Third International Conference on Multimedia Modeling, November 1996, Toulouse, France

Statistical Admission Control in Video Servers with Constant Data Length Retrieval of VBR Streams

Ernst W. BIERSACK, Frédéric THIESSE
Institut Eurécom, 2229 Route des Crêtes,
06904 Sophia-Antipolis — France
Phone: +33 93002611
FAX: +33 93002627
email: erbi@eurecom.fr

We consider the admission control problem in video servers that retrieve video data from disk storage. Admission control decides whether or not a new client can be accepted without affecting the quality of service promised to the already admitted clients. Assuming variable bit rate (VBR) video streams, we consider an admission control policy that provides statistical service guarantees and evaluate its performance in terms of the number of clients admitted.
The admission control criterion needs to take into account how the data are retrieved from disk We assume GCDL retrieval [1] that reads the video data from the disk as several constant size data blocks. We are the first to introduce and analyze statistical admission control with GCDL retrieval and to establish the correspondence between GCDL retrieval and ON-OFF sources in ATM networks.We show that *statistical* admission control for GCDL, compared to a deterministic admission control, admits up to twice as many clients with one overload event every few hours of video server operation.

Keywords: Video server, admission control, disk storage, disk retrieval.

1. Introduction

Video servers store digitized, compressed continuous media information on secondary or tertiary storage. The secondary storage devices allow random access and provide short seek times compared to tertiary storage. Video server design differs significantly from that of traditional data storage servers due to the large size of the objects stored and the real-time requirements for their retrieval. The critical resources in a video server are disk bandwidth, storage volume, and main memory. Given a fixed amount of these resources, a video server can only deliver a limited number of video streams simultaneously. Before admitting a new client, a video server must use an admission control algorithm to check if there are enough resources for serving the additional client.

The admission control criterion needs to take into account how the data are retrieved from the disk.

1.1 Retrieval Schemes in Video Servers

A video server must meet the requirements that stem from the continuous nature of audio and video and must guarantee the delivery of continuous media data in a timely fashion. We assume that video information is encoded as a **variable bit rate stream** (**VBR**) of *constant* quality. VBR requires sophisticated resource reservation mechanisms for the server and network to achieve a good utilization of the resources while maintaining a constant quality playback.

1.1.1 Deterministic Constraint Function for VBR Video

To provide deterministic quality of service (QOS) for VBR video, the admission control must employ *worst-case assumptions* about the data rate of the VBR video when computing the number of streams to be admitted. To offer deterministic service, we use a traffic model that is deterministic. The so-called empirical envelope presented in [5] provides a deterministic traffic constraint function for a given video trace. If $A_i[t, t+\tau]$ denotes the amount of video data consumed by a stream s_i in the interval $[t, t+\tau]$, an upper bound on A_i can be given by the **empirical envelope function** $\varepsilon_i(\tau)$ that is defined as:

$$\varepsilon_i(\tau) = \max_t A_i[t, t+\tau], \forall t \in [0, T_{total} - \tau] \quad (1)$$

1.1.2 Round-Based Retrieval Schemes

In the simplest case, continuous playback can be ensured by buffering the entire stream prior to initiating the playback [4]. Such a scheme, however, requires very large buffer space and causes a very large start-up latency. Consequently, the problem of efficiently servicing a single stream becomes one of preventing buffer starvation while at the same time minimizing the buffer requirement and the start-up latency. A video server that operates in rounds generally avoids starvation by *reading ahead* an amount of data that lasts in terms of playback duration through the next round (see figure 1). Data retrieval techniques determine the way data is read from the disk during a service round. Scheduling determines the order in which the requests within a round are served. Throughout the paper we assume SCAN scheduling that minimizes the seek overhead between adjacent retrievals.

The admission control scheme considered in this paper allows VCR functions such as fast forward, reverse, or pause) under the condition that the data rate required to support these functions is *not higher* than the data rate for normal playback.

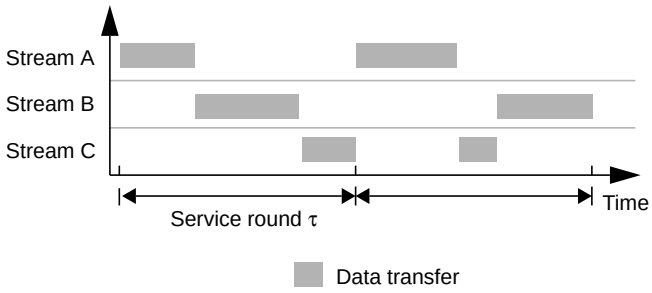

Figure 1. Sequence of service rounds with SCAN scheduling

Using VBR as data model for a video, one can map video data onto **data blocks (segments)** stored on the disk in two ways: **constant time length** (**CTL**) and **constant data length** (**CDL**) [2]. Throughout the paper we assume CDL retrieval. For a comparison between CDL and CTL see [3].

Constant data length (CDL) retrieval performs *non-periodic* retrieval of *constant* amounts of data from the disk (see figure 2). To make CDL compatible with round-based disk retrieval, we introduce the restriction that the distances between retrieval operations must be *multiples of a service round* τ, which will yield a sequence of **active** and **idle** rounds. During an active round, a *constant size* data block is read from the disk. Since the data must always (even in the worst case) be sufficient to supply the client with sufficient video data during the following round, the (fixed) size of the data block retrieved is of size $\varepsilon_i(\tau)$. During an idle round, no data at all is retrieved. The decision, whether a round will be active or not, can be made on-line: If there is still enough data in the buffer for the current and the next round, the current round is idle, otherwise it must be active.

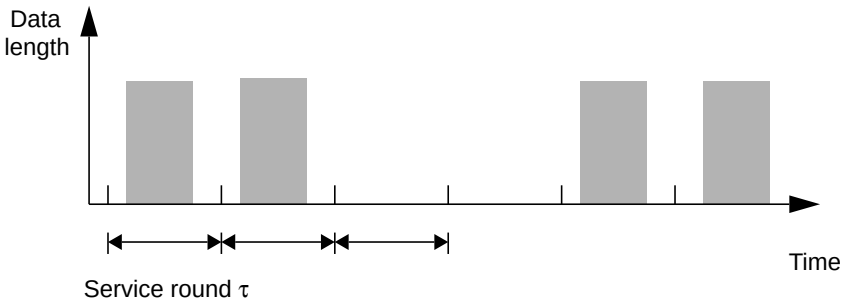

Figure 2. Constant data length retrieval

1.2 Generalized CDL

1.2.1 Introduction

Traditionally, the papers on periodic retrieval schemes have assumed that

- the **disk service round**, during which data for each stream are read exactly once from disk, and
- the **smoothing interval,** for which we compute the peak consumption rate

have the *same length*.

We have recently proposed [3] to distinguish the two and to make the smoothing interval a *multiple* of a disk service round. In detail, GCDL retrieval works as follows:

- The disk scheduling and retrieval still proceeds in rounds of length τ.
- However, we use a set $T = \{\tau_1, \ldots, \tau_n\}$ of smoothing intervals with $\tau_i \geq \tau$. To avoid starvation, we require that the amount of data retrieved for stream s_i from the disk during each interval τ_i must last for at least a period of τ_i. The smoothing interval duration τ_i is an integer multiple m_i of the disk service round duration τ (see figure 3). $\varepsilon_i(\tau_i)$ being the deterministic upper bound on the amount of data retrieved for stream s_i during any period τ_i, we require that the amount of data retrieved during any of the $m_i = \tau_i/\tau$ disk service rounds is the same, namely $\varepsilon_i(\tau_i)/m_i$.

The separation of disk service round and smoothing interval reduces the **peak consumption rate** defined as $\varepsilon_i(\tau)/\tau$. We see from figure 3 that the peak consumption rate decreases with increasing τ. GCDL gives the possibility to choose an optimal smoothing interval for each stream, which significantly reduces the buffer demand and the start-up latency while admitting the same number of clients.

A sequence of m_i consecutive disk service rounds where data for stream s_i are retrieved is also called an **active CDL round**. When during m_i consecutive disk service rounds no data are retrieved, that sequence is called an **idle CDL round** (see figure 3).

During an active CDL round a fixed amount $\varepsilon_i(\tau_i)$ of data is read, during an idle CDL round no data is read. Note that $\varepsilon_i(\tau_i)/m_i \leq \varepsilon_i(\tau)$, i.e. the amount of data that must be retrieved during an active disk service round becomes smaller, since increasing the smoothing interval τ_i reduces the peak consumption rate. Therefore, the smoothing effect gets stronger and less disk bandwidth is wasted. As the worst-case server load $\varepsilon_i(\tau_i)/m_i$ becomes smaller, less disk bandwidth must be reserved for that particular stream.

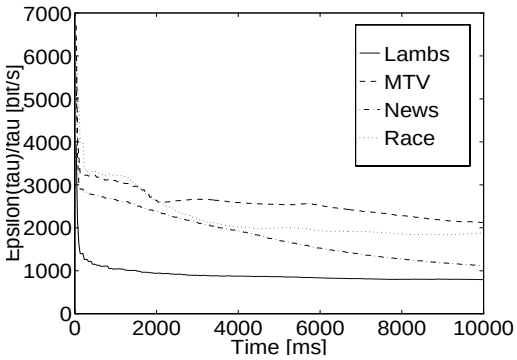

Figure 3. Peak consumption rate $\varepsilon_i(\tau)/\tau$ for 4 different videos

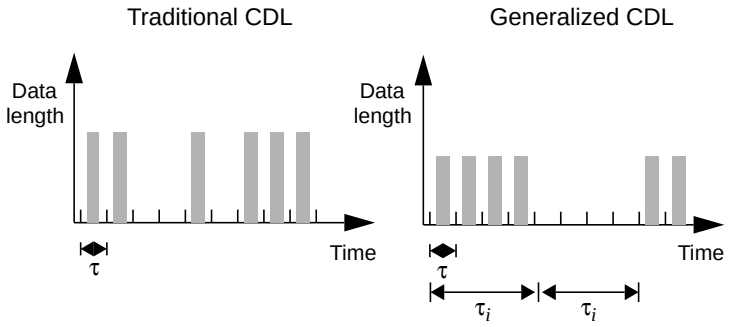

Figure 4. Traditional and generalized CDL

In the following, we will refer to the CDL retrieval where smoothing intervals and disk rounds have the same length ($\tau = \tau_i$) as **traditional** CDL. When smoothing intervals and disk rounds have different length ($\tau \neq \tau_i$), the scheme is referred to as **generalized CDL (GCDL)** retrieval. The traditional CDL can be regarded as a special case of GCDL with $m_i = 1$.

2. Admission Control Strategies

Because of its limited resources in terms of buffer space and disk bandwidth, a video server needs to decide for each new client whether an additional multimedia stream can be admitted without degrading the quality of service (QoS) of all other clients that are already admitted. Two major service models are:

- *Deterministic*: In order to assure a deterministic service without any loss or delay

of frames at any time, the admission control algorithm considers only worst-case scenarios before admitting a new client. Deterministic service for VBR traffic results in a very inefficient allocation of resources.

- *Statistical*: The decision whether a new client will be admitted or not depends on the overload probability that all clients (admitted and new ones) are willing to accept. This service models achieves a much more efficient resource utilization than deterministic service. The deterministic case can be regarded as a special case with an overload probability of zero.

2.1 Deterministic Admission Control for GCDL Retrieval

The number of streams admitted is limited by the length of a disk service round, the available buffer space and the disk bandwidth. If we assume that the buffer space is not a scarce resource, the admission control criterion for GCDL, when SCAN scheduling is used, is given by [1]:

$$\sum_{i=1}^{n} \left\lceil \frac{\varepsilon_i(\tau_i)}{m_i} \right\rceil \cdot r_{disk}^{-1} + \sum_{i=1}^{n} \left\lceil \left\lceil \frac{\varepsilon_i(\tau_i)}{m_i} \right\rceil \cdot c_{cyl}^{-1} \right\rceil \cdot t_{track} + n \cdot (t_{track} + t_{rot}) + t_{seek} \leq \tau \qquad (2)$$

In this formula $r_{disk} = 24 \cdot 10^6$ bit/s denotes the disk bandwidth, $c_{cyl} = 4 \cdot 10^6$ bit equals the capacity of a single cylinder and $t_{track} = 1.5$ ms, $t_{rot} = 11.11$ ms and $t_{seek} = 20.0$ ms denote the track-to-track seek time, the rotational latency and the maximum seek time for a complete scan over the entire disk, and m_i denotes the number of disk service rounds within a single CDL round.

2.2 Statistical Admission Control for GCDL Retrieval

A deterministic service can be assured at any time during the playback by using the *worst case* traffic characterization, given by $\varepsilon_i(\tau_i)$, for the admission control criterion. Deterministic service results an inefficient use of the server's resources, such as disk bandwidth and buffer space. Figure 5 shows histograms of a client's data consumption per stream during a CDL round of 4 sec for 18 different video traces[1]. Obviously, during the majority of all CDL rounds, a client consumes much less video data than the envelope function $\varepsilon_i(\tau_i)$ (worst-case consumption) suggests, i.e. in most rounds the server allocates much more bandwidth than necessary. Therefore, a high number of CDL rounds will be *idle*.

[1] The video traces were produced by O. Rose using a MPEG 1 codec and are available via ftp anonymous on the machine ftp-info3.informatik.uni-wuerzburg.de in the directory /pub/MPEG.

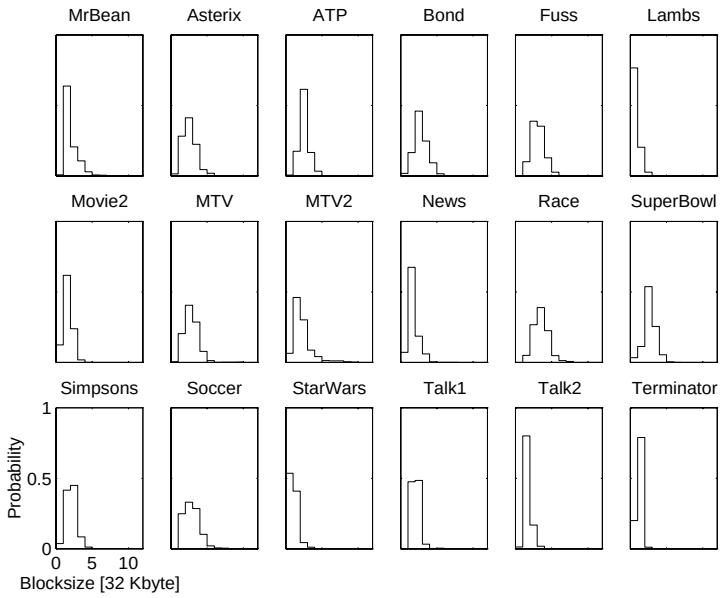

Figure 5. Histograms of the amount of data consumed during a CDL round of length 4 sec, i.e $\tau_i = 4$ s.

2.2.1 Statistical Traffic Characterization

To provide statistical service guarantees, a video server needs a precise statistical traffic characterization that is used to compute the probability of a server overload, which is defined as the probability of the occurrence of a situation where the video server cannot assure the timely delivery of all requested media data to all clients.

If a GCDL round is active, m_i constant data blocks of size $\varepsilon(\tau_i)/m_i$ are retrieved from the disk during m_i consecutive disk service rounds. This allows us to describe the retrieval behavior for all disk service rounds by a **CDL sequence** $\{x_1, \ldots, x_{T_{total}/\tau}\}$ with $x_k \in \{0, 1\}$ depending whether this disk service round is idle or active. If mbr_i denotes the **average bit rate** of stream s_i, the **probability of an active round** p_i is given by

$$p_i = \frac{mbr_i \cdot \tau_i}{\varepsilon_i(\tau_i)} \tag{3}$$

The process that produces such a CDL sequence will be called a **CDL process**. In this section we present an analysis of CDL processes that introduces an efficient method for calculating the overload probabilities.

The objective of the characterization of the CDL process is to capture its mathematical properties in order to get a correct approximation of its behavior. The first step is to test if the CDL process has the same characteristics as a memoryless Poisson process. In this case, the On-Off nature of CDL retrieval would allow for a direct application of the binomial distribution to model simultaneous CDL processes. For that purpose we check for video stream s_i, whether equation (4) is fulfilled:

$$p(x_{j+k} = 1 | x_j = 1) = p(x_{j+k} = 1), \forall j, k \tag{4}$$

The results can be seen in figure 6 for a set of four different videos with a lag k between 1 and 25.

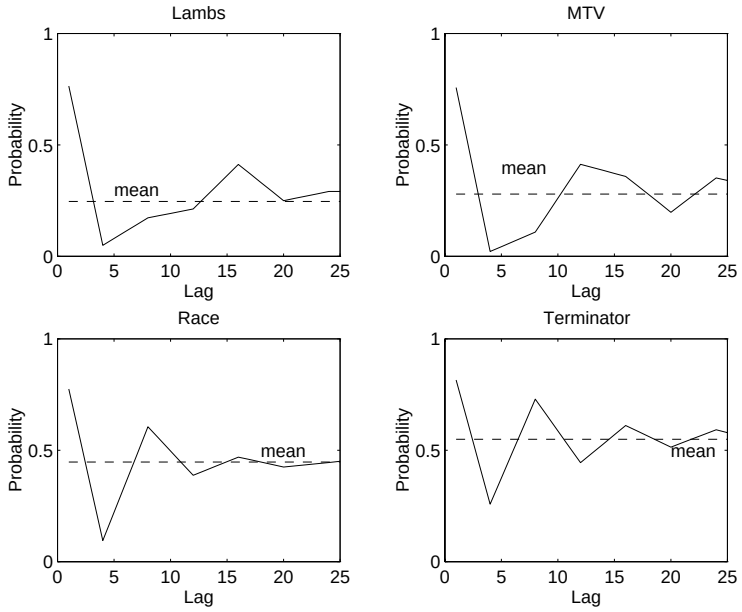

Figure 6. Memory behaviour of CDL processes, i.e. $p(x_{j+k} = 1 | x_j = 1)$

Obviously the CDL process differs from an ideal Poisson process for small lags k since there are always m_i *consecutive* active or idle disk service rounds. On the other hand, as k increases, we see that the conditional probability $p(x_{j+k} = 1 | x_j = 1)$ converges to its mean value. This analysis tells us that the considered CDL process is not memoryless. However, we can also conclude that it can be regarded as approximately memoryless for $k > 10$, i.e. under the assumption that two clients of the same video

watch at the same instant of time parts of the video that are at least 10 disk service rounds apart. Since in reality this assumption normally holds true, the retrieval operations of these two streams that take place within the same disk service round can be regarded as independent even for the same video. Furthermore we assume that two CDL sequences obtained from two different videos are fully independent.

As our simulations showed, the binomial distribution delivers a very good approximation for the distribution of the number of active streams during any round (see figure 7). Interestingly, the binomial distribution produces an estimation of the overload that yields values slightly higher, i.e. is more conservative, which assures that we do not underestimate the actual overload probabilities.

Furthermore, we see that the Gaussian distribution proposed by Vin [7] does not properly approximate the behavior obtained in our simulation. In particular, in the probability range of interest the Gaussian distribution commits an average error of one stream. We also see that the Gaussian distribution is not conservative, i.e. for a certain number of streams yields a lower number of concurrent requests than the actual value obtained in our simulations. Therefore, the overload probability will be *underestimated* when using the Gaussian distribution. If we compare the plots for 20 and 80 streams, respectively, we see that Gaussian approximation gets better for a higher number of streams due to the central limit theorem.

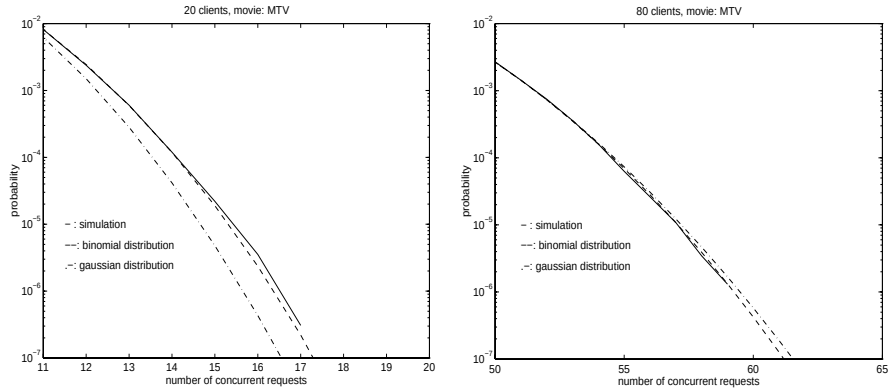

Figure 7. Comparison of simulations to the binomial distribution and the Gaussian distribution. MTV video with $\tau = 1$ s and $\tau_i = 4$ s.

Using the Binomial distribution, the probability of k active rounds for n homogeneous streams s_i is then given by

$$p_i(k) = \binom{n}{k} \cdot p_i^k \cdot (1 - p_i)^{n-k} \tag{5}$$

Note that this calculation is equivalent to histogram convolution using a histogram that contains only two values.

2.2.2 Overload Probability for Homogeneous Streams

If all clients request the same video (homogenous case), the maximum number of parallel requests that can be served during one disk service round is equivalent to the maximum number of simultaneous streams in the deterministic case N_{det}, and can be stated as follows (see Eq (2)):

$$N_{det} = \left\lfloor \frac{\tau - t_{seek}}{\left\lceil \frac{\varepsilon_i(\tau_i)}{r_{disk} \cdot m_i} \right\rceil + \left\lceil \frac{\varepsilon_i(\tau_i)}{c_{cyl} \cdot m_i} \right\rceil \cdot t_{track} + t_{track} + t_{rot}} \right\rfloor \quad (6)$$

An overload event will occur when more than N_{det} streams are active during the same disk service round. Therefore, we compute the overload probability $p_i^{overload}$ as:

$$p_i^{overload} = \sum_{k = N_{det}+1}^{n} p_i(k) \quad (7)$$

2.2.3 Overload Probability for Heterogeneous Streams

In the heterogenous case, where different clients request different videos, the probabilities can be calculated by *histogram convolution*. The histogram of a stream s_i contains only two values that are non-zero, namely the probability $1 - p_i$ of an idle round, where no data are read, and the probability p_i of an active round, where $\varepsilon_i(\tau_i) / m_i$ data are read. However, we also need to take into account the overhead that occurs when reading the data for a stream[2], which amounts to $2 \cdot t_{seek} + t_{rot}$. For this purpose, we assume that the amount of data to be read during an active round is $\varepsilon_i(\tau_i) / m_i + (2 \cdot t_{track} + t_{rot}) \cdot r_{disk}$ instead of $\varepsilon_i(\tau_i) / m_i$. Therefore, the characteristic **data rate histogram** h_i for stream s_i is defined as follows:

- $h_i(0) = 1 - p_i$
- $h_i(\varepsilon_i(\tau_i) / m_i + (2 \cdot t_{track} + t_{rot}) \cdot r_{disk}) = p_i$.
- All the other values of h_i are zero.

---

[2] We assume that two track-to-track seek operation occur when reading the data for a stream: One during the retrieval of the stream's data block and another one when the server switches from one stream to the next. In our simulations this simplification has no influence on the results since the capacity of a disk cylinder is always larger than the size of a data block for all considered τ and τ_i, i.e. $c_{cyl} > \varepsilon_i(\tau_i) / m_i$.

The characteristic server load histogram after the admission of the (n+1)-th stream s_{n+1} can be computed by convolving the **server load histogram** H_n with the histogram of the newly requested video, h_{n+1}. The result H_{n+1} is a histogram whose k-th element is given by:

$$H_{n+1}(k) = \sum_{i=1}^{k-1} H_n(i) \cdot h_{n+1}(k-i) \tag{8}$$

If the random variable $\tilde{D}$ denotes the server load during a disk service round, the probability of a server overload $p(\tilde{D} > D^{max})$ is obtained by computing the tail of H_{n+1} beyond the maximum data rate D^{max}:

$$p(\tilde{D} > D^{max}) = \sum_{k=D^{max}+1}^{\infty} H_{n+1}(k) \tag{9}$$

D^{max} is the maximum amount of data that can be retrieved during a single disk service round τ. D^{max} can be defined as the product of the time available for data transfer and the disk's transfer rate:

$$D^{max} = (\tau - t_{seek}) \cdot r_{disk} \tag{10}$$

Having given the formulas for computing the overload probabilities, we are now evaluating the performance gains due to statistical admission control. We assume the use of a single disk with the parameters given in 2.1 and the video traces given in 2.2.

2.2.4 Performance Comparison for Homogeneous Streams

In the following, we will restrict our evaluation of statistical admission control of GCDL to the homogeneous case. The number of admitted streams is given for different overload probabilities $N_{10^{-5}}$, $N_{10^{-4}}$ and $N_{10^{-3}}$, where $N_{10^{-x}}$ denotes the number of streams that can be admitted without exceeding an overload probability of 10^{-x}. For example, an overload probability of 10^{-4} means that on the average every 10^4-th disk service round an overload occurs. For $\tau = 1$ s, this is equivalent to an overload event every 2.8 hours. The values are compared to the following results of deterministic admission control: N_{det} denotes the maximum number of admitted streams for deterministic GCDL retrieval as given in Eq (6).

To compare the deterministic and the statistical admission control, we use the gain $G_{10^{-x}}$, which is defined as the improvement in terms of the number of streams admitted in the statistical case when compared to the deterministic case:

$$G_{10^{-x}} = \frac{N_{10^{-x}} - N_{det}}{N_{det}} \tag{11}$$

The results for the homogenous case are summarized in table 1. We note that the gain to be achieved by a statistical service depends on the mean bit rate $mbr_i = p_i \cdot \varepsilon_i(\tau_i) / \tau_i$ of a video. The range of the gain varies widely, going from 0% for the video 'Race' up to 89% for 'Lambs'.

Table 1: Statistical admission control for GCDL data retrieval (homogeneous case)

This table shows the results of statistical admission control for different videos and $\tau = 1\,\text{s}$, $\tau_i = 4\,\text{s}$, and $r_{disk} = 24 \cdot 10^6$ bit/s. The videos are sorted in increasing order of their mean bit rate.

Video	Det. adm	Statistical admission		$\frac{\varepsilon_i(\tau_i)}{\tau_i}$ [Mbit/s]	p_i	mbr_i [Mbit/s]
	N_{det}	$N_{10^{-4}}$	$G_{10^{-4}}$			
Lambs	19	36	0.89	0.85	0.24	0.22
StarWars	17	28	0.65	0.97	0.27	0.28
Terminator	25	29	0.16	0.57	0.54	0.33
Movie2	16	20	0.25	1.04	0.39	0.43
News	10	15	0.50	1.89	0.23	0.46
MrBean	10	13	0.30	1.75	0.28	0.53
Simpsons	13	15	0.15	1.33	0.40	0.56
MTV2	8	11	0.38	2.47	0.22	0.59
Asterix	11	12	0.09	1.61	0.39	0.67
MTV	7	8	0.14	2.53	0.27	0.74
Fuss	11	11	0.00	1.66	0.46	0.81
Race	9	9	0.00	1.98	0.44	0.92

However, there is no linear dependence between the mean bit rate and the gain: For instance, the 'Terminator' trace achieves a lower gain than 'Movie2' although 'Terminator' has a lower mean bit rate. We observe that the highest gain is obtained for a stream s_i in which both,

- the probability p_i of an active round and
- the peak consumption rate $\varepsilon_i(\tau_i) / \tau_i$

  are *small* (see figure 8).

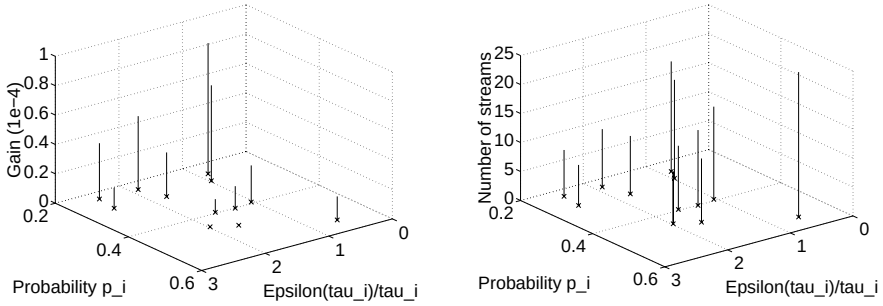

Figure 8. $G_{10^{-4}}$ and N_{det} as function of $\varepsilon_i(\tau_i)/\tau_i$ and p_i
All computations were done for $\tau = 1$ s, $\tau_i = 4$ s, and $r_{disk} = 24 \cdot 10^6$ bit/s.

Our observation is in conformance with what has been defined as **statistical multiplexing gain** for ATM nodes [6]. Figure 9 illustrates the 'classical wisdom' of statistical multiplexing.

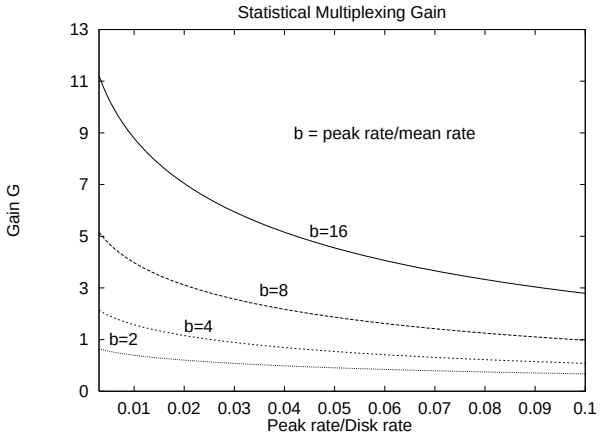

Figure 9. Multiplexing Gain

In order to achieve a high statistical multiplexing gain:

- The **burstiness** b of a stream, which is defined as the ratio between peak rate and average rate, must be high. In our case, the ratio of the peak rate and the average rate is given by $1/p_i$, i.e. the burstiness is inversely proportional to p_i and takes values between 2 and 4.
- The number of streams must be large, i.e. the peak rate of any individual stream should be low with respect to the link rate. In our case, the link rate corresponds to the disk bandwidth and the peak consumption rate of stream s_i to $\varepsilon_i(\tau_i)/\tau_i$.

When reading from disk, a significant overhead occurs due the required head movement and disk rotation between read operations that does not have an equivalent in the model of the service of an ATM node. Therefore, we must use $1/N_{det}$ as value for the ratio of peak rate to link rate**.**

In table 2, we have increased the disk transfer rate by a factor of four compared to the value used in table 1.

Table 2: Statistical admission control for GCDL data retrieval (homogeneous case)

This table shows the results of statistical admission control for different videos and $\tau = 1\,\text{s}$, $\tau_i = 4\,\text{s}$, and $r_{disk} = 96 \cdot 10^6$ bit/s. The disk transfer rate was increased by factor of four compared to table 1.

Video	Det. adm.	Statistical admission		$\frac{\varepsilon_i(\tau_i)}{\tau_i}$	p_i	mbr_i [Mbit/s]
	N_{det}	$N_{10^{-4}}$	$G_{10^{-4}}$	[Mbit/s]		
Lambs	41	99	1.41	0.85	0.24	0.22
StarWars	39	84	1.15	0.97	0.27	0.28
Terminator	48	62	0.29	0.57	0.54	0.33
Movie2	34	60	0.58	1.04	0.39	0.43
News	28	64	1.29	1.89	0.23	0.46
MrBean	29	56	0.93	1.75	0.28	0.53
Simpsons	34	52	0.53	1.33	0.40	0.56
MTV2	23	49	1.31	2.47	0.22	0.59
Asterix	30	45	0.50	1.61	0.39	0.67
MTV	23	42	0.83	2.53	0.27	0.74
Fuss	30	40	0.33	1.66	0.46	0.81
Race	27	36	0.33	1.98	0.44	0.92

Increasing the disk rate results in higher values for N_{det}, which means that the ratio of the peak rate of a stream to the disk rate will become smaller, while the burstiness of a stream remains unchanged. For small values of burstiness, as is our case, the slope of the curves for the gain is quite flat (see figure 9), the increase of the gain $G_{10^{-4}}$ is modest. The gain obtained is now between 0.33 and 1.41, as compared to values between 0.0 and 0.89.

2.2.5 Performance Comparison for Heterogeneous Streams

We also simulated the heterogeneous case, where clients request videos out of a set of three different videos. For each triple of videos, we computed the number of streams admitted under deterministic and the statistical admission control. We assumed that videos are requested in a round-robin fashion, i.e. client i requests video $((i-1) \bmod 3) + 1$. New clients are admitted in a round-robin fashion as long as the overload probability is smaller than 10^{-4}. Our results are summarized in table 3. For the heterogeneous case, statistical admission control yields similar improvements as in the homogeneous case. The gain obtained always lies somewhere between the gains we observe for the considered videos in the homogeneous case. The same holds for the number of admissions in the deterministic and the statistical case, i.e. an estimation of the performance in the heterogeneous case can easily be obtained using the results from the homogeneous case. However, the gain in the heterogeneous case is not simply the mean of the three gains obtained in the homogeneous case, since a gain of a video with a large retrieval block size $\varepsilon_i(\tau_i)/m_i$ will have a stronger impact on the gain in the heterogeneous case than a video with a small retrieval block size.

Table 3: Statistical admission control for GCDL data retrieval (heterogeneous case). This table shows the results for different videos and $\tau = 1$ s and $\tau_i = 4$ s.

	Videos	Det. adm.	Statistical admission	
		N_{det}	$N_{10^{-4}}$	$G_{10^{-4}}$
$r_{disk} = 24 \cdot 10^6$ [bit/s]	Lambs, StarWars, Terminator	19	29	0.53
	Movie2, News, MrBean	11	15	0.36
	Simpsons, MTV2, Asterix	10	12	0.20
	MTV, Fuss, Race	9	10	0.11
$r_{disk} = 96 \cdot 10^6$ [bit/s]	Lambs, StarWars, Terminator	42	78	0.86
	Movie2, News, MrBean	31	59	0.90
	Simpsons, MTV2, Asterix	28	49	0.75
	MTV, Fuss, Race	26	39	0.50

2.3 Conclusion

We have introduced a statistical admission control criterium for GCDL data retrieval and evaluated the improvement in terms of additional streams admitted compared to the case of deterministic admission control. We also established the correspondence between the CDL retrieval process and the On-Off process used to model ATM data streams, which allowed us to use the results about the statistical multiplexing gain in ATM nodes to explain and predict the performance improvements obtained for the statistical admission control criterium applied to the data retrieval from disk. For very low overload probabilities in the order of one overflow round within 2.8 hours, the maximum number of streams can be significantly increased when using statistical admission control.

References

[1] E. W. Biersack, F. Thiesse, and C. Bernhardt. Constant data length retrieval for video servers with variable bit rate streams. In *IEEE Conf. Multimedia Systems*, pages 151–155, Hiroshima, Japan, June 1996.

[2] E. Chang and A. Zakhor. Admission control and data placement for VBR video servers. In *Proceedings of the 1st International Conference on Image Processing*, Austin, Texas, November 1994.

[3] J. Dengler, C. Bernhardt, and E. W. Biersack. Deterministic admission control strategies in video servers with variable bit rate streams. In B. Butscher, E. Moeller, and H. Pusch, editors, *Interactive Distributed Multimedia Systems and Services, European Workshop IDMS'96, Berlin, Germany*, volume 1045 of *LNCS*, pages 245–264. Springer Verlag, Heidelberg, Germany, Mar. 1996.

[4] D. J. Gemmell, H. M. Vin, D. D. Kandlur, P. V. Rangan, and L. A. Rowe. Multimedia storage servers: A tutorial and survey. *IEEE Computer*, 28(5):40–49, May 1995.

[5] E. W. Knightly, D. E. Wrege, J. Liebeherr, and H. Zhang. Fundamental limits and tradeoffs of providing deterministic guarantees to VBR video traffic. In *Proceedings Sigmetrics '95 / Performance '95*, volume 23 of *Performance Evalution Review*, pages 98–107, Ottawa, Canada, May 15-19 1995.

[6] D. E. McDysan and D. L. Spohn. *ATM: Theory and Application*. McGraw Hill, New York, 1995.

[7] H. M. Vin, P. Goyal, A. Goyal, and A. Goyal. A statistical admission control algorithm for multimedia servers. In *Proceedings of the 2nd ACM International Conference on Multimedia*, San Francisco, CA, October 1994.