of cells, we have to subsume in one equation the behavior of the modeled system until the next frame arrival, i.e., equally spaced cells entering a buffer which is served at a constant rate. Integrating both arrival and service process over D, and taking into account the content of the buffer before the frame arrival as well as the limited buffer size leads to Equation (1). For illustration, Figure 2 shows a snapshot of the system evolution containing the three possible cases. If the buffer content plus the new video data minus the amount of data which can be transmitted exceeds the buffer size then cells are lost (left bar). If more data can be transmitted than the buffer content plus arriving data then the buffer runs empty (center bar). In all other cases the buffer contains cells and there are no losses during the frame duration (right bar).

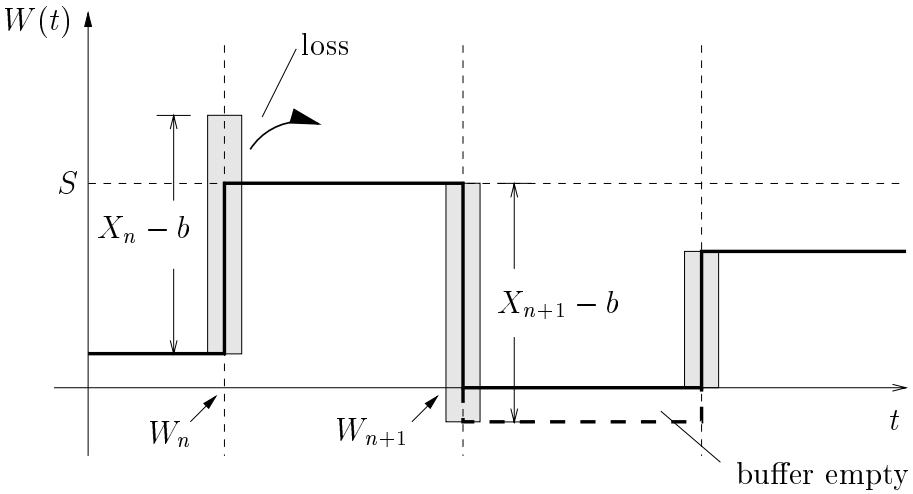

Figure 2: Snapshot of the system evolution

Equation (1) leads to the following recursion for the distribution of the buffer content:

$$w_{n+1}(k) \quad = \quad \pi^S \circ \pi_0 \left[w_n(k) \circledast x(k) \circledast \delta(k+b) \right]. \tag{2}$$

Here, $\circledast$ denotes the discrete convolution, π_0 and π^S are sweep operators, and $\delta(k+b)$ is the shifted Kronecker delta. They are defined as follows.

- *Discrete convolution.*

$$c(k) \quad = \quad a(k) \circledast b(k) \quad = \quad \sum_{j=-\infty}^{\infty} a(k-j) \cdot b(j). \tag{3}$$

- *Low sweep operator.*

$$\pi_0[a(i)] \quad = \quad \begin{cases} 0 & : \quad i < 0, \\ \sum\limits_{j=-\infty}^{0} a(j) & : \quad i = 0, \\ a(i) & : \quad i > 0. \end{cases} \tag{4}$$

- *High sweep operator.*

$$\pi^S[a(i)] \quad = \quad \begin{cases} a(i) & : \quad i < S, \\ \sum\limits_{j=S}^{\infty} a(j) & : \quad i = S, \\ 0 & : \quad i > S. \end{cases} \tag{5}$$

- *Kronecker delta.*

$$\delta(i) \quad = \quad \begin{cases} 1 & : \quad i = 0, \\ 0 & : \quad \text{otherwise.} \end{cases} \tag{6}$$

Figure 3 provides a pictorial description of Equation (2). In addition, Figure 4 shows a diagram of the computation of the buffer content distribution. This diagram constitutes the basic building block of the algorithms presented in the following.

Assuming that the buffer content distribution of the multiplexer converges to a steady state, $w(k) = \lim_{n\to\infty} w_n(k)$ is the solution of the fix-point equation

$$w(k) \quad = \quad \pi^S \circ \pi_0 \left[w(k) \circledast c(k) \right], \tag{7}$$

where

$$c(k) \quad = \quad x(k) \circledast \delta(k+b). \tag{8}$$

Equation (7) is the discrete-time analogue of the Lindley integral equation of $GI/GI/1-S$ queuing systems (see [7]).

In the following, we extend this basic algorithm to cope with arrivals which do not follow a single distribution but a cyclic sequence of distributions which reflects the GOP pattern of length N of MPEG video traffic streams. A single step of the iterative system description with the general distribution $x(k)$ is now replaced by a sequence of N sub-steps with the discrete distributions of the I-, P-, and B-frame sizes $x_j(k)$, $j = 1, \ldots, N$, where the order is determined by the GOP pattern of the empirical data set.

In case of stationarity, we obtain the following system of equations.

$$\begin{aligned} w_{j+1}(k) &= \pi^S \circ \pi_0 \left[w_j(k) \circledast c_j(k) \right] \quad j = 1, \ldots, N-1 \\ w_1(k) &= \pi^S \circ \pi_0 \left[w_{N+1}(k) \circledast c_N(k) \right], \end{aligned} \tag{9}$$

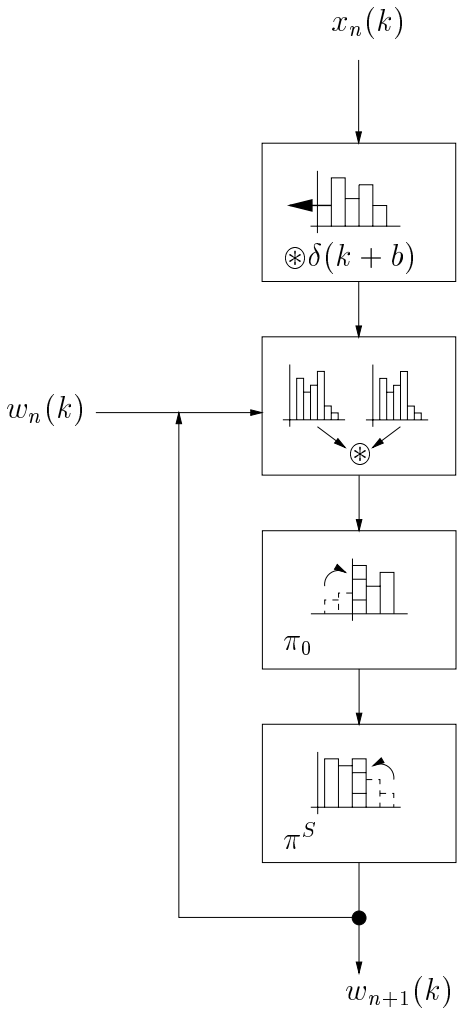

Figure 3: Block scheme of the basic algorithm

where $c_j(k) = x_j(k) \circledast \delta(k+b)$.

We determine the solution of this fix-point problem iteratively by applying the scheme shown in Figure 5. The following steps constitute the algorithm:

1. Initialize $w_1(k)$ with the distribution of an empty system,
   i.e., $w_1(0) = \delta(0)$.
2. Compute the system functions $c_j(k)$.
3. Apply convolution and sweep operators π_0 and π^S.
4. Repeat this operation for each system function (in total N times).
5. If the convergence criterion is not met, go to Step 3.

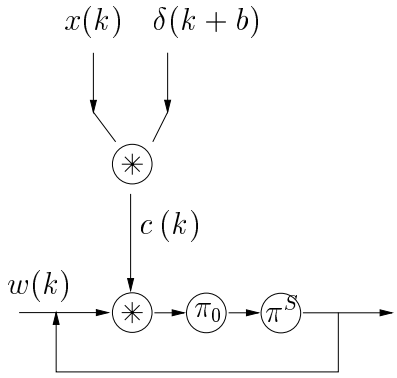

Figure 4: Computation of the buffer content distribution (fundamental algorithm)

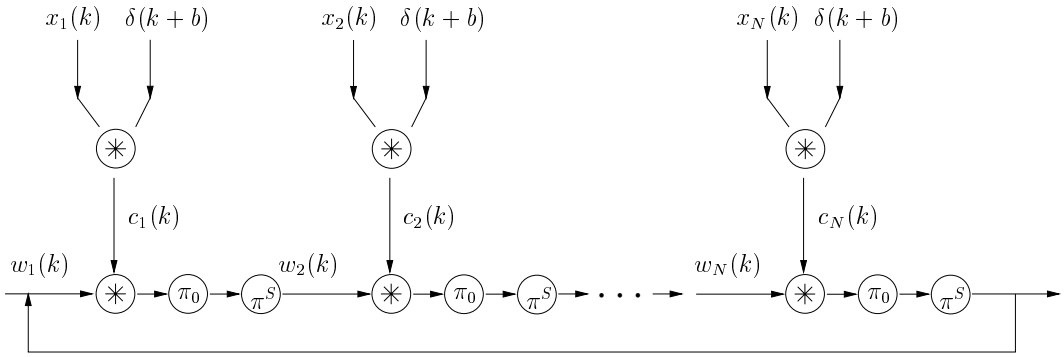

Figure 5: Computation of the buffer content distribution (basic model)

The buffer content distributions $w_j(k)$ are now used to compute the cell loss probability P_{loss}. Since we have to consider the whole GOP the loss probability is the average loss during a GOP duration divided by the average GOP size

$$P_{\text{loss}} \quad = \quad \frac{\mathrm{E}[L]}{\mathrm{E}[Y]} \quad = \quad \frac{\sum_{j=1}^{N} \mathrm{E}[L_j]}{\sum_{j=1}^{N} \mathrm{E}[X_j]}, \tag{10}$$

where L (L_j) denotes the amount of cells lost during one GOP (frame) duration, and Y (X_j) is the GOP (frame) size.

To determine the average amount $\mathrm{E}[L_j]$ of losses during a frame duration, we have to consider all cases where

$$k \quad > \quad (S-i)+b \qquad i=0,\ldots,S, \tag{11}$$

with k cells arriving at a buffer that contains i cells. In other words, cells are lost if

$$k \quad = \quad (S-i)+b+1,\ \ldots\ , x_{\max}, \tag{12}$$

where $x_{\max}$ denotes the maximum frame size of all frames.

The average amount of losses is then computed by

$$\mathrm{E}[L_j] \;=\; \sum_{i=0}^{S} w_j(i) \cdot \left[\sum_{l=1}^{x_{\max}-(S-i)-b} x_j\left((S-i)+b+l\right) \cdot l \right]. \tag{13}$$

5 Analysis with GOP-correlations

From the results of [12], we conclude that the *basic model* leads only to good cell loss predictions if the multiplexer buffers are small. For larger buffers, models including GOP-by-GOP correlations are assumed to outperform models without GOP correlations. Therefore, we extend the model of Section 2 to include GOP-by-GOP correlations for a small number of lags.

In addition to the notations introduced for the *basic model*, we use the following terms for the *extended model*:

X_{n_j}	:	random variable for the number of cells in the jth frame of the nth GOP following the distribution $x_j(k)$ with $j = 1, \ldots, N$.
Y_{n-h}	:	random variable for the class of GOP $n-h$, where GOP n is currently being processed by the algorithm.

To determine the GOP class of a given GOP, we first have to compute the histogram of the GOP sizes of the considered video sequence. Then, the GOP class is defined as the sequence number of the GOP size histogram interval in which this particular GOP falls. Hence, to obtain M GOP classes, a histogram with M intervals is used. GOP classes have to be introduced instead of actual GOP sizes to keep the model numerically tractable.

For the *extended model*, we replace the frame sizes X_n by $(h+1)$-dimensional vectors $(X_n, Y_{n-1}, \ldots, Y_{n-h})$, where

$$x(k, t_1, \ldots, t_h) \quad = \quad \Pr\{X = k, Y_{n-1} = t_1, \ldots, Y_{n-h} = t_h\}. \tag{14}$$

This means that the current frame size depends on the GOP sizes of one or more previous GOPs. For illustration see Figure 6.

To simplify the notation we define $\mathbf{t} = (t_1, \ldots, t_h)$ to represent the currently considered conditions, for instance, $x(k, t_1, \ldots, t_h) = x(k, \mathbf{t})$.

The conditional frame size distributions $x_j^{(\mathbf{t})}(k)$ are given by

$$x_j^{(\mathbf{t})}(k) \quad = \quad \frac{x_j(k, \mathbf{t})}{\sum_k x_j(k, \mathbf{t})}. \tag{15}$$

Let N denote the GOP length, M the number of GOP classes, and q the probability of the vector of conditions $\mathbf{t}$. Hence, we obtain a new set of equations which determine the

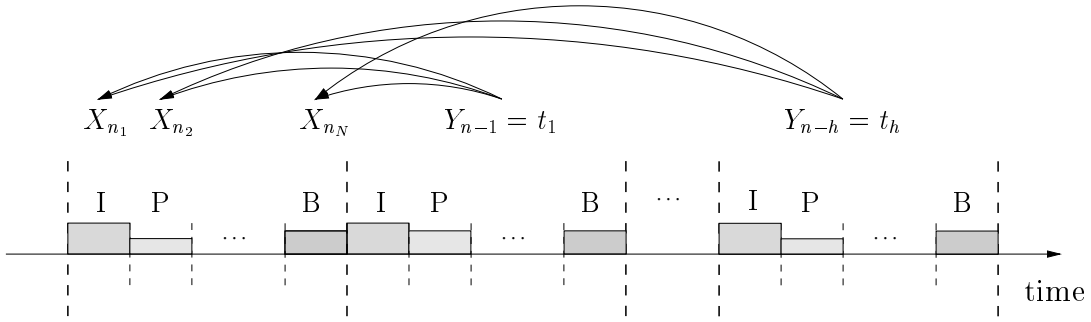

Figure 6: Conditional frame size probabilities

evolution of the buffer content.

$$\begin{aligned} w_{j+1}^{(\mathbf{t})}(k) &= \pi^S \circ \pi_0 \left[w_j^{(\mathbf{t})}(k) \circledast x_j^{(\mathbf{t})}(k) \circledast \delta(k+b) \right], \\ w_1^{(\mathbf{t})}(k) &= \sum_{\mathbf{t}} q \cdot w_{N+1}^{(\mathbf{t})}(k) \end{aligned} \tag{16}$$

for $j = 1, \ldots, N$ and $\mathbf{t} \in \{1, \ldots, M\}^h$.

The system of equations can be solved by an iteration as depicted in Figure 7. The operator in the hexagon is introduced to improve the readability.

Starting with a common buffer content distribution for all cases at the beginning of a GOP, we have to compute the iterations for each set of conditions $\mathbf{t}_i$ separately. Each of these sequences of iterations forms a row of the computing scheme. At the end of the GOP, all these distributions have to be aggregated to a single distribution by means of the weights q_i. For $h = 0$, Figure 7 reduces to Figure 5.

The cell loss probability has to be computed as a weighted sum of the loss probabilities on each path of Figure 7:

$$P_{\text{loss}} = \sum_{i=1}^{M^h} q_i \cdot p_{\text{loss}}^{(\mathbf{t}_i)} \tag{17}$$

with (see Equation (10))

$$\begin{aligned} p_{\text{loss}}^{(\mathbf{t}_i)} &= \\ &= \frac{\displaystyle\sum_{j=1}^{N} \left\{ \sum_{k=0}^{S} w_j^{(\mathbf{t}_i)}(k) \cdot \left[\sum_{l=1}^{x_{\max}-(S-k)-b} x_j\left((S-k)+b+l\right) \cdot l \right] \right\}}{\displaystyle\sum_{j=1}^{N} \mathrm{E}[X_j]}. \end{aligned} \tag{18}$$

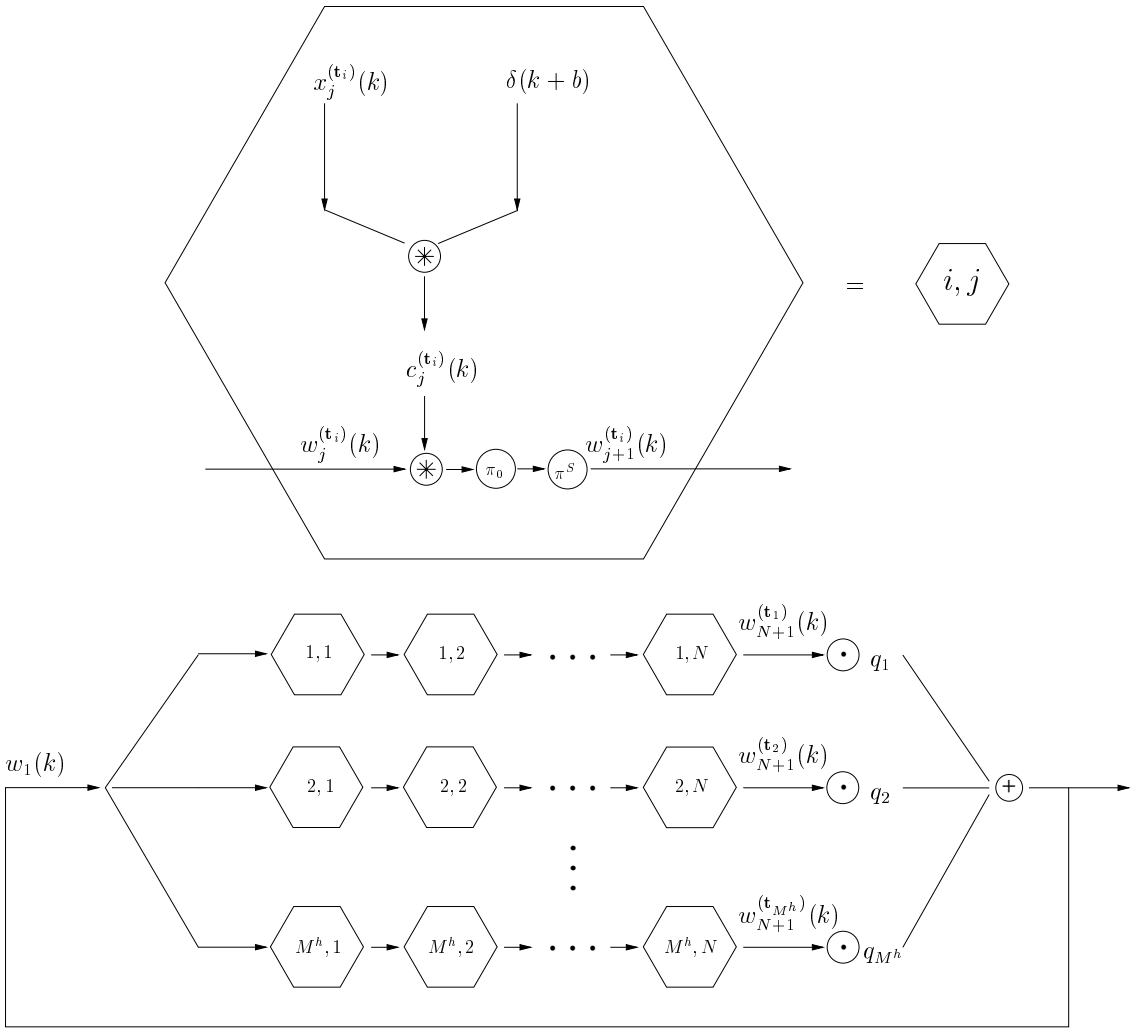

Figure 7: Computation of the buffer content distribution (extended model)

6 Numerical results

In this section, we present cell loss results for both the *basic model* and the *extended model* with $h = 1, 2, 3$. The frame size distributions were derived from the Bellcore *Star Wars* data set, i.e. $N = 12$. The number of GOP classes M is 5.

Figure 8 shows the cell loss results for a buffer size of 100 cells. The offered load ρ is given by $\rho = \mathrm{E}[X]/b$, where $\mathrm{E}[X]$ denotes the average frame size of the whole sequence. For buffers of this size and smaller, all model variants lead to approximately the same good prediction of the cell losses of the trace-driven simulation. Therefore, it is sufficient to use simple histogram-based models in the presence of buffers with a size in the order of the average frame size.

The situation is different for larger buffers. Here, as depicted in Figure 9 for a buffer size

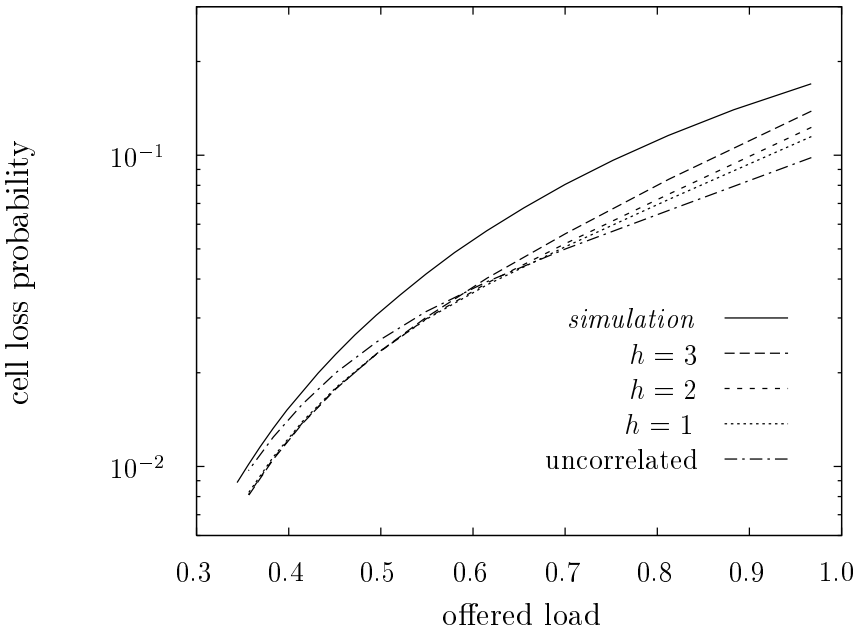

Figure 8: Cell loss curves for a buffer size of 100 cells

of 1000 cells, the modeling of the GOP-by-GOP correlation becomes important. This is reflected by the ranking of the model approximation quality, where the *extended model* with $h = 3$ is several orders of magnitude better than the *basic model*.

In Figure 10, we set the offered load to $\rho = 0.9$ and compute the cell loss probabilities for a wide range of buffer sizes. Only the model with $h = 3$ leads to acceptable estimates of the trace-based cell losses up to a buffer size of about 400 cells. The predicted losses of the other models are accurate up to a buffer size of 100 cells and underestimate the losses for larger buffer sizes.

7 Conclusion and Outlook

In this paper, we presented a discrete-time analysis of a buffer with VBR MPEG video traffic input. The developed video models consider the different frame size distributions, the cyclic GOP pattern, and the GOP-by-GOP correlation for a small number of lags. We provided versions of the analysis algorithm for both a basic video model without GOP-by-GOP correlations and an extended video model with GOP-by-GOP correlations.

The cell loss results indicate that up to a buffer size in the order of the average frame size it is sufficient to use the basic video traffic model which only reflects the cyclic GOP pattern. For larger buffers, GOP-by-GOP correlations also have to be taken into account. For instance, if the GOP-by-GOP correlation over three GOPs is considered the extended model estimates are accurate up to a buffer size of 400 cells compared to 100 cells for the

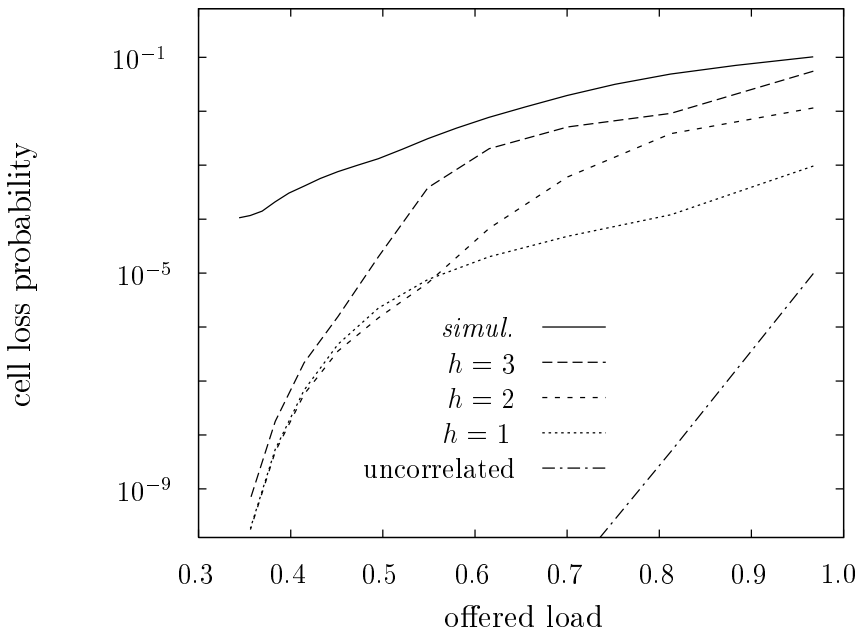

Figure 9: Cell loss curves for a buffer size of 1000 cells

basic model.

The implementation of the algorithm is simple but there are a number of limitations with respect to memory consumption and running time. Generally, the iteration takes longer to converge if the buffer size is increased or the offered load decreased. In addition, for the extended model the time to converge is considerably longer than for the basic model. Another difficulty that arises for the extended model is the memory requirements to store the conditional frame sizes. Even if we apply sophisticated storage techniques the correlation order which can be used for experiments is clearly limited since the number of probabilities grows exponentially with the size of GOP correlation interval.

To sum up, the paper shows that it is feasible to carry out an analytical performance evaluation of a buffer with MPEG video input if the system memory, or, in other words, the time range of the considered video traffic correlations does not become too large.

Currently, an extension of the algorithms to a superposition of video input sources is under development.

Acknowledgments

The author would like to thank Mark Garrett (Bellcore, Morristown, NJ) for providing the *Star Wars* data set, and Ulrike Schullerus for her programming efforts.

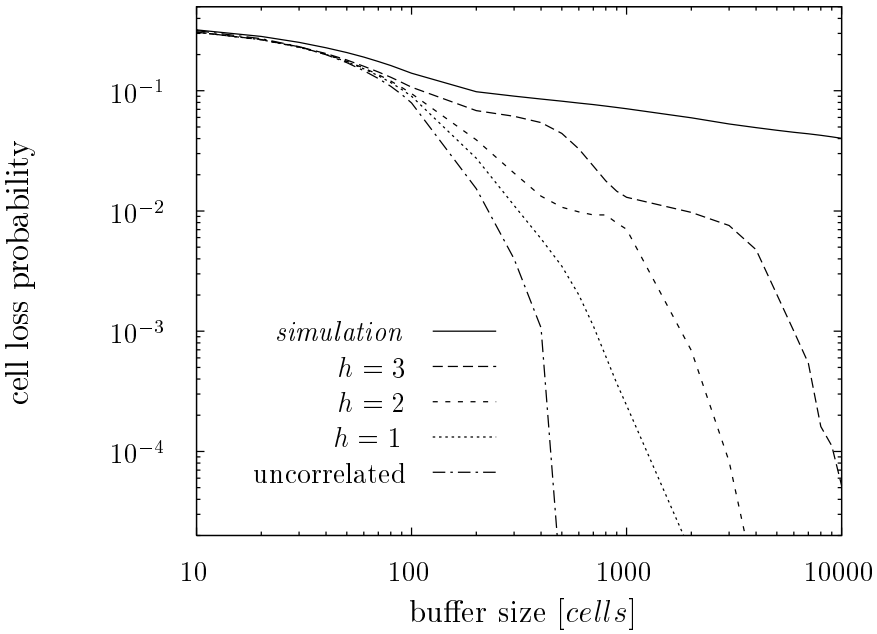

Figure 10: Cell loss curves for $\rho = 0.9$

References

[1] A. Adas. Supporting real time VBR video using dynamic reservation based on linear prediction. In *P*roceedings of the Infocom '96, pages 1476–1483, 1996.

[2] V. S. Frost and B. Melamed. Traffic modeling for telecommunication networks. *IEEE* Communications Magazine, 32(3):70–81, Mar. 1994.

[3] M. W. Garrett. *C*ontributions toward real-time services on packet switched networks. PhD thesis, Columbia University, 1993.

[4] M. W. Garrett and W. Willinger. Analysis, modeling and generation of self-similar VBR video traffic. In *P*roceedings of the ACM SIGCOMM '94 Conference, pages 269–280, 1994.

[5] ISO. *C*oding of Moving Pictures and Associated Audio for Digital Storage Media up to 1.5 Mbit/s – Part 2: Video. International Standard: ISO/IEC IS 11172-2, 1993.

[6] ISO. *G*eneric Coding of Moving Pictures and Associated Audio – Part 2: Video. International Standard: ISO/IEC IS 13818-2, 1994.

[7] L. Kleinrock. *Q*ueueing Systems – Volume 1: Theory. Wiley, New York, 1975.

[8] D. Le Gall. MPEG: A video compression standard for multimedia applications. *C*ommunications of the ACM, 34(4):46–58, Apr. 1991.

[9] D. M. Lucantoni, M. F. Neuts, and A. R. Reibman. Methods for performance evaluation of VBR video traffic models. *IEEE/ACM* Transactions on Networking, 2(2):176–180, Apr. 1994.

[10] D. Reininger, B. Melamed, and D. Raychaudhuri. Variable bit rate MPEG video: Characteristics, modeling and multiplexing. In *P*roceedings of the ITC-14, pages 295–306, 1994.

[11] O. Rose. Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems. In *P*roceedings of the 20th Annual Conference on Local Computer Networks, pages 397–406, Oct. 1995.

[12] O. Rose and M. Ritter. MPEG-video sources in ATM-systems – a new approach for the dimensioning of policing functions. In T. Hasegawa, G. Pujolle, H. Takagi, and Y. Takahashi, editors, *L*ocal and Metropolitan Communication Systems. Volume 3, pages 108–126. Chapman & Hall, London, 1995.

[13] P. Skelly, M. Schwarz, and S. Dixit. A histogram-based model for video traffic behavior in an ATM multiplexer. *I*EEE/ACM Transactions on Networking, 1(4):446–459, Aug. 1993.

[14] P. Tran-Gia and H. Ahmadi. Analysis of a discrete-time $G^{[X]}/D/1-S$ queueing system with applications in packet-switching systems. In *P*roceedings of the Infocom '88, pages 861–870, 1988.

Verantwortlich: Die Vorstände des Institutes für Informatik.

[100] U. Hertrampf, H. Vollmer und K.W. Wagner. *On the Power of Number-Theoretic Operations with Respect to Counting.* Januar 1995.

[101] O. Rose. *Statistical Properties of MPEG Video Traffic and their Impact on Traffic Modeling in ATM Systems.* Februar 1995.

[102] M. Mittler und R. Müller. *Moment Approximation in Product Form Queueing Networks.* Februar 1995.

[103] D. Rooß und K.W. Wagner. *On the Power of Bio-Computers.* Februar 1995.

[104] N. Gerlich und M. Tangemann. *Towards a Channel Allocation Scheme for SDMA-based Mobile Communication Systems.* Februar 1995.

[105] A. Schömig und M. Kahnt. *Vergleich zweier Analysemethoden zur Leistungsbewertung von Kanban Systemen.* Februar 1995.

[106] M. Mittler, M. Purm und O. Gihr. *Set Management: Synchronization of Prefabricated Parts before Assembly.* März 1995.

[107] A. Schömig und M. Mittler. *Autocorrelation of Cycle Times in Semiconductor Manufacturing Systems.* März 1995.

[108] A. Schömig und M. Kahnt. *Performance Modelling of Pull Manufacturing Systems with Batch Servers and Assembly-like Structure.* März 1995.

[109] M. Mittler, N. Gerlich und A. Schömig. *Reducing the Variance of Cycle Times in Semiconductor Manufacturing Systems.* April 1995.

[110] A. Schömig und M. Kahnt. *A note on the Application of Marie's Method for Queueing Networks with Batch Servers.* April 1995.

[111] F. Puppe, M. Daniel und G. Seidel. *Qualifizierende Arbeitsgestaltung mit tutoriellen Expertensystemen für technische Diagnoseaufgaben.* April 1995.

[112] G. Buntrock, und G. Niemann. *Weak Growing Context-Sensitive Grammars.* Mai 1995.

[113] J. García and M. Ritter. *Determination of Traffic Parameters for VPs Carrying Delay-Sensitive Traffic.* Mai 1995.

[114] M. Ritter. *Steady-State Analysis of the Rate-Based Congestion Control Mechanism for ABR Services in ATM Networks.* Mai 1995.

[115] H. Graefe. *Konzepte für ein zuverlässiges Message-Passing-System auf der Basis von UDP.* Mai 1995.

[116] A. Schömig und H. Rau. *A Petri Net Approach for the Performance Analysis of Business Processes.* Mai 1995.

[117] K. Verbarg. *Approximate Center Points in Dense Point Sets.* Mai 1995.

[118] K. Tutschku. *Recurrent Multilayer Perceptrons for Identification and Control: The Road to Applications.* Juni 1995.

[119] U. Rhein-Desel. *Eine "'Übersicht"' über medizinische Informationssysteme: Krankenhausinformationssysteme, Patientenaktensysteme und Kritiksysteme.* Juli 1995.

[120] O. Rose. *Simple and Efficient Models for Variable Bit Rate MPEG Video Traffic.* Juli 1995.

[121] A. Schömig. *On Transfer Blocking and Minimal Blocking in Serial Manufacturing Systems — The Impact of Buffer Allocation.* Juli 1995.

[122] Th. Fritsch, K. Tutschku und K. Leibnitz. *Field Strength Prediction by Ray-Tracing for Adaptive Base Station Positioning in Mobile Communication Networks.* August 1995.

[123] R.V. Book, H. Vollmer und K.W. Wagner. *On Type-2 Probabilistic Quantifiers.* August 1995.

[124] M. Mittler, N. Gerlich, A. Schömig. *On Cycle Times and Interdeparture Times in Semiconductor Manufacturing.* September 1995.

[125] J. Wolff von Gudenberg. *Hardware Support for Interval Arithmetic - Extended Version.* Oktober 1995.

[126] M. Mittler, T. Ono-Tesfaye, A. Schömig. *On the Approximation of Higher Moments in Open and Closed Fork/Join Primitives with Limited Buffers.* November 1995.

[127] M. Mittler, C. Kern. *Discrete-Time Approximation of the Machine Repairman Model with Generally Distributed Failure, Repair, and Walking Times.* November 1995.

[128] N. Gerlich. *A Toolkit of Octave Functions for Discrete-Time Analysis of Queuing Systems.* Dezember 1995.

[129] M. Ritter. *Network Buffer Requirements of the Rate-Based Control Mechanism for ABR Services.* Dezember 1995.

[130] M. Wolfrath. *Results on Fat Objects with a Low Intersection Proportion.* Dezember 1995.

[131] S.O. Krumke and J. Valenta. *Finding Tree–2–Spanners.* Dezember 1995.

[132] U. Hafner. *Asymmetric Coding in (m)-WFA Image Compression.* Dezember 1995.

[133] M. Ritter. *Analysis of a Rate-Based Control Policy with Delayed Feedback and Variable Bandwidth Availability.* January 1996.

[134] K. Tutschku and K. Leibnitz. *Fast Ray-Tracing for Field Strength Prediction in Cellular Mobile Network Planning.* January 1996.

[135] K. Verbarg and A. Hensel. *Hierarchical Motion Planning Using a Spatial Index.* January 1996.

[136] Y. Luo. *Distributed Implementation of PROLOG on Workstation Clusters.* February 1996.

[137] O. Rose. *Estimation of the Hurst Parameter of Long-Range Dependent Time Series.* February 1996.

[138] J. Albert, F. R"ather, K. Patzner, J. Schoof, J. Zimmer. *Concepts For Optimizing Sinter Processes Using Evolutionary Algorithms.* February 1996.

[139] O. Karch. *A Sharper Complexity Bound for the Robot Localization Problem.* June 1996.

[140] H. Vollmer. *A Note on the Power of Quasipolynomial Size Circuits.* June 1996.

[141] M. Mittler. *Two-Moment Analysis of Alternative Tool Models with Random Breakdowns.* July 1996.

[142] P. Tran-Gia, M. Mandjes. *Modeling of customer retrial phenomenon in cellular mobile networks.* July 1996.

[143] P. Tran-Gia, N. Gerlich. *Impact of Customer Clustering on Mobile Network Performance.* July 1996.

[144] M. Mandjes, K. Tutschku. *Efficient call handling procedures in cellular mobile networks.* July 1996.

[145] N. Gerlich, P. Tran-Gia, K. Elsayed. *Performance Analysis of Link Carrying Capacity in CDMA Systems.* July 1996.

[146] K. Leibnitz, K. Tutschku, U. Rothaug. *K"unstliche Neuronale Netze f"ur die Wegoptimierung in ATG Leiterplattentestern.* Juli 1996.

[147] M. Ritter. *Congestion Detection Methods and their Impact on the Performance of the ABR Flow Control Mechanism.* August 1996.

[148] H. Baier Saip, K.W. Wagner. *The Analytic Polynomial Time Hierarchy.* September 1996.

[149] H. Vollmer, K.W. Wagner. *Measure One Results in Computational Complexity Theory.* September 1996.

[150] O. Rose. *Discrete-time Analysis of a Finite Buffer with VBR MPEG Video Traffic Input.* September 1996.