VBR VIDEO SOURCE CHARACTERIZATION AND A PRACTICAL HIERARCHICAL MODEL

István Cselényi [a], Sándor Molnár [b]

[a]Department of Network Services, Telia Research AB

Vitsandsgatan 9 B, S-123 86, Farsta, Sweden, E-mail: istvan.i.cselenyi@telia.se

[b]High Speed Networks Laboratory, Dept. of Telecommunications and Telematics,

Budapest University of Technology and Economics

Pázmány Péter sétány 1/D, H-1117, Budapest, Hungary, E-mail: molnar@ttt-atm.ttt.bme.hu

Abstract

There are many ways to build up traffic models for VBR video sources. A frequently applied methodology is to use mathematical analysis based on realistic assumptions to set up a source model that generates traffic according to a stochastic process. In this case, the critical issue is the validation of the synthetic trace by comparing statistics to results obtained from measurements on the real source.

In this paper, we choose a different and more practical approach to model the behavior of the real traffic source. Our model building philosophy is that we *analyze and understand* what happens with the video information on its way from the ingress to the multimedia terminal to the egress of the network card. Throughout this journey the information is processed by several mechanisms and we build an empirical model step by step based on our measurement-based observations.

Besides understanding the traffic generation procedure, statistical analysis of VBR traffic traces captured from a number of video sequences was also carried out in several scenarios. Using the knowledge of encoding, encapsulation and scheduling processes and results of the trace analysis, a *hierarchical source model* is set up for modeling the multimedia terminal. Thereby our model imitates the generation of video frames and the inner working of each level of protocol hierarchy and tries to reproduce the complex behavior of the real source. We use the *leaky bucket analysis* for verification of the model in order to capture directly the behavior of the traffic in a queue.

Keywords: ATM traffic characterization, VBR video source model, hierarchical modeling, white box modeling, IP over ATM

1. Introduction

Variable Bit Rate video has an important role in broadband Internet, because a substantial portion of forecast traffic is produced by multimedia sources (e.g. teleconferencing terminals, video-on-demand servers, etc.). The characteristics of VBR video traffic based on measurement studies and the related networking problems have been of increasing interest in the last decade of teletraffic research [8,11,14,19,22].

Source models of VBR video are needed to dimension networks and control methods to achieve acceptable quality and optimal usage of network resources [4,8]. A number of different models have been proposed for VBR video modeling (for an overview see [9,25]). The large variety of modeling approaches can be divided into the following three categories: Markov models [1,17], autoregressive processes [10,23] and fractal models [18,28]. However, all common to these approaches that a specific stochastic process is chosen and the parameters are set by a specific method to fit some statistics of the real source. This methodology can be regarded as a *black box approach*, which is based merely on the characteristics of measured traffic. An alternative method is the so-called *white box approach*, which attempts to reproduce the detailed behavior of the source by imitating its inner working [12]. This methodology has received little attention from the video modeling community so far. However, we believe that this modeling concept can be very successful in practice, because it can capture the impact of encoding and encapsulation procedures on the generated data traffic. Therefore we combine the white and black box modeling approaches in this work.

The main contribution of our paper is twofold. On one hand we give a comprehensive *source analysis study* and reveal the very nature of measured video traffic by performing traffic intensity analysis, correlation structure analysis, silence period analysis on several burst levels ranging from packet to scene level. This black box analysis is followed by a white box analysis that is aimed at detecting the internal behavior of the multimedia terminal. On the other hand, we introduce a practical, *hierarchical, video source model*, which is built up using the observations and parameters identified during the black and white box analysis. We describe a simple technique to model the encoding and scheduling of video frames and emulate the impact of layers in the protocol stack on the ATM traffic.

The main advantages of our proposed model building technique compared to previous models are the following:

- We map the results of black box traffic analysis to our knowledge of the traffic generation procedure. This white box approach yields a model whose behavior is very close to that of the real source.
- We present a relatively simple algorithm for traffic generation where the parameters can be easily set based on measurements.
- The model targets to capture directly the queuing behavior (i.e. leaky bucket or burstiness curve, queue length and cell loss) of the real source. We avoid the complexity of fitting different statistical characteristics and investigating a rather complex queuing model.
- The modeling concept is verified by comparing the queuing performance of the synthetic and captured traffic traces. We restrict the use of statistical assumptions about the traffic and set our model parameters directly from measurements.

This paper is organized as follows. The measurement scenario, investigated multimedia platform and video sequences are introduced in Section 2. We explore the VBR source from outside as a black box by analyzing its traffic at different time scales in Section 3. Than we look inside the box (i.e. white box approach) and identify the units which have significant impact on the resultant traffic in Section 4. We construct a hierarchical traffic generation model from the black and white box analysis in Section 5. The verification and application of this model are given in Section 6. Finally, conclusions are summarized in Section 7.

2. Traffic Measurement

2.1 Measurement Scenario

A videocassette recorder provided video and audio signals in order to have a repeatable measurement procedure. The optical signal from the multimedia server was copied by means of an optical splitter to ATM test equipment without affecting the behavior of the application in use. The interarrival time of ATM cells of interest was recorded in real-time by a module developed by Telia AB, Sweden. It resides in an ATM test instrument developed in the RACE PARASOL project [3]. The traffic records were post-processed with software developed by the authors.

Sun SPARC 10 workstations equipped with video, audio hardware and ATM interface cards were used as end stations. Permanent virtual connections were established among them with classical IP over ATM protocol stack. IP datagram were encapsulated using IEEE 802.2 LLC/SNAP and segmented into ATM cells using AAL5. No shaping was applied to the ATM cell stream which was carried over STM-1 SDH physical layer. The video and audio information were multiplexed on a single Virtual Channel. The workstation platform, desktop multimedia application, video card and coding scheme we used in our investigations are frequently used as a multimedia terminal of reasonable price. This scenario is adequate to traffic characterization of multimedia sources in case of distributed multimedia applications such as videoconference, movie teleswitching or video on demand. Evaluation of measurements for different platforms is in progress.

2.2 Video Sequences

Several standard CCIR video sequences recorded in VHS quality were used as video source [5,6]. Three of them were selected for this paper:

- *Girl With Toys* A sequence of almost still pictures – no motion dynamics
- *Sussie* Head and shoulder scene, no camera movements – low motion dynamics
- *Popple* Zoom of moving object – high motion dynamics

The video sequences were coded using CellB compression and segmented by the desktop multimedia application so that a constant rate of frames of varying sizes was generated. The CellB compression algorithm is a low-cost video compression scheme used mostly on Sun platform [26]. In comparison with standard video compression algorithms such as MPEG or H.261, it has the advantage of cheap hardware support and less variable output video rate. The audio information was coded and transferred in a stream of 64 Kbps constant bit rate. The cases examined in this paper are given in Table 1. The last column of this table presents the long-term average rate, which is denoted by R.

Notation	Name of Video Sequence	Frame Rate [frame/sec]	Resolution [pixels]	Length of Trace [sec]	Average Rate [Kbps]
GT10	Girl with Toys	10	384x288	21.41	452.93
SU10	Sussie	10	384x288	37.17	582.75
PL10	Popple	10	384x288	24.04	1500.01
GT25	Girl with Toys	25	384x288	35.89	922.26
SU25	Sussie	25	384x288	34.55	1026.76
PL25	Popple	25	384x288	35.19	2251.11

Table 1 Parameters of the Measured Video Sequences

2.3 Traditional Analysis of Video Sequences

The Squared Coefficient of Variation (SCV) of Cell Interarrival Times (CITs) is used as the burstiness measure. This descriptor and other traditional traffic source statistics are listed in Table 2. The values confirm that *the burstiness of the traffic is determined by the video content and it is independent of the frame rate*. Thus the GT10-GT25 and SU10-SU25 sequences have almost the same burstiness. However, the burstiness of PL25 is considerably less than PL10's; probably due to the saturation of terminal performance. It can be seen from the peak cell rate values that no shaping was applied.

Name of Stream	CIT mean [cell time]	CIT variance	Burstiness	Peak Cell Rate [cell/sec]	Number of captured cells
GT10	342.89	9 077 900	77.21	366 792	22 871
SU10	266.59	6 997 300	98.52	366 792	51 089
PL10	103.54	2 309 000	215.38	366 792	85 060
GT25	168.40	2 235 300	78.82	366 792	78 066
SU25	151.20	2 115 300	92.53	366 792	83 696
PL25	68.99	818 380	171.93	366 792	186 827

Table 2 Traffic Characteristics of the Captured Cell Streams

3. Source Analysis

In this section we consider the VBR source as a black box and analyze the ATM cell stream produced by an unknown mechanism. Different methods are applied on the measured data at different time scales ranging from cell level to scene level. As a result of the detailed analysis, we make some hypothesis about the traffic generation procedure inside of the black box and obtain several traffic parameters that are used in our model.

3.1 Traffic Intensity Analysis

The usual way of traffic characterization is to measure its intensity, i.e. the amount of cell arrivals within a given time slot. The traffic intensity of recorded ATM traffic traces is analyzed at different time scales, in order to investigate the burst structure. Figure 1-4 show traffic intensity of a short trace from the PL25 video sequence on different burst levels. Each column represents the number of arrivals in one time slot of 58330, 750, 38 and one cell times in Figure 1,2,3 and 4, respectively.

The complete PL25 sequence is shown in Figure 1. The two level shifts are caused by two high speed zoom periods with an intermediate period of partial motion in the picture field. Figure 2 magnifies the next time scale (i.e. *video frame level*) and shows the arrival pattern of seven video frames of varying size with three audio packets in between. The internal structure of a frame (i.e. *IP packet level*) is shown in Figure 3. Finally, we show a single packet in Figure 4 that contains 172 ATM cells arriving at practically full link rate.

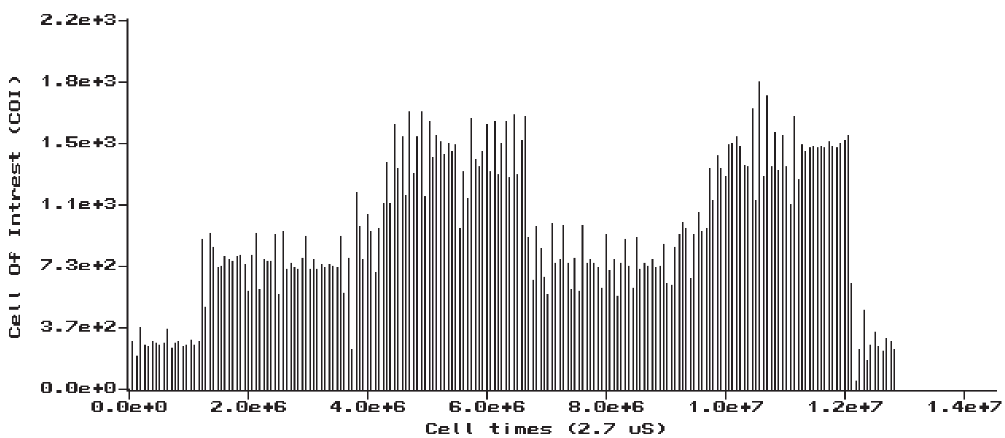


Figure 1 The Cell Arrival Intensity of The PL25 Sequence on Scene Level

Length of Window: 38 sec, Mean Rate in The Window: 2251 Kbps, Size of Time Slot: 58330 cell times

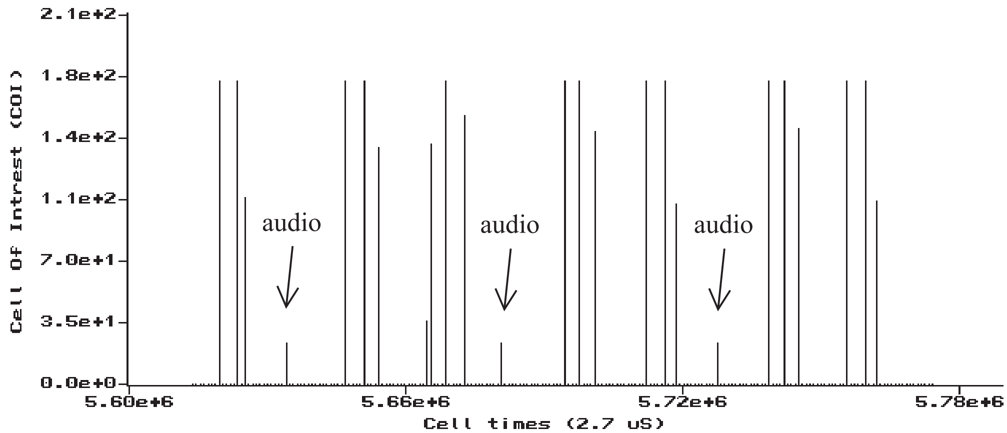


Figure 2 The Cell Arrival Intensity of The PL25 Sequence on Video Frame Level

Length of Window: 490 ms, Mean Rate in The Window: 3.26 Mbps, Size of Time Slot: 750 cell times

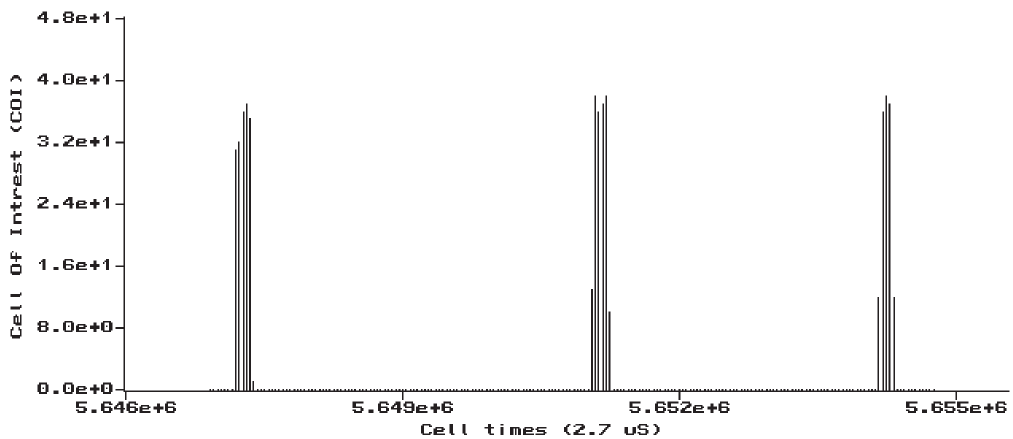


Figure 3 The Cell Arrival Intensity of The PL25 Sequence on IP Packet Level

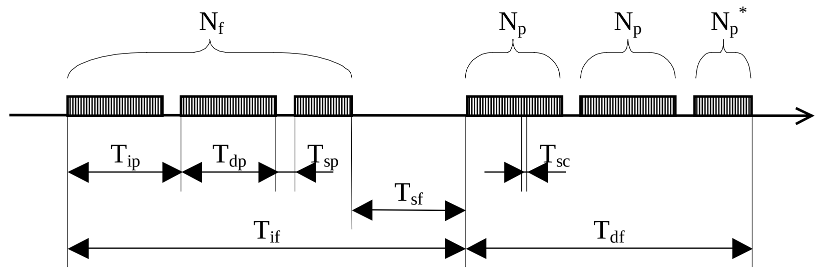
Length of Window: 24.5 ms, Mean Rate in The Window: 9.45 Mbps, Size of Time Slot: 38 cell times

Figure 4 The Cell Arrival Intensity of The PL25 Sequence on Cell Level

Length of Window: 654 μs, Mean Rate in The Window: 133.4 Mbps, Size of Time Slot: 1 cell time

Based on the traffic intensity analysis the multi-level burst structure of the examined VBR traffic is well pronounced, as it is shown in other works [5,14,24]. Observing these figures noteworthy is the regular arrival of frames, packets and cells at different time scales. Audio packets also arrive in a regular manner as it is illustrated in Figure 3. By plotting similar figures for the other video sequences we can conclude that the size of video frames depend on the content of video sequence while the structure of frame internal packets looks very much the same for each frame.

Based on these observations our hypothesis is that *the generation of frames and packets is independent thus these burst levels can be distinguished in our model*. We introduce the following notation for describing the multilevel burst structure:



N_p	packet size	N_f	frame size
T_{ip}	packet interarrival time	T_{if}	frame interarrival time
T_{dp}	packet duration	T_{df}	frame duration
T_{sp}	packet silence period	T_{sf}	frame silence period
N_{paudio}	size of audio packet	T_{sc}	cell silence period

Figure 5 Multilevel Bursts in the Traffic Stream

Figure 5 depicts the interarrival time and duration of frames (T_{if}, T_{df}) and packets (T_{ip}, T_{dp}) and the silence periods between frames and between packets (T_{sf} and T_{sp}), respectively. N_f and N_p denote the number of ATM cells in a frame and a packet, respectively. N_p^* denotes the size of the last packet in a frame, which is usually shorter than other packets (it contains the last fragment of the frame), while N_{paudio} refers to the packet containing an audio transfer unit (which is much smaller than a video packet). For the sake of simplicity we use the same notation for audio traffic, although “frame” does not exist in that case.

In the rest of this chapter, we further analyze the VBR traffic and try to quantify these parameters, which will contribute to our traffic model.

3.2 Silence Period Analysis

Beside the traffic intensity analysis (i.e. analysis of busy periods) the other native way of gaining information about the traffic is to calculate the probability mass function (PMF) of the Cell Interarrival Times (CIT). This latter method can be considered as analysis of the silence periods (T_{sf}, T_{sp} and T_{sc}).

The probabilities are estimated by counting the occurrence of CITs of different lengths in the captured trace. The values are smoothed by a moving average technique before drawing figures. The PMF of CITs is depicted in Figure 6 and the complementary probability density function (CPDF) is presented in Figure 7.

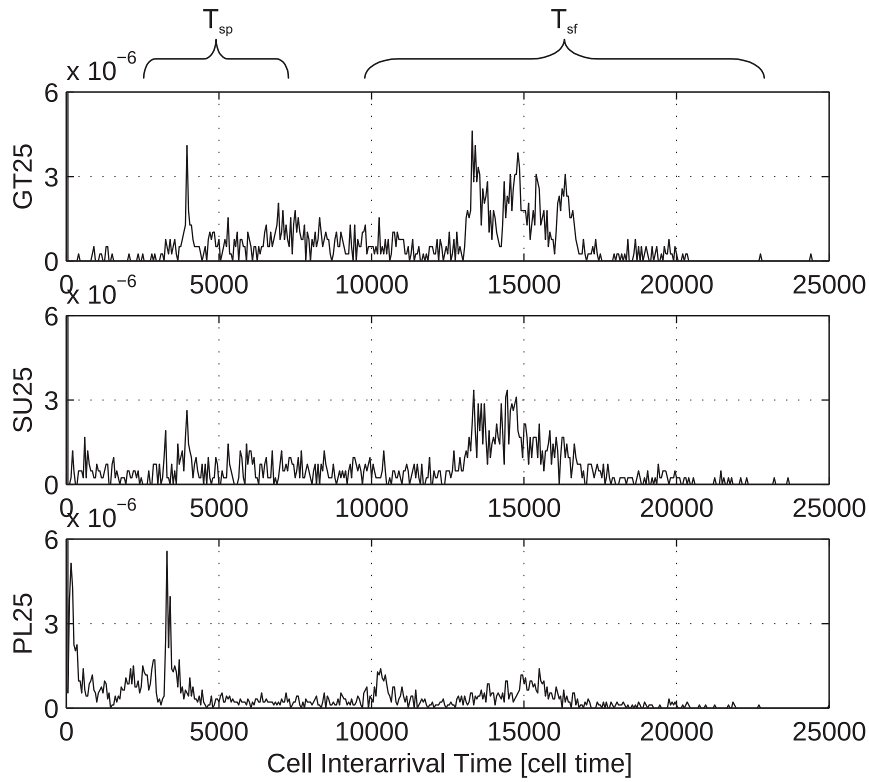


Figure 6 **Probability Mass Function of Cell Interarrival Times (GT25, SU25, PL25)**

The silence periods can be divided into three groups according to Figure 6, which characterizes the GT25, SU25 and PL25 sequences. The longest interarrival times (above 8000 cell times) represent the frame silence periods (T_{sf}). The medium values (around 4000 cell times) correspond to the silence periods within the video frames, i.e. between consecutive packets (T_{sp}) while the smallest values (below 10 cell times) express the short silent periods inside the packets (T_{sc}). The evaluation of each group is given below.

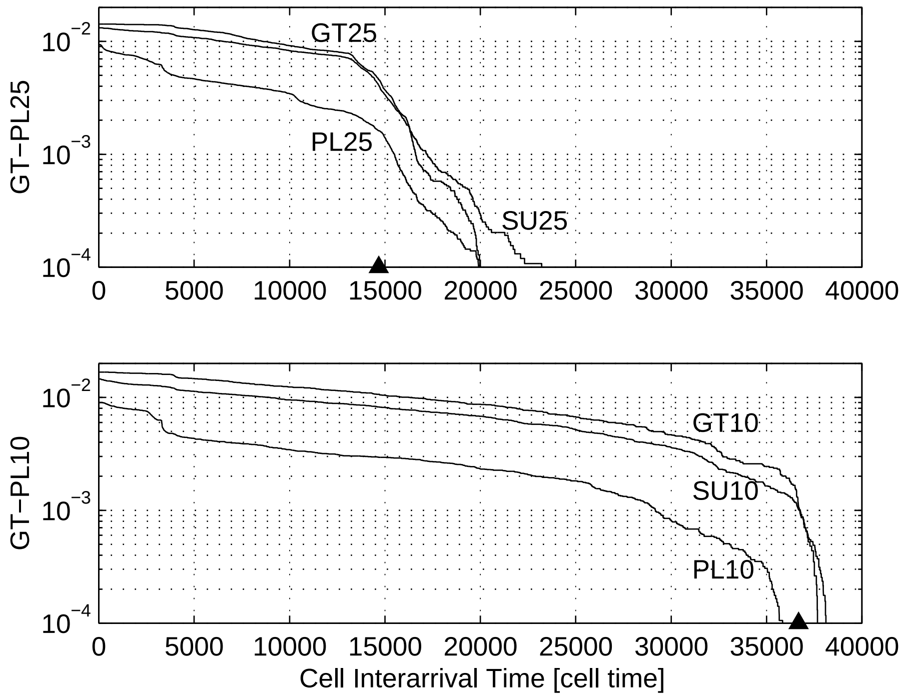


Figure 7 Complementary Distribution Function of Cell Interarrival Times (GT10, SU10, PL10, GT25, SU25, and PL25)

3.2.1 Frame Level

In case of GT-PL10 sequences, the video frame rate is 10 frame/sec thus the theoretical frame interarrival time (T_{if}) is around 36 679 cell time. For the GT-PL25 sequences with 25 frame/sec, the theoretical frame interarrival time (T_{if}) is 14 672 cell time. Two arrows in Figure 8 indicate these values. Theoretically, the silence period between frames can not be longer than the frame interarrival time (since $T_{sf} = T_{if} - T_{df}$ from Figure 5). However, it is clearly shown in Figure 7 that there are several silence periods longer than 36 679 and 14 672 cell times for the GT-PL10 sequences and GT-PL25 sequences, respectively. That is in terms of frame rate, *the investigated terminal platform is not able to produce frames at the theoretical rate* (i.e. 10 and 25 fps).

Moreover, the moderate declination of the CPDF curves in Figure 7 indicates that the frame generation time and duration varies. Another phenomenon to be noticed in Figure 7 is that the maximum silence period (i.e. T_{sfmax}) is significantly shorter for the PL sequences than for the others, probably due to the higher traffic intensity and larger size of video frames.

3.2.2 Packet Level

In case of SU25 sequence, the probability of normal packet silence period (3000 < T_{sp} < 5000 cell time) is less than in the case of PL25 (see Figure 6). The reason is that there are more frames, which consist of more than one packet in case of the more intensive PL25 sequence, thus there are more intra-frame packet silence periods in the captured cell stream. *The mean packet silence period read from Figure 6 is around 4000 cell time for SU25, GT25 and around 3400 cell time for PL25.* The packet silence period can be recognized also in Figure 7 in form of a sudden declination on the CPDF curves. By comparing the beginning of the CPDF curves in the CIT range of 2000-5000 cell time it is seen that the packet silence period is the same for the GT-PL10 and GT-PL25 sequences. Thus another hypothesis is that *the frame rate setting* (i.e. 10 fps or 25 fps in our paper) *has no impact on the packet generation.*

3.2.3 Cell Level

Figure 6 highlights that the probability of silence periods with length shorter then three cell time are high (see the peak at the left part of the graph). The reason is that most of the cells are transmitted back-to-back in the unshaped traffic stream. Although physical layer information was discarded by the measurement instrument, the traces of SDH overhead information can be observed too in the form of discontinuities within the packets, resulting in silence periods of two – three cell times.

3.3 Packet Level Analysis

After characterizing the silence periods on the three burst levels, we can filter out the shortest CITs from the captured traffic trace – which represent the inter packet silence periods (T_{sc}) – and identify the packets. This section analyses the burst size, maximum silent period and interarrival time parameters on the packet level (i.e. N_p, N_{paudio}, N_p^*, T_{sp} and T_{ip}).

3.3.1 Packet Size

First we analyze the probability mass function of the size of packets in the different traces (see Figure 8).

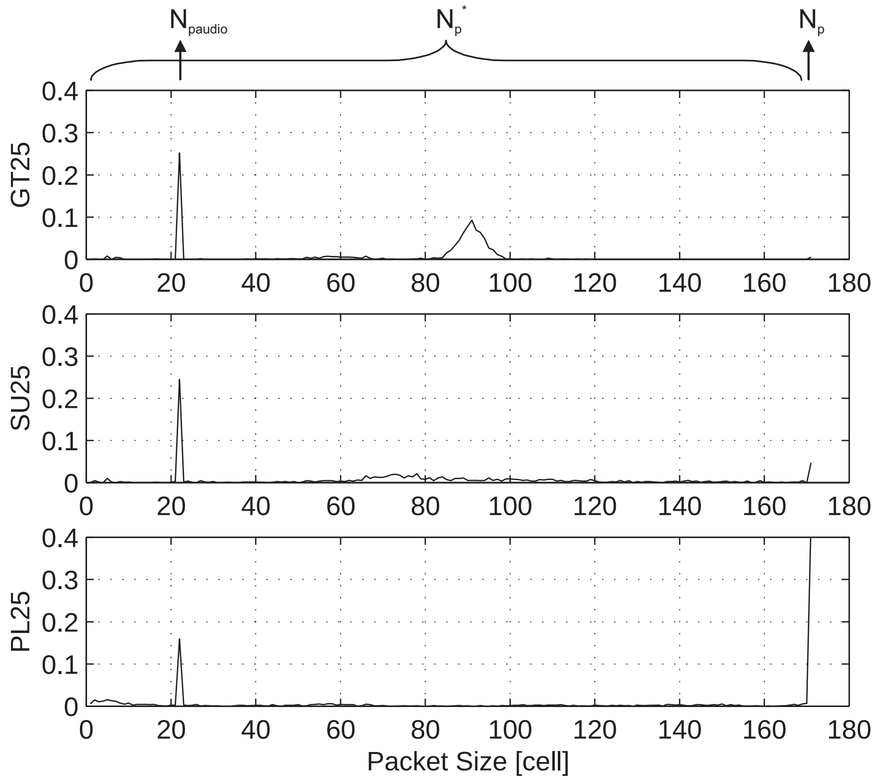


Figure 8 **Probability Mass Function of Packet Size (GT25, SU25, and PL25)**

It is seen from this figure, that the full packet size (N_p) is 172 cells and the size of audio packets is 23 cells. *By comparing the graphs of the three sequences one can see that the video content has a pronounced impact on the distribution of packet size.* Most of the video packets in the PL25 sequence are full-size (N_p) and the probability of the occurrence of shorter (i.e. not full-size) packets is equally small. The reason of this phenomenon is that this sequence contains intensive motion that results in large video frames, and fills up several full-size packets and one additional packet of varying size. The size of packets in the SU25 video sequence varies more around the mean packet size, but there are still many full-size packets. This is not the case for the GT25 sequence, where middle-size packets ($80 < N_p < 100$ cell time) have much higher probability than full-size packets.

3.3.2 Packet Scheduling

The relationship between the packet silence period (T_{sp}) and the packet size (N_p) is analyzed in this study on the packet level. One can observe in Figures 2 and 3, that the packet silence period is shorter for the last packet in the frame. Thus our next hypothesis is *that the multimedia terminal can produce a shorter packet faster than a larger one*. We can confirm this assumption by investigating the relationship between the length of the *k*th silence period (denoted by T_{sp}^{k}) and the size of the packet generated right after that period (denoted by N_p^{k}). Figure 9 presents the relationship between these factors and the empirical distribution of the packet silence period for the PL25 sequence.

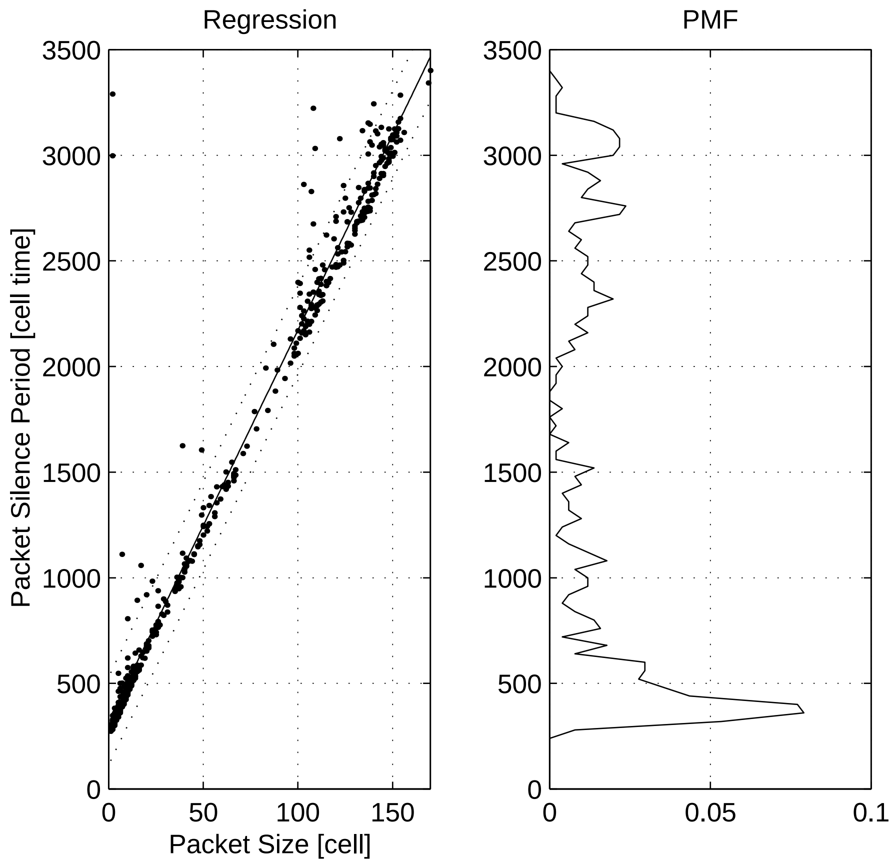


Figure 9 Regression of Packet Size and Packet Silence Period (PL25)

It is very pronounced that there is a linear relationship between these factors, therefore we establish a linear approximation:

$$T_{sp}^{k} = \alpha N_{p}^{k+1} + \beta \qquad (4\text{-}1)$$

The α and β constants can be determined by regression from the corresponding T_{sp} and N_p value pairs. In our case, the value of α is 18,5 and β is 323 and at least 50 % of the predictions are contained by the +/- 200 cell times wide environment of the regression line (see dotted lines in the figure). We have got very similar values for the other sequences, proving that *the packet level statistics are independent from the video content*. Related works [20,24] confirm this observation and equation (4-1) for TCP/IP over ATM traffic.

3.4 Frame Level Analysis

The frame level behavior of the traffic is analyzed in this subsection. The counts of cells in video frames (N_f) and the frame interarrival time (T_{if}) are derived from the captured cell streams by software analysis. Consecutive frames were distinguished using the maximum frame silence periods (T_{sf}) determined by the silence period analysis.

3.4.1 Frame Size

The audio packets, which have minor effect on the characteristics, get high importance here, since their presence had to be neglected during frame separation. An audio packet consists of 23 cells (see previous section); therefore the filtering software neglects every packet of 23 cells. However, the cells of audio packets are included in the calculation of frame size.

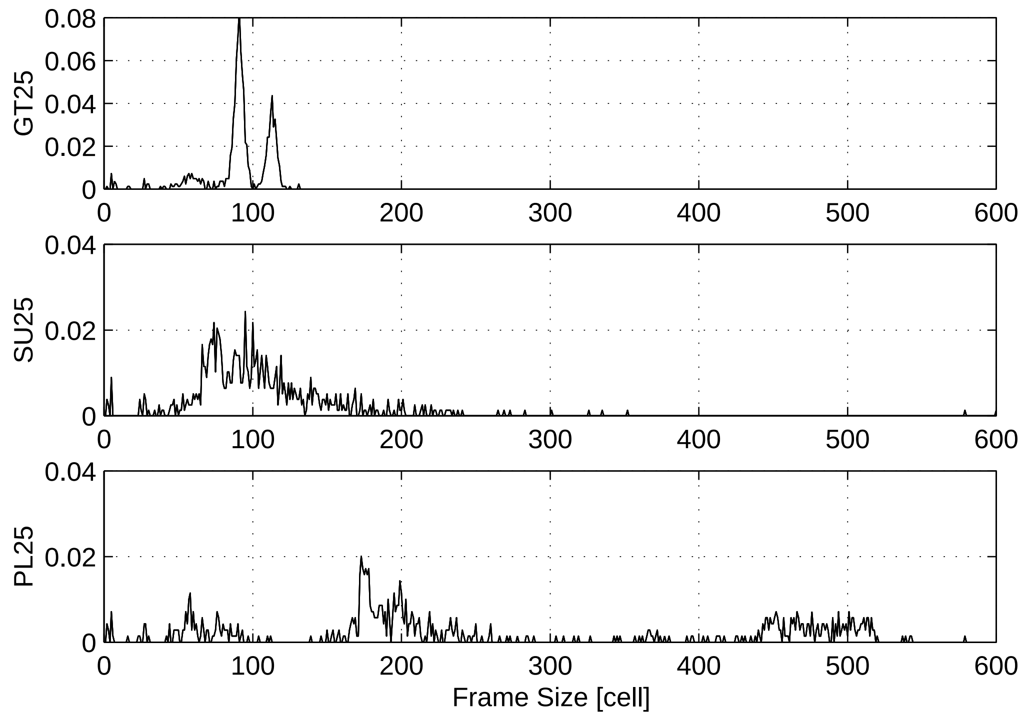


Figure 10 Probability Mass Function of Frame Sizes (GT25, SU25, and PL25)

It can be seen in Figure 10 that the frame sizes are more dispersed in case of the more dynamic video (PL25) than in the case of the almost still picture sequence (GT25). There is almost no full-size packet (N_p^*=172 cells) in GT25, while the large video frames are fragmented into two or three full-size and a smaller packet in the PL25 sequence. We have chosen the histograms depicted in this figure for describing the three video sequences in our model on frame level (see Section 5). We have performed the frame size analysis for the GT-PL10 sequences too. Noticeable is that the frame size statistics are very similar for the corresponding sequences and independent of the frame rate. This observation underscores the fact *that the frame size is determined by the video content and the scheduling of frames is an independent process*.

3.4.2 Frame Scheduling

Figure 11 demonstrates that in reality the frame interarrival time (T_{if}) is far from being a constant. Moreover, the mean frame interarrival time is 16 415 cell times for the GT-PL25 sequences. This value is larger than the theoretical value of 14 672 cell times which corresponds to 25 frame per second, therefore our hypothesis in Section 3.2.1. is confirmed, i.e. the terminal can not cope with higher frame rates. However, the mean frame interarrival time is 36 630 cell times for the GT-PL10 sequences which corresponds to the theoretical value.

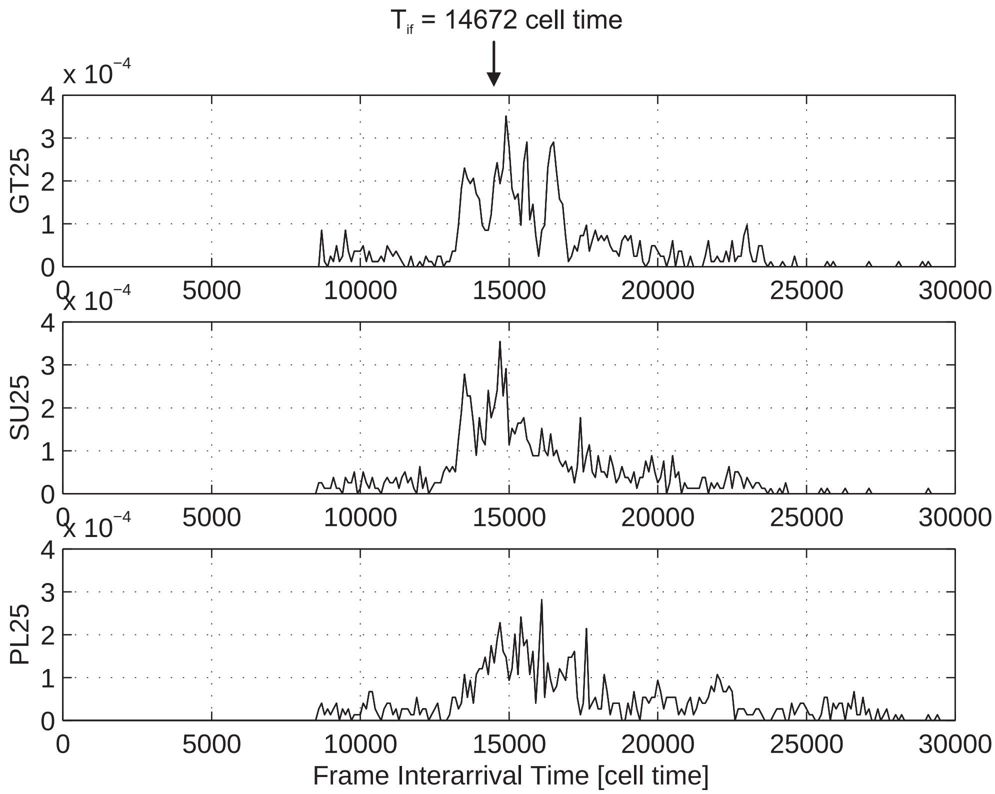


Figure 11 Probability Mass Function of Frame Interarrival Time (GT25, SU25, and PL25)

Independently from the video content, a common range can be determined between 8 000 and 28 000 cell time that contains the T_{if} values of the GT-PL25 sequences. This observation will be utilized for modeling frame scheduling (see section 5).

3.5 Scene Level Analysis

The frame correlation structure of the measured video sequences is shown in Figure 12. We can see that in each investigated case there is positive correlation among the frames. However, the correlation structure depends on the video content of the frames under investigation. In case of the bursty PL25 video sequence high short-term correlation with a linear decrease can be observed. In contrast, the less bursty SU25 and GT25 video sequences exhibit smaller short-term correlation but with a lower decay of the autocorrelation functions.

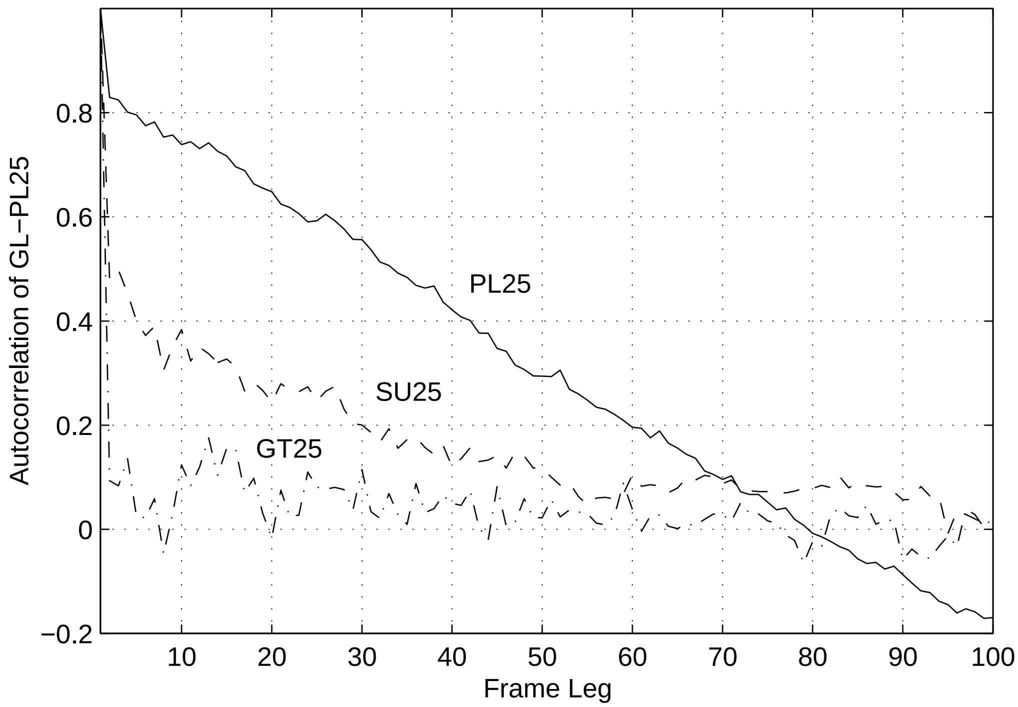


Figure 12 Autocorrelation of Frames (GT25, SU25, and PL25)

4. Inside the Black Box

We analyzed the VBR cell streams without examining the traffic generation procedure in the previous chapter, i.e. we considered the multimedia terminal as a black box. The next step is to investigate the internal behavior of this black box and map the traffic characteristics to information processing stages. This is a white box analysis, which helps to understand and enhance the results of the traffic analysis in Section 3.

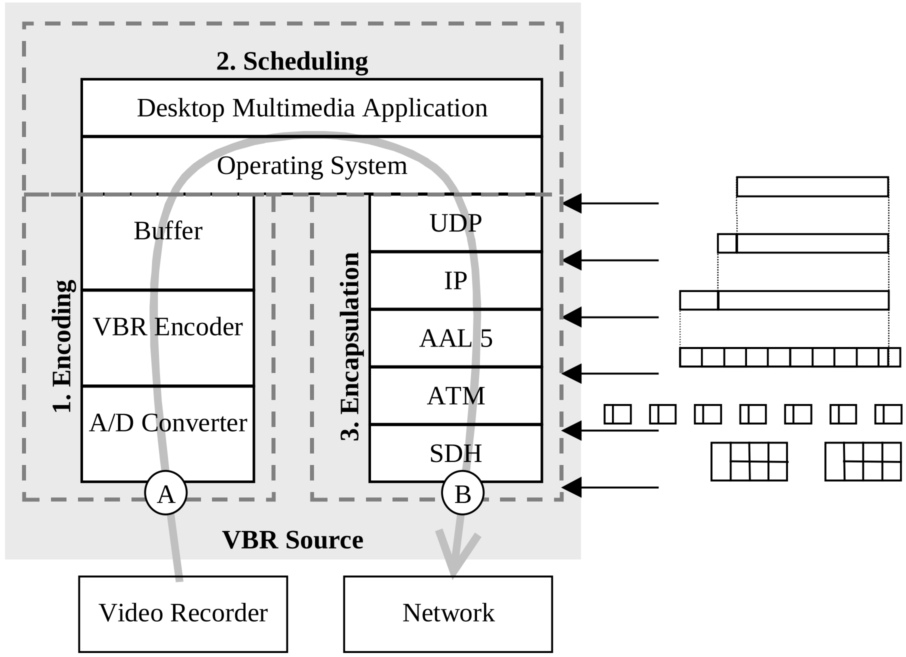


Figure 13 Three Stages of the Traffic Generation Procedure

The traffic generation procedure and the flow of video information from the video recorder to the network are depicted in Figure 13. The multimedia information is processed and transferred by several hardware and software units on its way in the terminal from the input of the video and audio card (denoted by “A”) to the output of the ATM card (denoted by “B”). Three main information-processing stages can be distinguished:

1. encoding
2. scheduling
3. encapsulation.

4.1 Encoding

The encoding stage receives video and audio signals from the VCR and provides video and audio frames to the multimedia application. This stage determines the frame size (N_f) and the scene level characteristics of the traffic. The analogue video signal arrives to the VBR codec after A/D conversion. Due to the variable bit rate coding, the size of video frame stored in the encoder’s buffer depends on the features of the video content (e.g. motion dynamics, spatial energy). The Cell-B compression uses motion estimation and intra-frame coding and it can achieve a range of compression from about 0.75 bits per pixel to 1.5 bits per pixel for each pixel in the image [26]. Thus the typical size of a coded video frame is between 10,000 and 17,000 bytes in case of a resolution of 384x288 pixels.

4.2 Scheduling

The multimedia application reads the codec's buffer, assembles the video frame and sends it to the output network socket according to the video frame rate set by the user. The *frame rate* was 10 fps and 25 fps in our investigations (see Table 1) and we denote it by *r*. The multitasking operating system determines when the multimedia application can read a frame from the codec's buffer; thus the actual frame interarrival time varies (T_{if}). Video frames are segmented by the application into transfer units that have a maximum length of 8192 bytes in our setting. This value corresponds to a full-size packet. These factors constitute the scheduling stage.

4.3 Encapsulation

The encapsulation stage determines the packet and cell level characteristics of the traffic. The transfer units of the application are further segmented into other data transfer units of different size, which are forwarded among the layers of protocol stack (as illustrated on the right side of Figure 13). For the sake of simplicity, we assume that there is no striping and each protocol layer sends its Protocol Data Unit (PDU) to the lower layer immediately after constructing it from the Service Data Unit (SDU) and the Protocol Control Information. In this case the processing time is proportional to the length of the SDU received from the above layer. This phenomenon has an impact on the length of silence period between bursts (see Section 3.3.2).

The maximum length of application data unit was 8192 bytes in our scenario. The UDP and IP layers insert an overhead of 8 and 20 bytes, respectively. The AAL5 layer adds an 8 bytes long trailer and a padding field of 28 bytes to get multiple of 48 octets, resulting in a packet of 172 cells on ATM level. That is the full packet size (N_p) can be calculated as:

$$N_p = (8192+8+20+8+28)/48 = 172 \text{ [cells]}$$

The encapsulation process is similar for shorter video frames (N_p^*). These calculations notify the results presented in section 3.3.1.

4.4 Scheduling and Encapsulation of Audio Frames

The audio information is processed in a similar way as video. The application sends audio PDUs eight times in every second. To get the constant bit rate of 64 Kbps, data units of 1024 bytes are forwarded to the protocol stack. Thus 1052 bytes arrive to the AAL5 layer that builds 23 ATM cells after segmentation and insertion of ATM headers:

$N_{paudio} = (1024+8+20+8+44)/48 = 23$ [cells]

The result is the same as that obtained in source analysis in section 3.3.1.

4.5 Summary of Main Source Characteristics

We have summarized the traffic characteristics retrieved from both black and white box analysis methods presented in the previous sections. Table 3 provides a set of parameters that can be considered from a modeling point of view. This set collects all the important information that can be gained from both the statistical analysis of measured traces and the understanding of the traffic generation procedure.

Parameter [Unit]	**Value @Video Sequence**	**Related Section of the paper**
$E\{N_f\}$ [cell]	250 @ PL25, 105 @ SU25, 94 @ GT25 358 @ PL10, 135 @ SU10, 100 @ GT10	3.5, 4.1
$Max\{N_f\}$ [cell]	524 @ PL25, 285 @ SU25, 145 @ GT25 545 @ PL10, 300 @ SU10, 145 @ GT10	3.5, 4.1
N_p^* [cell]	< 172	3.3, 4.3
N_p [cell]	91 @ GT; 172 @ PL and SU	3.3, 4.3
N_{paudio} [cell]	23	3.3, 4.4
T_{if} [cell time]	18000 – 53000, mean ~ 36630 @ GT-PL10 8000 – 28000, mean ~ 16415 @ GT-PL25	3.2, 3.5, 4.2
$T_{ipaudio}$ [cell time]	45860	4.4
T_{sf} [cell time]	14672 – 22000 @ GT-PL25, 30000 – 38000 @ GT-PL10	3.2
T_{sp} [cell time]	1500 – 3500 @ GT; 300 – 3500 @ PL	3.2, 3.3, 3.4
T_{sc} [cell time]	1-3	3.2
α [cell time/cell]	18.5	3.3
β [cell time]	323	3.3

Table 3 Traffic Parameters of Video Sequences Retrieved From The Black Box and White Box Analysis

5. Hierarchical Model

The previous sections showed that the characteristics of video traffic are determined by several independent factors, which belong to different stages of information processing within the VBR source. Our aim in this section is to construct a hierarchical model, which captures the behavior of these independent stages and imitates their effect on the video information. This model utilizes the results of both black box and white box analysis (see Figure 14). The synthetic traffic stream yielded by the model is verified by the Leaky Bucket Analysis method [5], which characterizes the burstiness of the VBR stream [15,16].

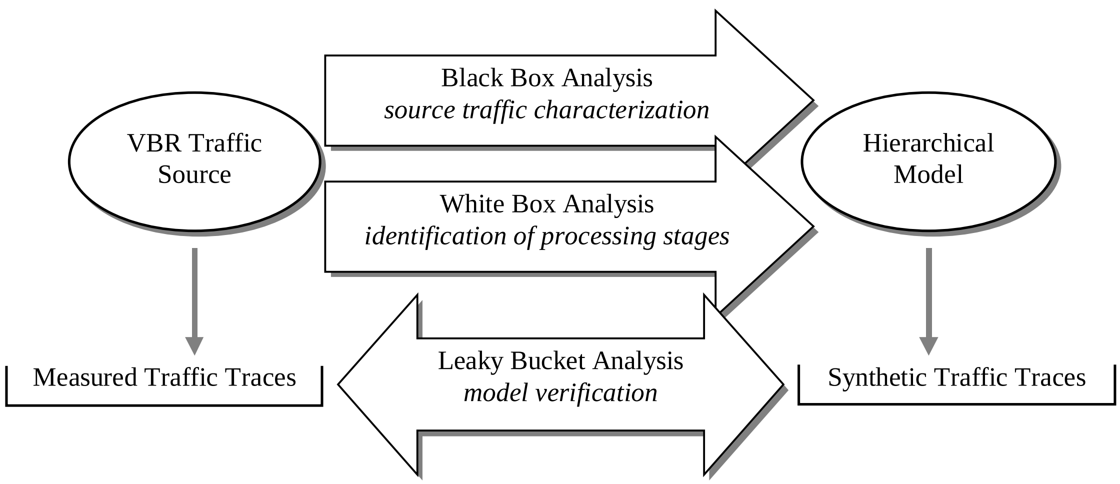


Figure 14 Concept of Model Building Technique and Verification

5.1 Model for Encoding

This model takes a sequence of captured frames, i.e. N_f^k, their probability mass function (see Figure 10) and the long term average rate (R) as inputs, and generates video frames of different size as output. We try to capture the scene level correlation between frames with a simple, two-state, discrete time Markov model (Figure 15), because the long-term correlation does not have significant impact on cell loss in most traffic situations [2,14,21,27] and the short-term correlation can be efficiently captured by Markov models. Moreover, increasing the Markovian order of the model does not improve its correlation behavior significantly [7]. Thus we avoid using complex, high-order Markov models.

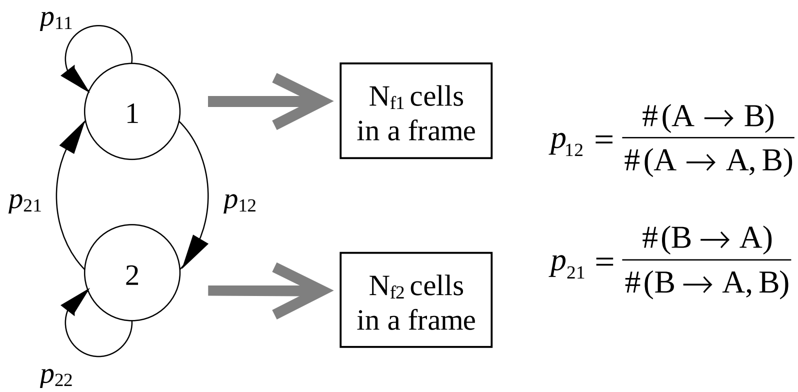


Figure 15 Frame Generation Model

The parameters of the Markov model are set according to the PMF of frame size and the N_f^k sequence, retrieved from the frame level analysis in section 3.4.1. The unit time of the Markov model is determined by the scheduling model, i.e. the next component of our hierarchical model. State 1 is assigned to the upper quantile of the probability mass (see shadowed area in Figure 16). The threshold (N_f^+) is determined by getting a predefined amount of the total probability mass to the upper part (we used 95%).

The transitions from State 1 to State 2 and vice versa are denoted by A→B and B→A in Figure 15, respectively. They can be described with the following expressions where N_f^k denotes the size of frame *k*:

A→B: $$N_f^k \subset \mathrm{A} \wedge N_f^{k+1} \subset \mathrm{B} \tag{5-1}$$

B→A: $$N_f^k \subset \mathrm{B} \wedge N_f^{k+1} \subset \mathrm{A} \tag{5-2}$$

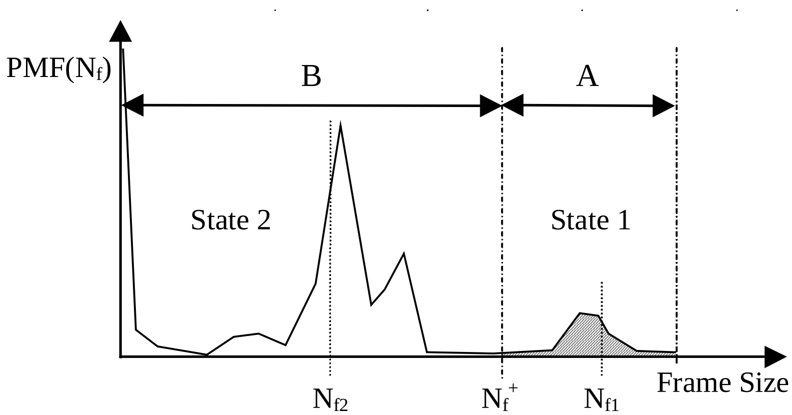


Figure 16 Frame Size Histogram

The encoding model generates N_{f1} cells in state 1, where N_{f1} is the sample mean of the upper quantile. In order to capture the long-term average rate, the frame size in state 2 is calculated with the following equation:

$$N_{f2} = T_{if} R \frac{p_{11} + p_{22}}{p_{11}} - N_{f1} \frac{p_{22}}{p_{11}} \tag{5-3}$$

where R denotes the long-term average cell rate of the video sequence (see Table 1). The transition probabilities (p_{12}, p_{21}) are estimated by the ratio of the number of events of state transitions to the total number of transitions as it is expressed in Figure 15.

5.2 Model for Scheduling

The scheduling process takes a synthetic frame sequence from the encoding stage as input and assigns timing to each frame according to the theoretical frame rate (in this paper r=10 or 25 fps). However, the operating system has a significant effect on scheduling in practice as we demonstrated in section 3.4.2. This stochastic nature of the operating system calls for a model to capture this behavior. We have chosen the normal distribution to describe the frame interarrival time with mean and variance set by measurements.

Consider M number of frames. Denote T_{if}^{i} the interarrival time of frame i. Let $m = \frac{1}{M}\sum_{i=1}^{M} T_{if}^{i}$ be the sample mean and $\sigma^2 = \frac{1}{M-1}\sum_{i=1}^{M}(T_{if}^{i} - m)^2$ be the sample variance of the normal variable t_{if} which is our model for the frame interarrival time. Note, that in this model we set the mean of t_{if} based on the measurements rather than based on the theoretical value of $T_{if}=1/r$, because our experiments showed that the mean frame interarrival time deviates from the theoretical value (see the Figure 11 in section 3.4.2 and Table 3 in section 4.5).

5.3 Model for Encapsulation

We model the encapsulation stage with a deterministic state machine, which receives frames of different size and timing from the scheduling stage and produces cell departures with certain packet and cell level characteristics. Figure 17 depicts the flowchart of sending one frame with this state machine.

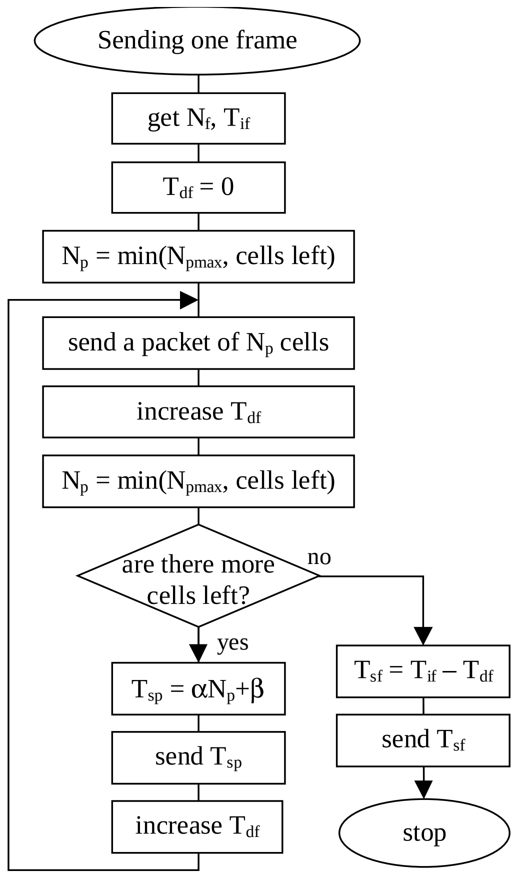


Figure 17 Deterministic Model for The Encapsulation of One Video Frame

The state machine reads a sequence of N_f, T_{if} pairs as input. The encoding model determines the frame size (N_f) while the (T_{if}) comes from the scheduling model. The parameters α, β and N_{pmax} are constant and they are retrieved from the analysis given in the previous sections. N_p, T_{df}, T_{sf} and T_{sp} are local variables of the state machine whose value is computed in runtime. The encapsulation model generates a cell interarrival time sequence, i.e. a synthetic traffic stream.

6. Model Verification and Applications

The aim of this chapter is to test the applicability and accuracy of our model. Thus we emulated the traffic of analyzed video sequences with our model and applied the synthetic traffic streams for queuing study. We derived the maximum queue length and estimated cell loss for different service rates and compared those characteristics with results for the captured traffic traces. At the end of this section we briefly overview the applications of our hierarchical model.

6.1 Leaky Bucket Analysis

The first performance test is the Leaky Bucket Analysis (LBA), because it characterizes the burstiness of the traffic simultaneously at several time scales [5,15]. Moreover, this analysis directly provides the traffic characteristics that are actually tested at the Usage Parameter Control, i.e. the peak cell rate and cell delay variation tolerance as well as sustainable cell rate and burst tolerance parameters [13]. That is we can see the traffic with the eyes of the ATM switch and judge the fitting of the model from that perspective. A detailed description of the leaky bucket traffic characterization method with applications can be found in [5].

In this paper, we use queuing theory terminology to describe the leaky bucket method, i.e. we consider the leaky bucket as a G/D/1 queue. We investigate the curves of the *maximum queue length* versus the *service rate* by feeding the recorded cell streams into the G/D/1 queue. The Leaky Bucket Curve and its slope (as a function of service time) are plotted for the three original sequences in Figure 18. It can be seen that the bursty PL25 sequence with higher mean cell rate has a stricter buffering requirement than the others. Different sections of the LB curve have different slope, representing different burst levels in the traffic stream, as it is emphasized by the slope curve on the right. One can observe, that the almost horizontal section of the three curves fits very well and they have very similar slope. This section characterizes the packet level behavior of the traces. The almost vertical sections differ due to the different scene level attributes. While there are one or more linear section between the aforementioned two extreme sections in case of the SU25 and PL25 sequences, the GT25 sequence seems to have only packet and scene level bursts. Table 3 acknowledges this observation showing that most of the frames consists only of one packet in the GT25 sequence.

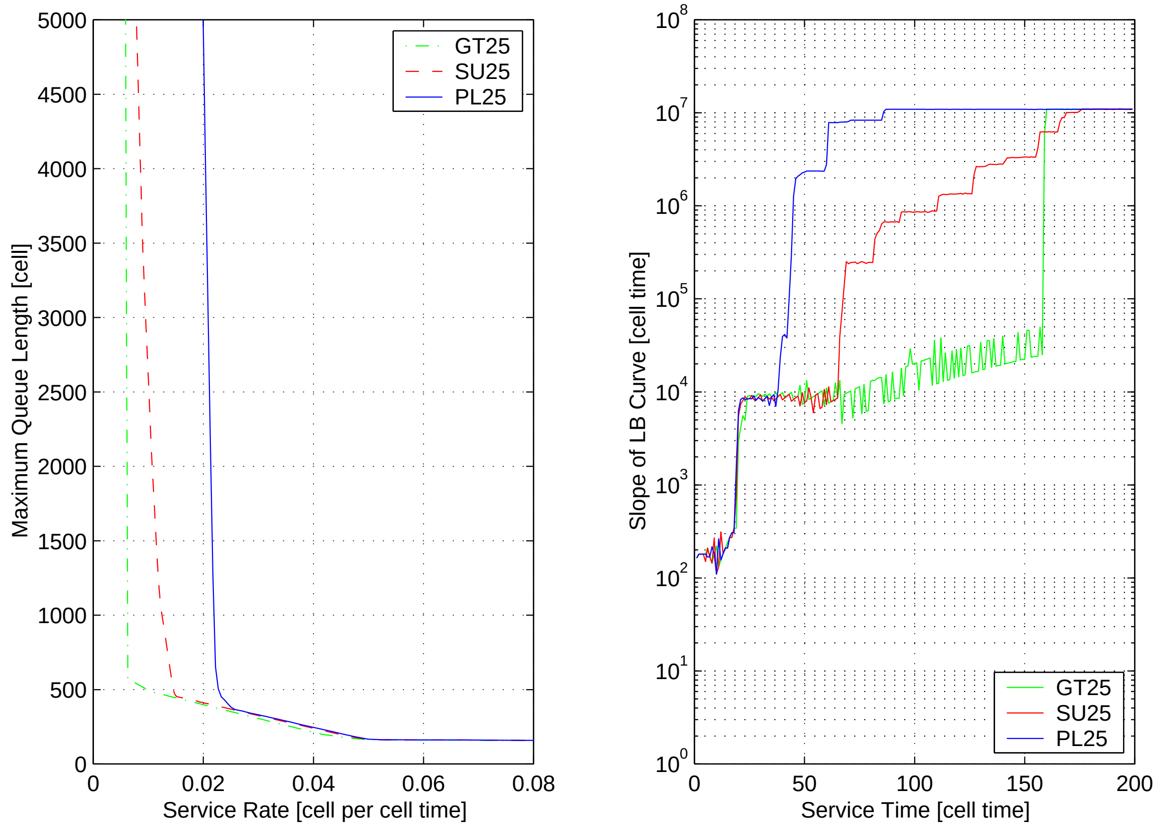


Figure 18 Leaky Bucket Curves (GT25, SU25, and PL25)

We performed the LBA both on the original sequences and the synthetic traces generated by our hierarchical model. In this way one can visualize the maximum queue length for different service rates. The results are plotted for the GT25 and PL25 sequences in Figure 19 and 20, respectively. Dashed lines represent the LB curves of the synthetic traces.

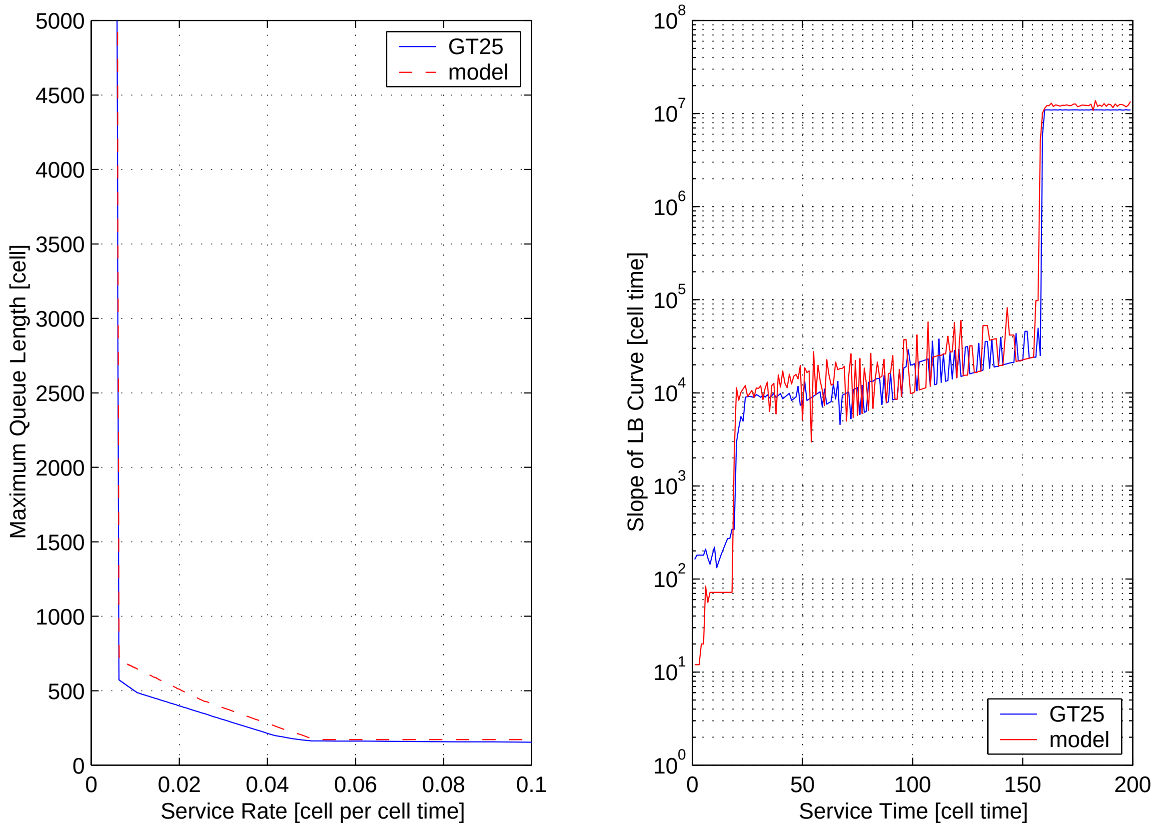


Figure 19 Leaky Bucket Curves of The Original and Synthetic GT25 Sequences

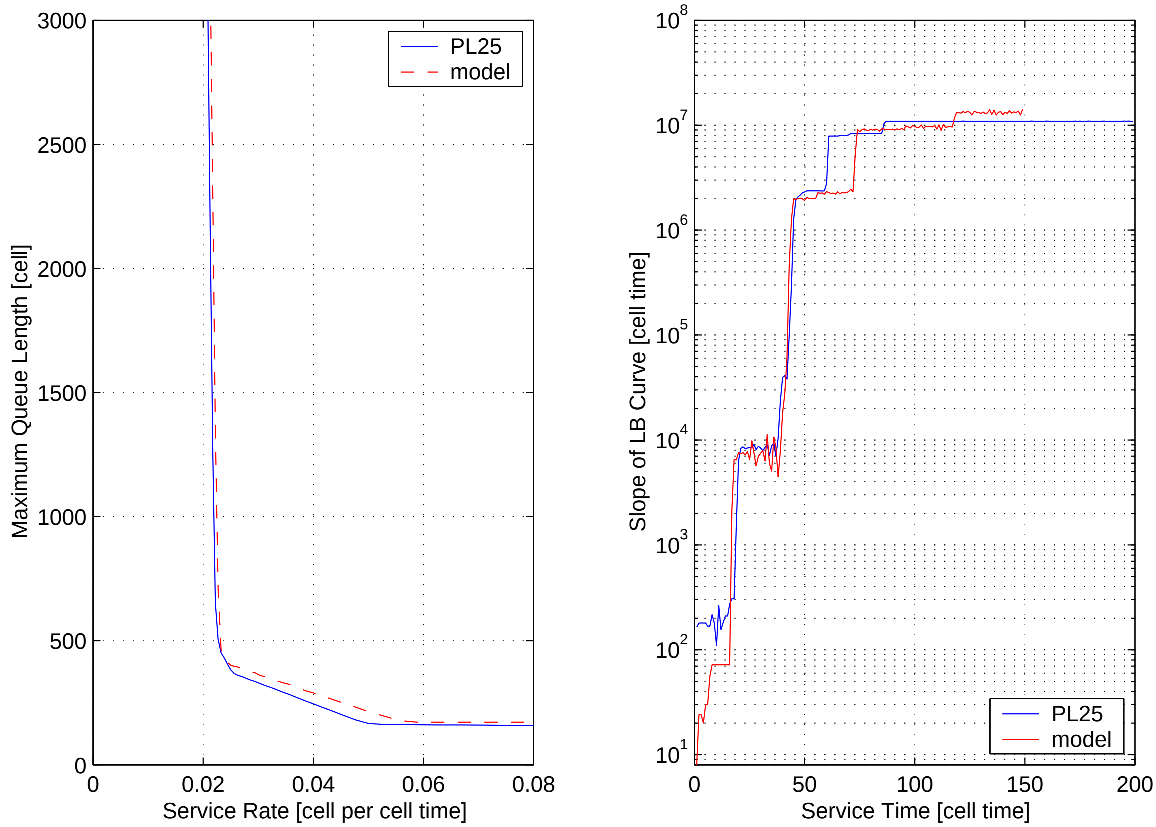


Figure 20 Leaky Bucket Curves of The Original and Synthetic PL25 Sequences

We can see in these figures that our model accurately captures the breaking points as well as the slope of the LB curve. The almost horizontal section of the curves is fully matched. The almost vertical section (i.e. where the service rate is below 2200 cell/sec for GT25 and 8300 cell/sec for PL25 sequence, respectively) represent the scene level. Although it was not our primary goal to have a good curve fitting in this range, the results are promising even on the scene level.

Therefore *our source model can be used for analyzing the queuing performance of video sources*.

6.2 Cell Loss Estimation

In order to show the performance of buffering, the CPDF for both the PL25 traffic and the model-generated traffic are depicted in Figure 21. We illustrated four cases corresponding to four different service rates. Dashed lines illustrate the curves of the model.

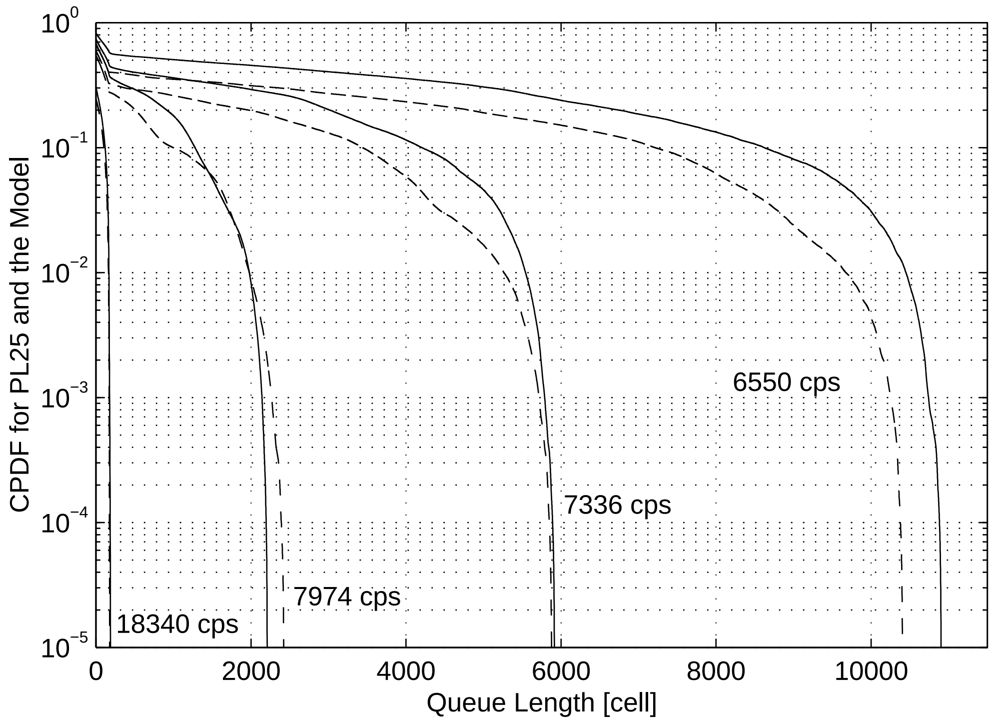


Figure 21 Complementary Distribution Function of Queue Length (PL25 and the Model)

This figure shows that the tail of the CPDF is close for the original and synthetic PL25 sequences in case of different service rates. In other words, *our model can reproduce the CPDF of queue length curve of the investigated VBR video sources and thus it can be used for cell loss estimation*. Note, that these results confirm that our model is able to capture those characteristics of the traffic which are important from the queuing point of view.

6.3 Applications and Generalization of the Model

Our model is built upon the detailed analysis of a particular multimedia platform. The results in the previous sections have shown that the queuing performance (in terms of maximum queue length and estimated cell loss) is successfully reproduced by the synthetic traffic of our model. Therefore a possible application of our model is to *emulate the behavior of a particular VBR source and reproduce its traffic*. In this case, the only variable input parameter of our model is the N_f^k sequence, which represents the scene level characteristics of particular video content, while other parameters are fixed.

Such emulated sources can be utilized in large simulation or measurement scenarios. Alternatively, one can determine the required ATM traffic contract parameters for a given source, perform multiplexing analysis using real and emulated sources, or estimate the foreseeable cell loss using this model. Moreover, network dimensioning and designing control methods (e.g. CAC) are further applications of this model.

Another interesting question is, whether *the presented model building technique is applicable for other multimedia platforms*. One of the key issues in our modeling concept is that different burst levels in the traffic correspond to different stages of the traffic generation process within the source host. Thus we construct a hierarchical model that consists of three independent model stages for the encoding, scheduling and encapsulation, respectively. We design and parameterize these models based on the analysis of a particular multimedia platform. However, most of the common VBR encoders, operating systems and hardware/software components (constituting the protocols stack) influence the traffic generation process in a similar way. Thus one can repeat the analysis steps given in Sections 3 and 4 for the new platform, and modify the encoding, scheduling and encapsulation models accordingly.

Therefore we hypothesize that our model building technique can be successfully adapted to other coding schemes (e.g. M-JPEG or MPEG), other operating systems and other network protocol stacks (e.g. IP over Ethernet). For instance, an investigation of bursts and silence periods in TCP/IP over ATM traffic can be found in [20,24].

7. Conclusion

This paper presents analysis of VBR video traffic. On top of traditional traffic analysis, we propose a "white-box" modeling technique, which is based on the a priori knowledge of the mechanisms in the multimedia terminal. Using this approach, we identified three independent traffic generation procedures and established a hierarchical model capturing them by three model stages. The video encoding and the scheduling of application transfer units are modeled with a two-state Markov process and a simple Gaussian process, respectively. A deterministic model imitates the data encapsulation in the protocol stack.

Our modeling technique focuses on the behavior of the traffic in the queue and not merely on the characteristics of the cell arrival process. The model is verified by leaky bucket analysis and cell loss analysis. Emulating VBR traffic sources for queuing analysis and for network dimensioning and control are typical applications of our model.

We infer the general validity of this modeling technique based on our study of a relatively large number of video sequences and performance settings. It is a reasonable belief that our concept can be adapted to other hardware and software platforms, because of the similar operation of other terminal architectures and VBR coding techniques. We have ongoing work on the generalization of our Hierarchical VBR Video Source Model.

Acknowledgements

We wish to thank Nils Björkman and Urban Hansson from Telia Research, Alexander Latour-Henner from Telia Business Unit Networks for their help during the preparation and fulfillment of the measurements and our colleagues from High Speed Networks Laboratory for their valuable comments on this paper.

References

[1] C. Blondia and O. Casals, Statistical Multiplexing of VBR Sources: A Matrix Analytic Approach, Performance Evaluation, Vol. 16, No. 5 (1992) 5-20.

[2] J. Beran, R. Sherman, M. S. Taqqu, W. Willinger, Variable-Bit-Rate Video Traffic and Long Range Dependence, IEEE Transactions on Communications, Vol. 43, No. 2/3/4 (1995) 1566-1579.

[3] N. Björkman, A. Latour-Henner, U. Hansson, O. Pers, A. Miah, Practical ATM Resource Dimensioning Based on Real-Time Traffic Measurements and Analysis, in: *Proc. IEEE GLOBECOM '95*, 1995, pp. 399-403.

[4] S. Blaabjerg, S. Molnár, Methods for UPC Dimensioning of a CDV Perturbated Cell Stream, in: *Proc. RACE BRAVE Workshop*, Milano, Italy, 1995

[5] I. Cselényi, N. Björkman and S. Molnár, How Does An ATM Switch See The Traffic Through The Leaky Bucket?, in: *4th IFIP book on ATM Networks, Performance Modelling and Analysis*, edited by D. Kouvatsos, Chapman and Hall, London, UK, 1998.

[6] I. Cselényi, S. Molnár, N. Björkman, A. Latour-Henner, Hierarchical VBR Video Source Model, in: *Proc. 5th International Conference on Telecommunication Systems Modeling and Analysis*, Nashville, Tennessee, 1997, pp. 401-410

[7] J. Roberts, U. Mocci, J. Virtamo, (Eds.) *Broadband Network Traffic, Performance Evaluation and Design of Broadband Multiservice Networks*, Final Report of Action COST 242, Spriger, 1996.

[8] N. G. Duffield, K. K. Ramakrishnan and Amy R. Reibman, Issues of Quality and Mulitplexing when Smoothing Rate Adaptive Video, in: *Proc. International Workshop on Network and Operating System Support for Digital Audio and Video* (NOSSDAV), Cambridge, England, 1998, pp. 265-278.

[9] V. S. Frost and B. Melamed, Traffic Modeling For Telecommunications Networks, IEEE Communications Magazine, (March 1994) 70-81.

[10] D. P. Heyman, A. Tabatabai, T. V. Lakshman, Statistical Analysis and Simulation Study of Video Teleconference Traffic in ATM Networks, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 2, No. 1 (1992).

[11] M. Hamdi, J. W. Roberts, P. Rolin, Rate Control for VBR Video Coders in Broad-Band Networks, IEEE JSAC, Vol. 15, No. 6, (1997) 1040-1051.

[12] P.E. Heegaard and B.E. Helvik, Establishing ATM Source Models for Traffic Measurement, in: *Proc. 11'th Nordic Teletraffic Seminar*, Stockholm, August 1993.

[13] ITU-T Recommendation I.371: *Traffic control and congestion control in B-ISDN*

[14] P. R. Jelenkovic, A. A. Lazar and N. Semret, The Effect of Multiple Time Scales and Subexponentiality in MPEG Video Streams on Queuing Behaviour, IEEE JSAC, Vol. 15., No. 5, (May 1997).

[15] D. M. Lucantoni, M. F. Neuts and A. R. Reibman, Methods for Performance Evaluation of VBR Video Traffic Models, IEEE/ACM Transactions on Networking, (Apr. 1994).

[16] S. H. Low and P. P. Varaiya, Burst Reducing Servers in ATM Networks, Queuing Systems: Theory and Applications 20 (1995) 61-84.

[17] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, J.D. Robbins, Performance Models of Statistical Multiplexing in Packet Video Communications, IEEE Transactions on Communications, Vol. 36, No. 7 (1988) 834-844.

[18] S. Molnár and I. Maricza, Source Characterization in Broadband Networks, COST 257 Interim Report, January, 1999.

[19] S. Molnár, I. Cselényi, N. Björkman, C-G. Perntz and M. Boda, ATM Traffic Measurements and Analysis on a Real Testbed, in: *Proc. 10th ITC Specialist Seminar*, Lund, Sweden, 1996, pp. 237-250.

[20] A. Rindos, S. Woolet, D. Cosby, L. Hango and M. Vouk, Factors Influencing ATM Adapter Throughput, IBM Technical Report 29.2099, available via WWW at URL: http://www.networking.ibm.com/per/per1.html (1998)

[21] B.K. Ryu, A. Elwalid, The Importance of Long-Range Dependence of VBR Video Traffic in ATM Traffic Engineering: Myths and Realities, in: *Proc. ACM SIGCOMM '96*, 1996, pp. 3-17.

[22] E.P. Rathgeb, Policing of Realistic VBR Video Traffic in an ATM Network, Int. Journal of Digital and Analog Communication Systems, Vol. 6 (1993) 213-226.

[23] G. Ramamurthy, B. Sengupta, Modeling and Analysis of Variable Bit Rate Video Multiplexer, in: *Proc. INFOCOM '92*, 1992, pp. 6.C.1.1 – 6.C.1.11.

[24] G. Seres, T. Éltetõ, A. Oláh, Measurement-Based Simulation Model for TCP over ATM, accepted for publication in Annals of Telecommunications, (May – June 2000), 278-287.

[25] G.D. Stamoulis, M.E. Anagnostou, A.D. Georgantas, Traffic Source Models For ATM Networks: A Survey, Computer Communications, Vol. 17, No 6 (1994) 428-438.

[26] *The Complete Guide to ShowMe 2.0.1*, SunSolutions, Sun Microsystems, Inc. 1994.

[27] A. Vidács, S. Monár, G. Gordos, I. Cselényi, The Impact of Long Range Dependence on Cell Loss in An ATM Wide Area Network, in: *Proc. IEEE GLOBECOM '98*, Sydney, Australia, November, 1998.

[28] W. Willinger, M. S. Taqqu and A. Erramilli, A bibliographical guide to self-similar traffic and performance modeling for high speed networks, in: *Stochastic Networks, Theory and Applications*, Oxford University Press, 1996, pp. 339-366.