Long Range Dependent Traffic: Modeling, Simulation and Congestion Control

by

Changcheng Huang, M.Eng.
Tsinghua University

A thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

Ottawa-Carleton Institute for Electrical Engineering
Department of Systems and Computer Engineering

Carleton University
Ottawa, Ontario
January 3, 1997

The undersigned recommend to the Faculty of Graduate Studies

and Research acceptance of the thesis

Long Range Dependent Traffic: Modeling, Simulation and Congestion Control

submitted by C. Huang, M.Eng.

in partial fulfillment of the requirements for

the degree of Doctor of Philosophy

Chair, Department of Systems and Computer Engineering

Thesis Supervisor

Thesis Supervisor

External Examiner

Carleton University

January 3, 1997

Abstract

A stochastic process is said to exhibit *long range dependence* (LRD) structure when it has a hyperbolically decaying autocorrelation function. *Self-similar* (or *fractal*) processes (both exact and asymptotic) are among those LRD processes which are widely used. Traditional traffic models, on the other hand, typically possess some form of Markovian structure and display *short range dependence* (SRD) only. Several recent papers have shown that traditional traffic models may be inadequate for modeling real traffic. Instead, self-similar stochastic processes were proposed as more accurate models of certain categories of traffic (e.g., Ethernet traffic, WAN traffic, variable-bit-rate video) which will be transported in ATM networks.

Due to the distinct differences between these two classes of models, their implications for network design and performance estimation will be significantly different. In this thesis, we will start with our work on modeling real traffic based on LRD traffic models. Then we will introduce our fast simulation technique for simulating the behavior of LRD traffic over ATM network. We will show that, some of the congestion control schemes proposed in the literature under the traditional models may fail under LRD models. In the last part, we will propose a new congestion control scheme which may work well under LRD traffic models.

Acknowledgements

I would like to thank my thesis supervisors, Prof. I.E. Lambadaris and Prof. A.R. Kaye for their guidance and encouragement throughout the course of my thesis. I appreciate their helpful suggestions and the patience they have shown to me during my study period here at Broadband Networks Laboratory.I gratefully acknowledge the funding support provided to my research work by the Telecommunications Research Institute of Ontario. Particularly I wish to thank Dr. M. Devetsikiotis and Prof. Peter Glynn of Stanford University for their valuable guidance and constructive suggestions during the research effort that led to this thesis.

To my parents

Table of Contents

List of Tables

List of Figures

List of Acronyms

ABR	Available Bit Rate
AR	Autoregressive
ARTA	Autoregressive To Arbitrary
ATM	Asynchronous Transfer Mode
BISDN	Broadband Integrated Services Digital Network
CAC	Call Admission Control
CBR	Constant Bit Rate
CDF	Cumulative Distribution Function
CI	Confidence Interval
DTMW	Double Threshold Moving Window
EFCI	Explicit Forward Congestion Indication
FARIMA	Fractional Autoregressive Integrated Moving-Average
FBM	Fractional Brownian Motion
FGN	Fractional Gaussian Noise
FIFO	First In First Out
GOP	Group of Pictures
HBC	High Boundary Check
HLBC	High Low Boundary Check
IDTMW	Inverse DTMW
IS	Importance Sampling
JPEG	Joint Picture Expert Group
LAN	Local Area Network
LBC	Low Boundary Check
LRD	Long-Range Dependence
MLE	Maximum Likelihood Estimate
MPEG	Motion Picture Experts Group
MTP	Marginal-Transformed Process
NTSC	National Television System Committee
QoS	Quality of Service
RCBR	Renegotiated Constant Bit Rate
RM	Resource Management
SRD	Short-Range Dependence
TES	Transform-Expand-Sample
VBR	Variable Bit Rate
VCI	Virtual Channel Identifier
WAN	Wide Area Network

Chapter 1

Introduction

1.1 Background

Traffic is the driving force of communications systems, and traffic models are of crucial importance for assessing their performance. In practice, stochastic models of traffic streams are relevant to network traffic engineering and performance analysis, to the extent that they are able to predict system performance measures to a reasonable degree of accuracy.

The fundamental systems, of which traffic is a major ingredient, are queueing systems. Traditional traffic models have often been devised and selected for the analytical tractability they induce in the corresponding queueing systems. However, a practitioners's confidence in a given traffic model is greatly diminished if the model is only able to crudely approximately basic statistics but cannot capture visually dominant features of empirical traffic collected from a variety of working communications systems.

While originally the validity and efficacy of models for modern high-speed network traffic were difficult to assess due to the unavailability of empirical data, recently very large sets of traffic measurements from working packet networks have become

available. More importantly, statistical analyses of these enormous traffic data sets have revealed features in measured network traffic that (i) have gone unnoticed by the teletraffic literature, (ii) suggest that from a statistical view point traditional traffic models have little in common with empirical data from modern high-speed networks, and (iii) seem to have serious implications for the design, management and control of modern telecommunications systems.

The most striking finding from these traffic data studies is that, in a statistical sense, one can sharply distinguish between empirical network traffic data and traffic generated from traditional models. Traditional traffic processes have in common that they are Markovian or, more generally, *short-range dependent* (SRD) in nature, that is, their autocorrelations decay exponentially fast [1]. On the other hand, measurements from modern networks give rise to empirical traffic processes that are generally non-Markovian in nature and exhibit *long-range dependence* (LRD). In other words, empirical traffic processes are characterized by slowly decaying autocorrelations (hyperbolic or power decay) which, in turn, result in *self-similar* or, to use a more popular term, "fractal" traffic. Specifically, studies have reported that LAN traffic [2], WAN traffic [3] and variable bit rate (VBR) video traffic [4, 5, 6] often display LRD and can be better modeled by self-similar processes.

Although fractal traffic exhibits properties that are dramatically different from those of traffic generated from traditional models, it is nevertheless possible to clearly identify the point-of-departure from traditional traffic modeling that results in fractal characteristics. It has been shown in [6] and [5] that, in addition to LRD structure, VBR video traffic also possesses low-lag correlation structures which suggests

an asymptotic self-similar model rather than exact self-similar model. Marginal probability distributions with heavy tails were reported in [5] and [6].

Although a *fractional autoregressive integrated moving-average* (FARIMA(p, d, q)) model [7] can be used to model both long term and short term autocorrelation structures at the same time, it may be difficult to obtain accurate estimates of the p and q parameters required for the generation of traces with arbitrary marginals [5]. This fact motivated us to develop modeling techniques that may capture the autocorrelation structure directly. In this thesis, we extend the work in [5] and present a unified approach which, in addition to modeling the marginal distribution of empirical records, also models directly both the SRD and LRD empirical autocorrelation structures. While here we utilize MPEG-1 compressed VBR video, the approach itself can be readily applied to other VBR video compression schemes (e.g., JPEG, MPEG-2, H.261) and other types of LRD traffic.

Briefly, we generate a background self-similar Gaussian process with both LRD and SRD explicitly incorporated. We then use a histogram-based inversion technique to generate a foreground process with the marginal distribution of the empirical data. Finally we systematically calculate and correct the asymptotic differences between the autocorrelations of the foreground processes and the autocorrelations of background processes so that the autocorrelations of the foreground processes asymptotically match that of the empirical streams. We also prove that the value of the Hurst parameter H is not affected by a large family of transformations. This class of traffic model, which defines a class of *Marginal Transformed Processes* (MTP), is quite general in the sense that it includes *Fractional Gaussian Noise* (FGN) and FARIMA(p, d, q) models as its subsets and it can have arbitrary marginal distributions.

Due to the distinct differences between LRD and SRD models, their implications for network design and performance estimation will be significantly different. However, there have been, in general, only a few analytical results reported in this area, with the notable exception of [8] and [9], where asymptotic expressions for the steady-state waiting time in single-server queues were derived by generalizing large deviation theorems to include self-similar processes. Analytical work related to this subject can also be found in [10]. In [11], a finite ATM buffer driven by a self-similar process from an infinity of on-off sources with Pareto service demands is studied. The resulting G/D/1 queueing model is mapped into a M/G/1 model where the service time is Pareto distributed with infinite variance. It is found that the buffer cell loss probability decreases with the buffer size not exponentially, as in traditional Markovian models, but algebraically.

Results in [8, 9] deal with the steady-state asymptotics for a single-server queue under FGN model. While the self-similar property captures the burstiness of traffic at all time scales, realistic ATM networks are expected to have a limiting time scale [12]. Therefore, predicted performance based on a steady-state regime may be overly pessimistic for practical applications. Furthermore, questions regarding the transient behavior, small buffer sizes, multiplexing effects, and, in general, the performance of ATM networks under LRD traffic, remain unanswered. For this purpose, analytical approaches become quickly intractable.

Given the difficulties in analysis, simulation can play an important role in the study of network performance under self-similar traffic. While several approaches have been proposed for the synthetic generation of self-similar traffic traces (e.g., Hosking's method [7], Mandelbrot's *fast fractional Gaussian noise* approach [13], *nonlinear*

chaotic maps [14]), they are, in general, efficient for generating only small numbers of relative long traces. Due to the long term dependent structure of self-similar traffic, accurate statistics can be obtained only from a large number of replications. This is especially true in ATM networks where one may want to simulate events that are *rare*, e.g., cell losses with probability $< 10^{-9}$. For this task, conventional simulation techniques can be extremely inefficient.

In this thesis, we propose a fast simulation approach based on *importance sampling* (IS) and Hosking's method in [7]. Using this approach, we simulate the transient queueing behavior of certain self-similar arrival processes, namely discrete-time FGN. We show that our transient results asymptotically approach the steady-state results in [8]. We verify experimentally the existence of a certain time scale at which the steady-state result is a good approximation for transient state. Furthermore, we apply our approach to the simulation of the multiplexing effect under both homogeneous and heterogeneous traffic sources.

We focus on the following key issues in ATM network design: the *buffering gain*, i.e., the reduction in buffer overflow probability as the buffer size increases, and the *multiplexing gain*, i.e., the reduction in buffer overflow due to statistical smoothing when multiple bursty sources are aggregated. If we define the burstiness of self-similar traffic as the Hurst parameter [15], our results indicate that, the higher the burstiness, the lower the buffering gain, as predicted by large deviation results. Our results also agree with the predictions that, compared with SRD models, self-similar models show smaller buffering gains. On the other hand, perhaps contrary to common belief, our results indicate significant gains from multiplexing. These multiplexing gains increase with the burstiness (Hurst parameter) of the self-similar traffic.

In addition to these results, we show both analytically and by simulation that when multiplexing two heterogeneous self-similar sources, the steady-state behavior will be dominated by the burstier one, as predicted by the analytical result we developed by generalizing the large deviation approach in [8]. This means that traffic streams with lower Hurst parameter values may suffer the same mean delay as traffic streams with higher Hurst parameter within a FIFO queue. In its extreme case, a starvation problem where a traffic stream has to wait indefinitely long time before it gets service may be introduced. This kind of problem can not be solved by traditional priority strategies. As pointed out in [3], for self-similar traffic, high priority traffic may block low priority traffic for quite a long time making it enter into starvation.

Furthermore, we extend the fast simulation approach to the MTP model. Here we use importance sampling techniques to efficiently estimate the probability of rare packet losses that occur when a multiplexer is fed with synthetic traffic from our self-similar VBR video model. We show that, while steady state results are useful for network planning and long time performance estimation, it may not be appropriate for estimating the QoS (Quality of Service) of a single user session.

Based on the above observations, we further investigate some congestion control schemes which may seem promising under traditional models. We show that, although some of the problems mentioned above can be solved by introducing a fair queueing approach [16, 17, 18, 19, 20, 21, 22, 23, 24], other problems remain to be addressed. Instead, we propose a new congestion control scheme which is designed to address LRD structure directly. Preliminary analytic and simulation results show that this scheme is robust in the sense that the detection probability for the congestion state is asymptotically Gaussian distributed which is totally determined by

the mean, variance and Hurst parameter of source. In addition, it can be smoothly integrated into traditional congestion control structures.

1.2 Thesis Contributions

The contributions of this thesis are:

i Prove a transform-invariant property of stationary Gaussian processes for a large class of transformations(MTP) and, based on this property, propose a new traffic modeling approach which combines direct modeling of the SRD and LRD empirical autocorrelation structures with marginal inversion and Hosking's technique. Systematically calculate and correct the asymptotic differences between the autocorrelations of background processes and the autocorrelations of the foreground processes.

ii Develop a fast simulation approach for MTP models based on importance sampling theory. Extensively simulate the buffering gain and multiplexing gain for LRD traffic streams. Develop an analytical result on the multiplexing effects of LRD traffic streams. Compare analytical results with simulation results and show the agreements.

iii Propose a new congestion control algorithm for LRD traffic and show both analytically and in simulation that it is robust to different traffic streams in terms of the setting of control parameters.

1.3 Thesis Organization

The remaining chapters of this thesis are organized as follows:

Chapter 2: Reviews the fundamentals of LRD traffic models and presents our proposed traffic modeling approach in detail. An analytical proof of the transform-invariant nature of Gaussian LRD process is provided.

Chapter 3: Describes the procedures of our fast simulation approach and investigates the simulation results of a multiplexer under LRD traffic.

Chapter 4: Reviews various congestion control schemes proposed in literature and describes our proposed congestion control algorithm. Analytical and simulation approaches are used to evaluate the algorithm.

Chapter 5: Presents conclusions and recommendations for future research.

Appendix: Extends existing Large Deviation results for a single server queue with self-similar input traffic to the cases where homogeneous or heterogeneous self-similar traffic streams are statistically multiplexed into a single server queue.

Chapter 2

LRD Traffic Models

2.1 Definitions

2.1.1 Definition of LRD Process

Let $\mathbf{X} = \{X_k : k = 1, 2, \ldots\}$ be a *covariance stationary* stochastic process with constant mean $m = E[X_k]$, finite and positive variance $\sigma^2 = E[(X_k - m)^2]$, and an autocorrelation function defined as follows:

$$r(k) = \frac{\mathrm{COV}(X_i, X_{i+k})}{\sigma^2}, \text{ for } i = 1, 2, \ldots \text{ and } k = 1, 2, \ldots \tag{2.1}$$

For each $n = 1, 2, 3, \ldots$, let

$$X_k^{(n)} = (X_{kn} + X_{kn-1} + \cdots + X_{kn-(n-1)})/n, \; k = 1, 2, 3, \ldots; \tag{2.2}$$

then the time series $\mathbf{X}^{(n)} = \{X_k^{(n)} : k = 1, 2, 3, \ldots\}$ is also a covariance stationary process. Let $r^{(n)}(k)$, $k = 1, 2, \ldots$, denote the corresponding autocorrelation function.

The process $\mathbf{X}$ is called a stationary process with long-range dependence (LRD) [1] if it satisfies

$$r(k) \sim k^{-\beta} L(k), \text{ as } k \to \infty, \tag{2.3}$$

where $0 < \beta < 1$, and $L(k)$ is slowly varying at infinity, i.e., $\lim_{t\to\infty} L(tx)/L(t) = 1$, for every $x > 0$ [2]. Furthermore, from definition (2.3) it follows that $\sum_k r(k) = \infty$. This non-summability of the correlations captures the intuition behind long-range dependence, namely that while high-lag correlations are all individually small, their cumulative effect is of importance and gives rise to features which are drastically different from those of the more conventional, i.e., short-range dependent(SRD) processes [1]. The latter are characterized by an exponential decay of the autocorrelations [1], i.e., $r(k) \sim \rho^k$,as $k \to \infty(0 < \rho < 1)$, resulting in a summable autocorrelation function $|\sum_k r(k)| < \infty$.

The process $\mathbf{X}$ is called *exactly second-order self-similar* [1] with Hurst parameter $H = 1 - \beta/2$ if it satisfies

$$r^{(n)}(k) = r(k), \text{ for all } n = 1, 2, 3, \ldots \text{ and } k = 1, 2, 3, \ldots \tag{2.4}$$

It is not difficult to show that the only solution of (2.4) is [1, 25]

$$r(k) = \delta^2(k^{2-\beta})/2 \tag{2.5}$$

where $0 < \beta < 2$ and $\delta^2(f(k)) = f(k+1) - 2f(k) + f(k-1)$. Here we are only interested in the range $0 < \beta < 1$. The process $\mathbf{X}$ is called *asymptotically second-order self-similar* [1] with Hurst parameter $H = 1 - \beta/2$, if

$$r^{(n)}(1) \;\to\; 2^{1-\beta} - 1, \text{ as } n \to \infty, \tag{2.6}$$

$$r^{(n)}(k) \;\to\; \delta^2(k^{2-\beta})/2, \text{ as } n \to \infty \ (k = 2, 3, \ldots), \tag{2.7}$$

Intuitively, one of the most striking features of self-similar(both exact and asymptotic) processes is that their aggregated processes $\mathbf{X}^{(n)}$ possess a nondegenerate correlation

structure as $n \to \infty$. In equation (2.7), for large k, differencing and differentiation are asymptotically equivalent [1], therefore we can get

$$r^{(n)}(k) \to (2-\beta)(1-\beta)k^{-\beta}/2 \tag{2.8}$$

This clearly indicates that asymptotically second-order self-similar processes are LRD processes. Detailed discussions about the definitions of self-similar processes can be found in [26, 25].

An important recent development in traffic modeling is that Leland *et al.* [2] have found that Ethernet traffic satisfies (2.4), Beran *et al.* [4] have shown that VBR video traffic satisfies (2.7) and also Paxson [3] has shown that WAN traffic exhibits LRD behavior.

2.1.2 Definition of the FGN Process

While there are numerous stochastic models which exhibit the self-similar property, two of them, namely the exactly self-similar *fractional Gaussian noise* (FGN) and the asymptotically self-similar *fractional autoregressive integrated moving-average* (FARIMA) process, are the most commonly used. FGN can be viewed as a reasonable first approximation of more complex LRD processes, since it can be derived from a special type of the central limit theorem applied to LRD processes, as shown in [27].

A fractional Gaussian noise process $\mathbf{X} = \{X_k : k = 1, 2, \ldots\}$ is a stationary Gaussian process with mean $m = E[X_k]$, variance $\sigma^2 = E[(X_k - m)^2]$, and autocorrelation function

$$r(k) = (|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H})/2, \; k = \ldots, -1, 0, 1, \ldots \tag{2.9}$$

Therefore, if $1/2 < H < 1$, FGN is exactly second-order self-similar with Hurst parameter H. Now define process $\mathbf{Z} = \{Z_k : k = 0, 1, 2, \ldots\}$ as

$$Z_k = \sum_{i=1}^{k} X_i, \text{ for } k = 1, 2, \ldots \tag{2.10}$$

Then $\mathbf{Z}$ is an stationary increment process called *fractional Brownian motion* (FBM) with mean $\alpha(k) = km$, and variance $\gamma(k) = \sigma^2 k^{2H}$ (see also [28]).

2.1.3 FARIMA(p,d,q) Process

A *FARIMA(p,d,q) process*, where p and q are non-negative integers and d is real, is defined to be a stochastic process $\mathbf{X} = \{X_k : k = 1, 2, \ldots\}$ [29, 30] with a representation given by

$$\Phi(B)\Delta^d X_k = \Theta(B)\varepsilon_k \tag{2.11}$$

where $\Phi(B) = 1 - \phi_1 B - \ldots - \phi_p B^p$ and $\Theta(B) = 1 - \theta_1 B - \ldots - \theta_q B^q$ are polynomials in the backward-shift operator $BX_k = X_{k-1}$, $\Delta = 1 - B$ denotes the differencing operator, and Δ^d is the fractional differencing operator defined by $\Delta^d = (1 - B)^d = \Sigma_k \binom{d}{k}(-B)^k$ with $\binom{d}{k}(-1)^k = \Gamma(-d + k)/(\Gamma(-d)\Gamma(k + 1))$ and $(\varepsilon_k : k = 1, 2, 3, \ldots)$ is a white noise process. It has been shown in [29] that for $d \in (-1/2, 1/2)$, $\mathbf{X}$ is stationary and invertible, and its autocorrelation satisfy $r(k) \sim ak^{2d-1}$ as $k \to \infty$, where a is a finite positive constant independent of k. Clearly X is a LRD process with Hurst parameter $d+1/2$, for all $0 < d < 1/2$. FARIMA processes are much more flexible with regard to the simultaneous modeling of the short- term and long-term behavior of a time series than a FGN process, mainly because the latter having only the three parameter μ, σ^2 and H is not capable of capturing the low-lag correlation structures encountered in practice.

2.2 Statistical Methods for Self-similar Process

2.2.1 Mean Estimation

In [25], it is shown that for a LRD process $\mathbf{X} = \{X_k, k = 1, 2, \ldots\}$, there exist the following result

$$\lim_{n\to\infty} \mathrm{var}(\frac{\sum_{i=1}^{n} X_i}{n})/[c_\gamma n^{2H-2}] = \frac{1}{H(2H-1)} \tag{2.12}$$

where c_γ is a positive constant.

Therefore, the variance of the sample mean decays to zero at a slower rate than n^{-1}.

2.2.2 Estimation of Self-Similarity

Variance-Time Plots

By equation 2.12, we have, for a self-similar process $\mathbf{X}$, that the variance of the marginal distribution of the aggregated processes $X^{(m)}$, defined by

$$X_k^{(m)} = (X_{km-m+1} + \ldots + X_{km})/m, \quad k \geq 1, \quad m = 1, 2, 3, \ldots$$

decreases linearly (for large m) in log-log plots against m. The *variance-time plots* are obtained by plotting the function $\log(\mathrm{var}(\mathrm{X^{(m)}}))$ against $\log(m)$ and by fitting a simple least squares line through the resulting points in the plane, ignoring the small values for m. An estimate $\hat{\beta}$ of β will be the absolute value of the slope of the line fit. Values of the estimate $\hat{\beta}$ between 0 and 1 suggest self-similarity, and an estimate for the degree of self-similarity is $\hat{H} = 1 - \hat{\beta}/2$.

The R/S Statistic

The R/S Statistic [31] which is based on the *Hurst effect* [15, 32, 33] is an empirical methodology proven to be more robust against the effects of marginal distributions in practice than the Variance-Time Plots. Briefly, for a given set of observations $(X_k : k = 1, 2, \ldots, n)$ with sample mean $\bar{X}(n)$ and sample variance $S^2(n)$, the rescaled adjusted range or the "R/S statistic" is given by

$$\begin{aligned} R(n)/S(n) &= [\max(0, W_1, \ldots, W_n) \\ &- \min(0, W_1, \ldots, W_n)]/S(n) \end{aligned} \tag{2.13}$$

with $W_k = (X_1 + X_2 + \ldots + X_k) - k\bar{X}(n), k = 1, 2, \ldots, n$. For self-similar processes, we have the following relation [15]:

$$E[R(n)/S(n)] \sim cn^H, \quad as\ n \longrightarrow \infty \tag{2.14}$$

Given a sample of N observations $(X_k : k = 1, 2, 3, \ldots, N)$, one subdivides the whole sample into K non-overlapping blocks which are then treated as independent replications(This assumption is approximately correct if N is very large and K is small.) and computes the rescaled adjusted range $R(i,n)/\ S(i,n)$ for each block. Here, the R/S statistic $R(i,n)/S(i,n)$ is defined as in (2.13) with i denotes the block number. Next, we plot $\log(R(i,n)/\ S(i,n))$ versus $\log(n)$ in a pox plot. This plot is the rescaled adjusted range plot. An estimate $\hat{H}$ is given by a least squares fit.

Frequency Domain MLE

While variance-time plots and pox plots of R/S are very useful tools for identifying self-similarity (in a mostly heuristic manner), the absence of any results for the limit

laws of the corresponding statistics make them inadequate when a more refined data analysis is required (e.g., confidence intervals for the degree of self-similarity H, model selection criteria, and goodness of fit tests). In contrast, a more refined data analysis is possible for maximum likelihood type estimates (MLE) and related methods based on the periodogram. In particular, for Gaussian processes $\mathbf{X} = \{X_k : k = 1, 2, \ldots\}$, Whittle's approximate MLE has been studied extensively [34, 25, 35, 36] and has been shown to be asymptotically normally distributed and asymptotically efficient for FGN or FARIMA process.

Applying this approach to empirical data, two problems of robustness due to (i) deviations from Gaussianity which both FGN and FARIMA possess, and (ii) deviations from the assumed model(FGN or FARIMA) spectrum are commonly encountered. Transforming the data so as to obtain approximately the desired marginal (normal) distribution is generally considered a viable heuristic method to overcome (i). In the presence of large data sets, a direct approach for tackling (ii) uses the method of aggregation. In [37], it is shown that there exists a central limit theorem for a large class of LRD processes. And further more, most of their aggregated processes converge weakly to FGN.

Combining Whittle's approximate MLE approach and the aggregation method give rise to the following operational procedure. For a given time series, consider the corresponding aggregated processes $\mathbf{X^{(m)}}$ with $m = 100, 200, 300, \ldots$, where the largest m-value is chosen such that the sample size of the corresponding series $\mathbf{X^{(m)}}$ is not less than about 100. For each of the aggregated series, estimate the self-similarity parameter H via Whittle's MLE. This procedure results in estimates $\hat{H}^{(m)}$ of H and corresponding, say, 95%-confidence intervals. Finally, we plot the estimates $\hat{H}^{(m)}$ of H

together with their 95%-confidence intervals versus m. Such plots will typically vary a lot for small aggregation levels, but will stabilize after a while and fluctuate around a constant value. Among the possible choices for the corresponding confidence interval, we obviously choose the one with the smallest value for m given the "stabilization" has occured, because the size of the confidence intervals increases in m (the more we aggregate, the less observations we have).

2.3 Modeling VBR Video

An important advantage of packet switched networks (e.g., ATM-based BISDN networks) is that such networks support variable bit rate (VBR) connections, thus allowing efficient statistical multiplexing of bursty traffic. Video sources (coders) generate inherently VBR traffic, however, in order to transmit video information in circuit-switched networks, the variable content of moving pictures has to be coded in constant bit rate (CBR) form, resulting in inefficient bandwidth utilization and variable picture quality.

Due to the advantages of VBR video transmission and the packet-switched nature of ATM, and given the development of highly-sophisticated compression techniques for video sources, VBR compressed video traffic is expected to become one of the main loading components in future BISDN networks. However, the high bandwidth and burstiness of VBR video traffic can make network design and management difficult to perform. Effective design and performance analysis depend on accurate modeling of the various traffic types. Among bursty traffic types, VBR video sources are arguably among the most important and demanding to model, due to their bandwidth fluctuation and autocorrelation, as well as their complex generation scheme

(coding algorithm). Numerous studies have been conducted on issues of video coding, transmission over packet networks, and related modeling and performance analysis topics, see for example [38, 39, 40, 41, 42, 43] and references within.

Traditional models based on Markovian structures (e.g., MMPP, IBP, etc.) have been widely used to statistically approximate VBR video traffic. All these models have in common an asymptotically exponential decay of the autocorrelation function and a rapidly decaying marginal distribution tail. Furthermore they lack a systematic way of simultaneously fitting both the empirical marginal distribution and the autocorrelation function.

In a series of papers (see [44] and references within), B. Melamed and colleagues at NEC USA, Inc., developed the TES (Transform-Expand-Sample) modeling technique which can capture both the marginal distribution and the autocorrelation structure of empirical records. The TES approach was used to model transmission of VBR video traffic over high-speed networks also in [45, 46]. A composite TES-based model of the "Star Wars" sequence was presented in [47].

TES processes can attain the full range of feasible lag-1 autocorrelations for a given marginal distribution[44], and can frequently match autocorrelations at higher lags. TES+ covers the positive lag-1 range [0,1] and TES- covers the negative lag-1 range [-1,0]. TES+ is defined as

$$U_i^+ = \begin{cases} U_0^+ & i = 0 \\ \langle U_{i-1}^+ + V_i \rangle & i = 1, 2, \ldots \end{cases} \tag{2.15}$$

while TES- is defined as

$$U_i^- = \begin{cases} U_i^+ & i \text{ is even} \\ 1 - U_i^+ & i \text{ is odd} \end{cases} \tag{2.16}$$

where $U_0^+ \sim U(0,1)$, and V_i is a random variable that is independent of $U_0^+, U_1^+, \ldots, U_{i-1}^+$. The notation $\langle x \rangle$ denotes modulo-1 arithmetic.

The key result is that these recursions define random variables with $U(0,1)$ marginals, and the autocorrelation structure of U_i depends only on the distribution of V_i. Therefore, the autocorrelations can be manipulated by modifying the distribution of V_i without changing the marginal distribution of U_i. However, altering the distribution of V_i typically changes the autocorrelations of U_i at all lags.

TEStool allows the user to interactively change the distribution of V_i until the autocorrelations of the input process match the desired autocorrelations. Experience is required to adjust the distribution in a systematic way. TEStool software is described in [48].

Similar to TES processes, AutoRegressive To Anything (ARTA), which was introduced by Cario and Nelson [49], is a transformation-oriented approach for modeling and generating a stationary time series $\mathbf{Y} = \{Y_i : i = 1, 2, \ldots\}$ with an arbitrary marginal distribution and autocorrelation structure specified through lag p. ARTA takes a process with a known and easily controlled autocorrelation structure, the background process $\mathbf{X} = \{X_i : i = 1, 2, \ldots\}$, and transforms it to achieve the desired marginal distribution for the input process,$\mathbf{Y}$. The target autocorrelation structure of $\mathbf{Y}$ is obtained by adjusting the autocorrelation structure of the background process. The background process is a standardized gaussian autoregressive process of order p, denoted AR(p). The critical step in constructing an ARTA process is finding the autocorrelations for the AR(p) background process $\mathbf{X}$, that yield the desired autocorrelations for the foreground ARTA process, $\mathbf{Y}$. A numerical approach to the characterization of the relationship between the AR-process autocorrelations and the

ARTA-process autocorrelations is developed in [49]. Comparing TES to ARTA, ARTA processes are guaranteed to match $p \geq 1$ autocorrelations automatically, without user intervention.

Earlier efforts in modeling video traffic have been confined to short traces of empirical records or to conference video, due to the difficulties in obtaining empirical data from realistically long sequences (Weeks of computer processing time are required at this time to generate statistics from fully compressed, full-length movies).

Recent extensive measurements of real traffic data [4], have led to the conclusion that VBR video traffic cannot be sufficiently represented by traditional models, but instead can be more accurately matched by *self-similar* (*fractal*) models [26, 2]. The crucial feature of self-similar processes is that they exhibit LRD. This is in contrast to traditional stochastic models, all of which exhibit SRD, i.e., have an autocorrelation function that decays exponentially. The serious implication for ATM network design is that conclusions based on traditional models may not be applicable under self-similar traffic. Recent studies on self-similar traffic have shown that the LRD structure may have a significant impact on queueing performance [8, 9, 50, 51].

In [5] the authors presented a detailed statistical analysis of a 2-hour long empirical VBR video trace ("Star Wars"). The authors estimated the Hurst parameter of the empirical stream, modeled the marginal distribution of the video "bandwidth" (i.e., number of bits per video frame or slice) with a combined Gamma/Pareto distribution, and generated synthetic traces by appropriately transforming a FARIMA$(0, d, 0)$ process [30] that provided the LRD behavior. However, explicit modeling of the SRD structure was left for future work.

Although a FARIMA(p, d, q) model [7] can be used to model both long term and short term correlation structures at the same time, it may be difficult to obtain accurate estimates of the p and q parameters required for the generation of traces with arbitrary marginals. This fact motivated us to develop modeling techniques that may capture the autocorrelation structure directly. In this thesis, we extend the work in [5] and present a unified approach which, in addition to modeling the marginal distribution of empirical records, also models directly both the SRD and LRD empirical autocorrelation structures. While here we utilize MPEG-1 compressed VBR video, the approach itself can be readily applied to other VBR video compression schemes (e.g., JPEG, MPEG-2, H.261). Briefly, we generate a background Gaussian process using Hosking's technique with both LRD and SRD explicitly incorporated. We then use a histogram-based inversion technique to generate a foreground process with the marginal distribution of the empirical data, and systematically calculate and correct the asymptotic difference between the autocorrelations of the foreground processes and the autocorrelations of the background processes so that the autocorrelations of the foreground processes match that of the empirical streams. We also prove that the value of the Hurst parameter H is not affected by a large family of transformations. Different from ARTA processes which can only match p autocorrelations, our approach matches the autocorrelations in an asymptotic sense so that the LRD nature is preserved.

Most often in the past, long video traffic traces have been taken from the "Star Wars" movie. In this paper, we use approximately two hours of video from the movie "Last Action Hero". The movie was initially encoded using the MPEG-1 algorithm [52, 53], with a hardware *intraframe* MPEG-1 encoder on a Sun SPARC 20 computer

Coder	MPEG-1
Duration	2 hours, 12 minutes, 36 seconds
Number of frames	238,626
Frame dimensions	320x240 pixels
Resolution	8 bits/pixel (3-band color)
Format	YUV colorspace, CCIR 601-2
Frame rate	30 per second
Slice rate	15 per frame

Table 2.1: Parameters of compressed empirical video sequence.

[54]. The movie was then decompressed and re-encoded with both intraframe and *interframe* coding, using the PVRG-MPEG 1.1 software codec [55]. A summary of the parameters of the empirical trace is given in Table 2.1.

2.3.1 Generation of FGN Trace

Clearly, generation of long synthetic traces from self-similar processes poses significant difficulties, due to their long range dependence. The earliest exact simulation method for FGN models was proposed by McLeod and Hipel [56]. That was based on Cholesky decomposition of correlation matrices which is efficient only for short traces. Hosking [7] improved the approach in [56] by employing the Levinson-Durbin algorithm to make it appropriate for longer time traces. We briefly describe Hosking's procedure in the following paragraphs.

For a FGN process $\mathbf{X}$ with $m = 0$, the conditional mean and variance of X_k, given the past values $x_{k-1}, x_{k-2}, \ldots, x_1$, may be written as [57]

$$m_k = \mathrm{E}(X_k | x_{k-1}, x_{k-2}, \ldots, x_1) = \sum_{j=2}^{k} \phi_{kj} x_{k-j+1} \quad \text{for } k \geq 2 \tag{2.17}$$

$$v_k = \mathrm{Var}(X_k | x_{k-1}, x_{k-2}, \ldots, x_1) = \sigma^2 \prod_{j=2}^{k} (1 - \phi_{jj}^2) \quad \text{for } k \geq 2 \tag{2.18}$$

Here ϕ_{jj} is the jth partial correlation coefficient of $\{X_k\}$ and the ϕ_{kj} are partial linear regression coefficients. For simulating a sample $\{x_1, x_1, \ldots, x_{n-1}\}$ of size n from a FGN process, [7] describes the following algorithm:

1. Generate a starting value x_1 from a Gaussian distribution $N(0, \sigma^2)$. Set $N_1 = 0$, $D_1 = 1$,$v_1 = \sigma^2$.

2. Set $N_2 = r(2)$,$D_2 = D_1$,$\phi_{22} = \frac{N_2}{D_2}$,$m_2 = \phi_{22}x_1$ and $v_2 = (1-\phi_{22}^2)v_1$, generate a value x_2 from a Gaussian distribution $N(m_2, v_2)$.

3. For $k = 3, \ldots, n-1$, calculate ϕ_{kj}, $j = 2, \ldots, k$, recursively via the equations

$$N_k = r(k-1) - \sum_{j=2}^{k-1} \phi_{k-1,j} r(k-j) \tag{2.19}$$

$$D_k = D_{k-1} - N_{k-1}^2 / D_{k-1} \tag{2.20}$$

$$\phi_{kk} = N_k / D_k \tag{2.21}$$

$$\phi_{kj} = \phi_{k-1,j} - \phi_{kk}\phi_{k-1,k-j+1} \quad j = 2, \ldots, k-1 \tag{2.22}$$

Calculate $m_k = \sum_{j=2}^{k} \phi_{kj} x_{k-j+1}$ and $v_k = (1-\phi_{kk}^2)v_{k-1}$. Generate x_k from the Gaussian distribution $N(m_k, v_k)$.

The above method is applicable to any Gaussian process as long as the correlation function $r(k)$ is known.

2.3.2 Generation of a Self-Similar Process with an Arbitrary Marginal Distribution

Let X_k be a Gaussian process with zero mean, unit variance and autocorrelation function $r(k)$. Let $F_X(x)$ be its marginal cumulative probability function. Let $F_Y(y)$ be a marginal cumulative probability function corresponding to a process Y_k. Then we can generate the process Y_k with the desired marginal cumulative probability function $F_Y(y)$ from the process X_k by using the following transformation [44, 5]:

$$Y_k = h(X_k) = F_Y^{-1}(F_X(X_k)) \quad k = 1, 2, \ldots \tag{2.23}$$

In real modeling procedures, $F_Y(y)$ can be obtained either by modeling an empirical distribution using parametric mathematical functions or, as we do in our approach, by inverting the empirical distribution directly. An important issue, however, regarding this approach is that if the process $\mathbf{X}$ is a self-similar Gaussian process with Hurst parameter H, then the nature of the process $\mathbf{Y}$ may not be known. In the following, we show that, under general conditions, the process $\mathbf{Y}$ will be a self-similar process having the same Hurst parameter with process $\mathbf{X}$.

Theorem 1: Let $\mathbf{X} = \{X_i, i = 0, 1, \ldots\}$ be a zero mean, unit variance stationary Gaussian process defined on a probability space $(\Omega, \mathcal{F}, P)$ and $h : \mathbf{R} \longmapsto \mathbf{R}$ be a Borel measurable function. If $h^2(X)$ is integrable with respect to P ,$\mathrm{E}(h(X)X) \neq 0$, and $r(k) \to 0$, then $\lim_{k\to\infty} r_h(k)/r(k) = [\mathrm{E}(h(X)X)]^2/\mathrm{VAR}(h(X))$, where $r(k)$ and $r_h(k)$ are the autocorrelation functions of process $\mathbf{X}$ and $h(\mathbf{X})$ respectively.

Proof: Since $h^2(X)$ is integrable with respect to P, it follows by a straightforward application of the Schwartz inequality that the autocovariance of $\mathbf{Y}$ is finite or

$$h(x_i)h(x_{i+k})\frac{1}{2\pi\sqrt{1-r^2(k)}}\exp\{-\frac{x_i^2 - 2r(k)x_ix_{i+k} + x_{i+k}^2}{2(1-r^2(k))}\} \tag{2.24}$$

is integrable with respect to Lebesgue measure.

Due to the symmetry of the zero mean Gaussian probability distribution, and the fact that $h^2(-X)$ is also integrable with respect to P we can easily conclude that the function

$$h(x_i)h(x_{i+k})\frac{1}{2\pi\sqrt{1-r^2(k)}}\exp\{-\frac{x_i^2 + 2r(k)x_ix_{i+k} + x_{i+k}^2}{2(1-r^2(k))}\} \tag{2.25}$$

is integrable with respect to Lebesgue measure.

Furthermore, since for any real numbers x and y we have $\exp(|xy|) < \exp(xy) + \exp(-xy)$ the function

$$h(x_i)h(x_{i+k})\frac{1}{2\pi\sqrt{1-r^2(k)}}\exp\{-\frac{x_i^2 - |2r(k)x_ix_{i+k}| + x_{i+k}^2}{2(1-r^2(k))}\} \tag{2.26}$$

is integrable with respect to Lebesgue measure. By the expansion of exponential function, we have

$$\exp\{|\frac{r(k)}{1-r^2(k)}x_ix_{i+k}|\} = \sum_{n=0}^{\infty}[|\frac{r(k)}{1-r^2(k)}x_ix_{i+k}|]^n/n! \tag{2.27}$$

Therefore the function

$$\sum_{n=0}^{\infty} h(x_i)h(x_{i+k})\frac{1}{2\pi\sqrt{1-r^2(k)}}[|\frac{r(k)}{1-r^2(k)}x_ix_{i+k}|]^n/n!\exp\{-\frac{x_i^2+x_{i+k}^2}{2(1-r^2(k))}\} \tag{2.28}$$

is integrable with respect to Lebesgue measure, that is

$$\int\int\{\sum_{n=0}^{\infty}|h(x_i)h(x_{i+k})|\frac{1}{2\pi\sqrt{1-r^2(k)}}[|\frac{r(k)}{1-r^2(k)}x_ix_{i+k}|]^n/n!\exp\{-\frac{x_i^2+x_{i+k}^2}{2(1-r^2(k))}\}\}dx_idx_{i+k} < \infty \tag{2.29}$$

Furthermore the function

$$h(x_i)h(x_{i+k})\frac{1}{2\pi\sqrt{1-r^2(k)}}[\frac{r(k)}{1-r^2(k)}x_ix_{i+k}]^n/n!\exp\{-\frac{x_i^2+x_{i+k}^2}{2(1-r^2(k))}\} \tag{2.30}$$

is also integrable with respect to Lebesgue measure for $\forall n \geq 0$.

By definition, we can calculate the autocovariance of $\mathbf{Y}$ as follows:

$$\begin{aligned}
&\mathrm{COV}(Y_i, Y_{i+k}) = \\
&= \int\int h(x_i)h(x_{i+k})\frac{1}{2\pi\sqrt{1-r^2(k)}}\exp\{-\frac{x_i^2 - 2r(k)x_ix_{i+k} + x_{i+k}^2}{2(1-r^2(k))}\}dx_idx_{i+k} - [E(h(X))]^2 \\
&= \int\int \frac{1}{2\pi\sqrt{1-r^2(k)}}h(x_i)h(x_{i+k})\{\sum_{n=0}^{\infty}[\frac{r(k)}{1-r^2(k)}x_ix_{i+k}]^n/n!\}\exp\{-\frac{x_i^2+x_{i+k}^2}{2(1-r^2(k))}\}dx_idx_{i+k} \\
&\quad -[E(h(X))]^2
\end{aligned} \tag{2.31}$$

By equation (2.29) and the dominated convergence theorem, for each $k > 0$, we can write

$$
\begin{aligned}
&COV(Y_i, Y_{i+k}) = \\
&= \frac{1}{2\pi\sqrt{1-r^2(k)}} \sum_{n=0}^{\infty} \int\int h(x_i)h(x_{i+k})[\frac{r(k)}{1-r^2(k)} x_i x_{i+k}]^n / n!] \exp\{-\frac{x_i^2 + x_{i+k}^2}{2(1-r^2(k))}\} dx_i dx_{i+k} \\
&\quad -[E(h(X))]^2 \\
&= \frac{r(k)}{2\pi[1-r^2(k)]^{3/2}} \{\int h(x_i) x_i \exp\{-\frac{x_i^2}{2(1-r^2(k))}\} dx_i\}^2 + \frac{r^2(k)}{[1-r^2(k)]^{5/2}} R(r(k)) \\
&\quad + \frac{1}{2\pi[1-r^2(k)]^{1/2}} \{\int h(x_i) \exp\{-\frac{x_i^2}{2(1-r^2(k))}\} dx_i\}^2 - [E(h(X))]^2
\end{aligned} \tag{2.32}
$$

where we define $R(r(k))$ as follows:

$$
\begin{aligned}
&R(r(k)) = \\
&= \frac{1}{2\pi} \sum_{n=2}^{\infty} \int\int h(x_i)h(x_{i+k})[\frac{r(k)}{1-r^2(k)}]^{n-2} [x_i x_{i+k}]^n / n!] \exp\{-\frac{x_i^2 + x_{i+k}^2}{2(1-r^2(k))}\} dx_i dx_{i+k}
\end{aligned} \tag{2.33}
$$

It is easy to see that

$$
\exp\{-\frac{x_i^2 + x_{i+k}^2}{2(1-r^2(k))}\} \leq \exp\{-\frac{x_i^2 + x_{i+k}^2}{2}\} \tag{2.34}
$$

Therefore, by using the Cauchy-Schwarz inequality, we have

$$
\begin{aligned}
&|R(r(k))| \leq \\
&\leq \frac{1}{2\pi} \sum_{n=2}^{\infty} \int\int |h(x_i)h(x_{i+k})|[|\frac{r(k)}{1-r^2(k)}|]^{n-2} [|x_i x_{i+k}|]^n / n!] \exp\{-\frac{x_i^2 + x_{i+k}^2}{2}\} dx_i dx_{i+k} \\
&= 2\pi \sum_{n=2}^{\infty} [|\frac{r(k)}{1-r^2(k)}|]^{n-2} / n! \{E[|h(X)||X|^n]\}^2 \\
&\leq 2\pi \sum_{n=2}^{\infty} [|\frac{r(k)}{1-r^2(k)}|]^{n-2} / n! E(h^2(X)) E(X^{2n})
\end{aligned} \tag{2.35}
$$

Given $r(k) \to 0$, there $\exists K$ such that, for $\forall k > K$, we have

$$
|\frac{r(k)}{1-r^2(k)}| \leq \frac{1}{4} \tag{2.36}
$$

Therefore, by equation 2.35, we have

$$
\begin{aligned}
&|R(r(k))| \leq 2\pi E(h^2(X)) \sum_{n=2}^{\infty} [1/4]^{n-2} / n! E(X^{2n}) \\
&\leq 8\pi E(h^2(X)) \sum_{n=0}^{\infty} 1/n! E((X/2)^{2n}) \\
&= 8\pi E(h^2(X)) E(\exp\{X^2/4\}) \\
&= 8\sqrt{2}\pi E(h^2(X)) < \infty \text{ for } k > K
\end{aligned} \tag{2.37}
$$

Now by equation 2.32 and 2.37, we have

$$\begin{aligned}\lim_{k\to\infty}\frac{r_h(k)}{r(k)} &= \lim_{k\to\infty}\frac{COV(Y_i,Y_{i+k})}{r(k)VAR(Y_i)}\\ &= \frac{1}{2\pi}\frac{\{\int h(x_i)x_i\exp\{-\frac{x_i^2}{2}\}dx_i\}^2}{VAR(h(X_i))}\\ &= \frac{[E(h(X_i)X_i)]^2}{VAR(h(X_i))}\\ &= a \end{aligned} \tag{2.38}$$

where $a = \frac{[E(h(X_i)X_i)]^2}{VAR(h(X_i))}$.Because $E(X) = 0$, we have $E[h(X)X] = \mathrm{E}[(h(X)-\mathrm{E}h(X))X]$. By using the Cauchy-Schwarz inequality it follows that $0 < a \leq 1$.□

A close but different conclusion proved by Dr. M. Taqqu can be found in Proposition 3.1 of [37] which has shown that, under similar conditions as the above Theorem 1, $VAR(\sum_{i=1}^{N} h(X_i))$ is regularly varying with exponent $2H$,$1/2 < H < 1$, as $N \to \infty$, if and only if $r(k)$ is regularly varying with exponent $-D = 2H - 2$ as $k \to \infty$. For more about this conclusion, please refer to [37].

Under the conditions of Theorem 1, if process $\mathbf{X}$ is a LRD process with Hurst parameter H, process $h(\mathbf{X})$ will also be a LRD process with the same Hurst parameter H. In the following parts, we will call the class of processes which satisfy Theorem 1 the class of *Marginal-Transformed Processes* (MTP).

The empirical distribution function and the transform $h(.)$ corresponding to the trace of "Last Action Hero" is shown in Fig. 2.1 and 2.2.

2.3.3 Generation of a Process with both LRD and SRD

In the following parts, we describe our approach to modeling VBR video with both LRD and SRD in four steps:

Step 1: Estimation of the Hurst parameter H:

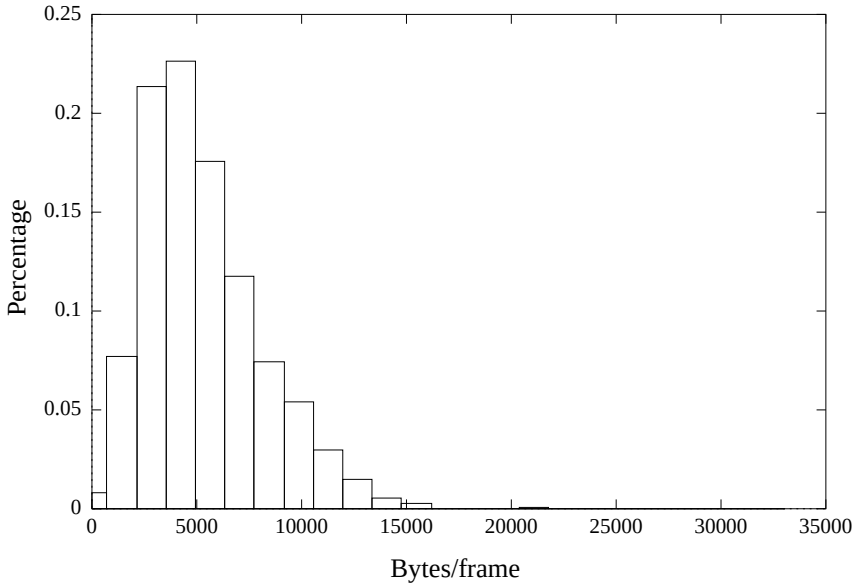

Figure 2.1: Empirical distribution function for "Last Action Hero".

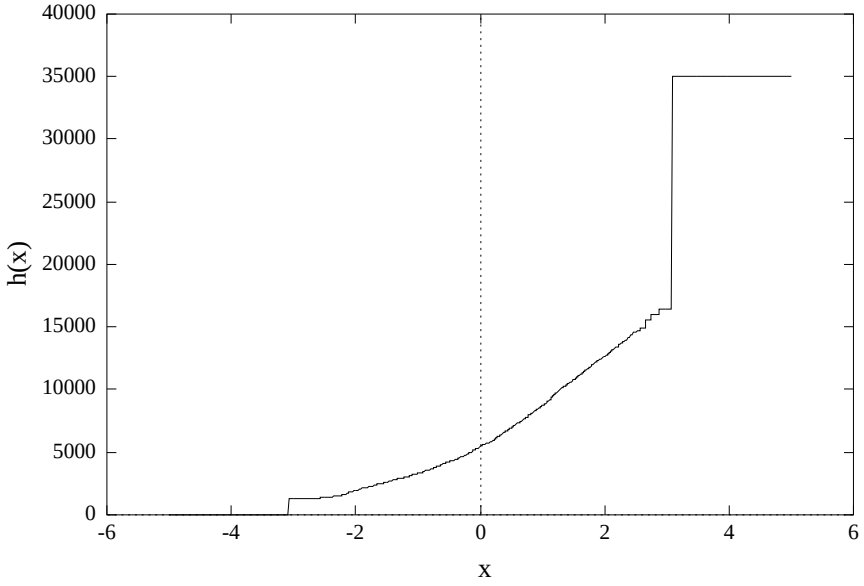

Figure 2.2: Transform function $h(X)$ that converts a normal distribution to the marginal distribution of the "Last Action Hero" trace.

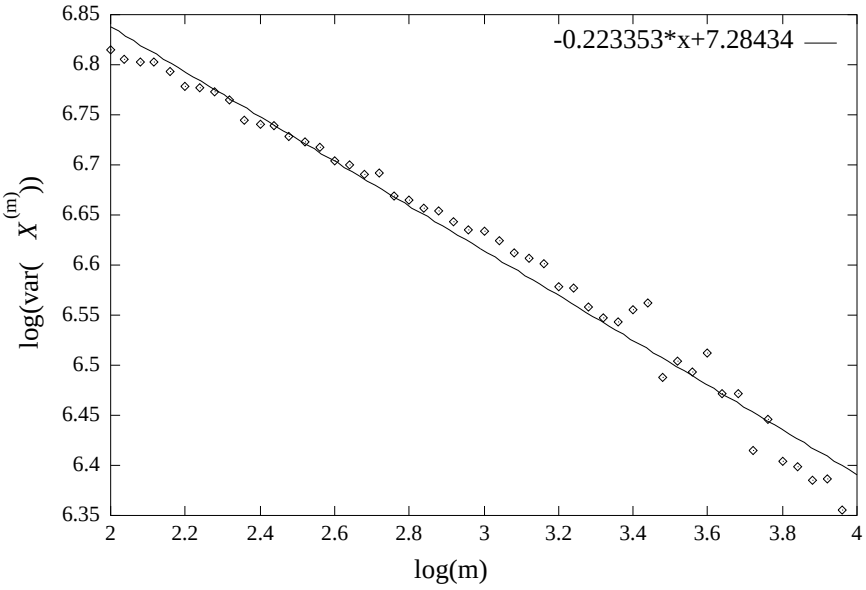

Figure 2.3: Variance-time plot for "Last Action Hero".

Fig.2.3 shows the variance-time plot for the empirical trace of "Last Action Hero". Logarithms are taken to base 10. An estimate for the corresponding Hurst parameter is $\hat{H} = 0.89$.

The R/S plot is shown in Fig. 2.4. An $\hat{H} = 0.92$ was determined.

Fig. 2.5 depicts the frequency domain MLE result. At aggregation level $m = 700$, an estimate of Hurst parameter H results in $\hat{H} = 0.95 \pm 0.07$.

Combining the results of the above three approaches, we decided to set $\hat{H} = 0.9$ and $\hat{\beta} = 0.2$ for the empirical trace of "Last Action Hero".

Step 2: Modeling the autocorrelation function:

The autocorrelation resulting from the actual empirical trace of "Last Action Hero" movie is shown in Fig. 2.6. Upon inspection of the plot it is evident that a "knee" around lag 60 to 80 exists. For lags less than the "knee" we observe that the autocorrelation decreases relatively fast thus indicating a short term dependence. When

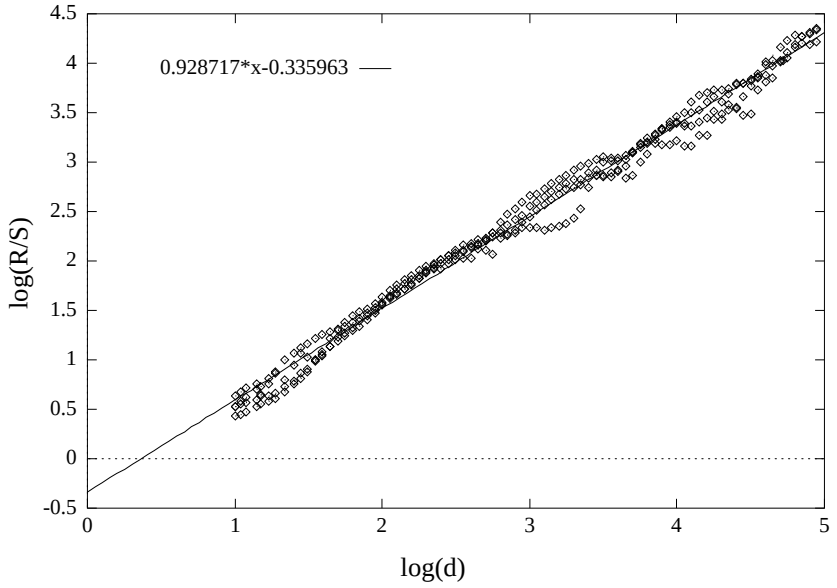

Figure 2.4: Pox diagram of R/S for "Last Action hero".

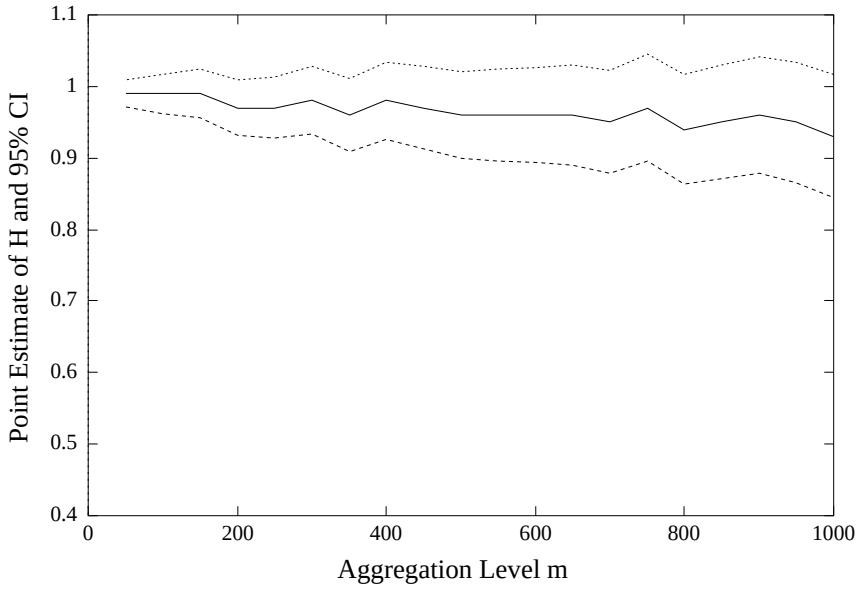

Figure 2.5: Frequency domain MLE estimate $\hat{H}^{(m)}$ of H (solid line) and 95%-confidence intervals (dotted lines), as a function of the aggregation level m for empirical video trace.

the lag is larger than the "knee", we can observe a slowly decreasing autocorrelation indicating long range dependence. The rapidly decaying part of the autocorrelation can be approximated by superimposing a number of decreasing exponentials of the form $\exp(-\lambda_i k)$ with different rates λ_i. Furthermore the part corresponding to long range dependence can be approximated by $Lk^{-\beta}$, where L is a constant. We can now write the following:

$$\begin{aligned} r(k) &= Lk^{-\beta} I(k \geq K_t) \\ &+ \sum_{i=1}^{j} w_i \exp(-\lambda_i k) I(k < K_t), k = 1, 2, \ldots \\ \sum_{i=1}^{j} w_i &= 1 \\ LK_t^{-\beta} &= \sum_{i=1}^{j} w_i \exp(-\lambda_i K_t) \end{aligned} \tag{2.39}$$

where K_t is the lag value corresponding to the "knee", $I(.)$ is the indicator function. In our case we used one exponential for modeling the SRD. To find the exact knee value, we follow the following procedures:

First, we set an error range ϵ. Starting from lag 0, we apply the least square fitting recursively to the autocorrelations with increasing lags until the error is larger than ϵ. The resulting exponential function is the model of short term autocorrelations. Similarly, starting from the largest lag, we recursively fit the model for LRD parts with decreasing lag by fixing the measured parameter β until the error is larger than ϵ. Then the lag value corresponding to the crossing of the two models is the knee value. Such a fit is shown in Fig. 2.7. Finally we obtained the following expression

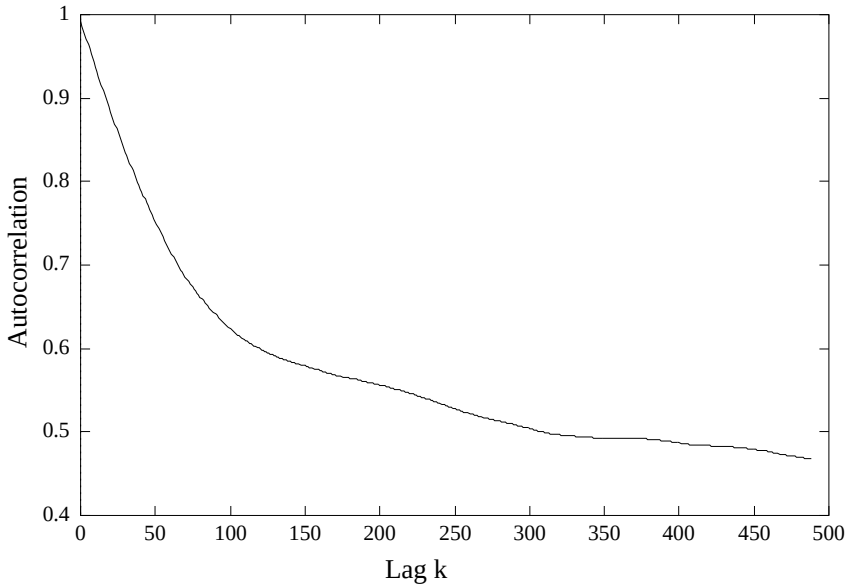

Figure 2.6: The estimated autocorrelation function of "Last Action Hero".

for the autocorrelation:

$$\hat{r}(k) = \exp(-0.00565k)I(k < K_t) + 1.59k^{-0.2}I(k \geq K_t) \tag{2.40}$$

As will be illustrated later on by simulation experiments, the exponential component was necessary since the polynomial component decays too fast in the early lags.

It should be noted that the $\hat{r}(k)$ obtained through the above approach might not be a well-defined autocorrelation function. In [58], it is shown that $\hat{r}(k)$ is a well-defined autocorrelation function if and only if $\hat{r}(k)$ forms a positive definite sequence. While this may be difficult to check, we find that,in practice, if the conditions in equation 2.39 are satisfied, $\hat{r}(k)$ will typically be a well-defined autocorrelation function.

Step 3: Calculation of the "attenuation" factor a:

Let process $\mathbf{X}$ be a stationary Gaussian process with zero mean, unit variance and autocorrelation function $\hat{r}(k)$. Define process $\mathbf{Y}$ as equation (2.23).Let $r_h(k)$ be

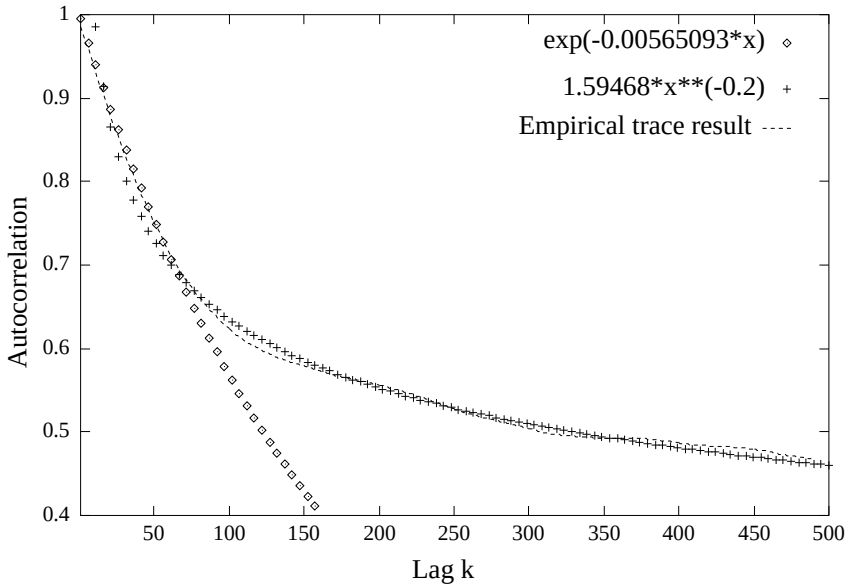

Figure 2.7: Autocorrelation fitting result.

the autocorrelation function of the processes $\mathbf{Y}$. By Theorem 1, we have $r_h(k) = ar(k)$, as $k \longrightarrow \infty$, where $a = \frac{[E(h(X_i)X_i)]^2}{VAR(h(X_i))} \leq 1$. We call a the "attenuation" factor. By this formula, a simulation results $a = 0.94$.

Step 4: Generation of a process with the desired autocorrelation:

Let $r(k) = \hat{r}(k)/a$, for $k \geq K_t$. Then for the short term part, we solve the following equation to obtain the rate λ:

$$\exp(-\lambda K_t) = 1/a \tag{2.41}$$

and we let $r(k) = \exp(-\lambda k)\hat{r}(k) \quad for\ k < K_t$. We decided to set $K_t = 60$ based on the intersection point of the two fitting curves. We then generate process $\mathbf{X}$ with the new autocorrelation function $r(k)$ using Hosking's method and the process $\mathbf{Y}$ using equation (2.23).

To compare our model with the original video trace, we generate a synthetic

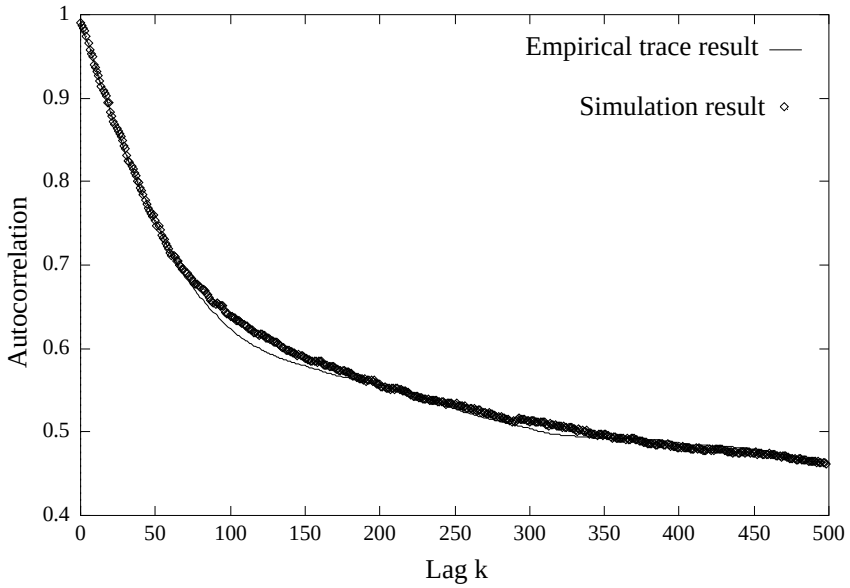

Figure 2.8: Autocorrelation of the empirical trace and the final simulated process.

trace that has the same length as the original video data. The final autocorrelation result of the synthetic data trace is shown together with the empirical autocorrelation of Fig. 2.8, indicating a satisfactory match.

In Fig. 2.9, the frequency domain MLE result of the synthetic trace is shown. At aggregation level $m = 450$, an estimate for H is $\hat{H} = 0.91 \pm 0.057$ consistent with the prediction of theorem 1(i.e. Processes $\mathbf{X}$ and $\mathbf{Y}$ have the same Hurst parameter).

To compare their marginal distributions, we show in Fig. 2.10 the Q-Q plot of the two data sets, which indicates a good agreement.

2.3.4 Modeling VBR Video with Interframe Compression

In this section, we generalize our approach to the modeling of VBR video with both intraframe and interframe compression. The codec we used is the PVRG-MPEG 1.1 software codec based on the Santa Clara 1991 draft of MPEG-1 [55].

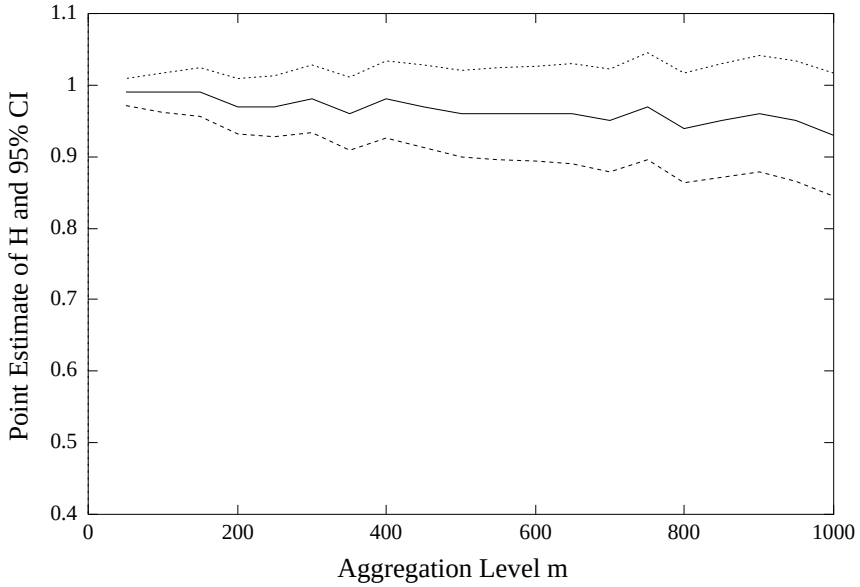

Figure 2.9: Frequency domain MLE estimate $\hat{H}^{(m)}$ of H (solid line) and 95%-confidence intervals (dotted lines), as a function of the aggregation level m for synthetic data trace.

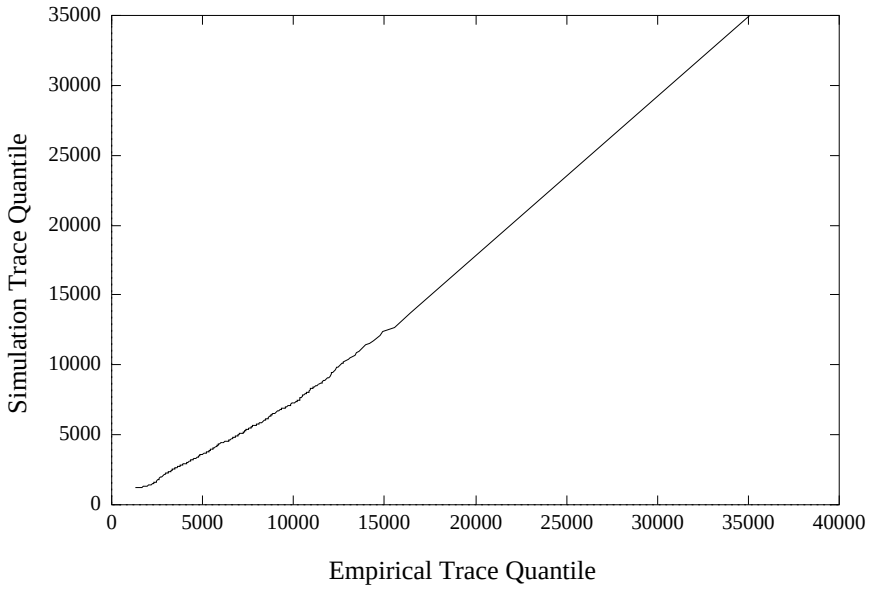

Figure 2.10: Q-Q plot comparing the marginal distributions of the simulation process and the empirical trace.

The MPEG-1 coder [53] consists of five stages: a motion compensation stage, a transformation stage, a lossy quantization stage, and two lossless coding stages. The motion compensation stage subtracts the current image from the shifted view of the previous image if they are both alike. The transform concentrates the information energy into the first few transform coefficients, the quantizer causes a controlled loss of information, and the two coding stages further compress the data closer to symbol entropy.

A MPEG-1 sequence consists of three separate parts: a series of intraframes (I frames), which are image frames coded individually without any temporal prediction; a series of forward predicted frames (P frames), interspersed between these I frames; and bidirectionally predicted frames (B frames) interspersed between the forward predicted frames and the intraframes. A typical frame sequence in a GOP (group of pictures) is as follows:

I B B P B B P B B P B B I ...

Our approach to modeling interframe-encoded MPEG-1 VBR video is to generate a single stationary background process $\mathbf{X}$ with both SRD and LRD structures and then generate the foreground process using three different transforms $h_I(X)$, $h_B(X)$ and $h_P(X)$ based on the histograms of I, B and P frames, respectively, according to above frame sequence structure. The PVRG-MPEG 1.1 software codec used in our experiments produced video traffic in which I frames appear periodically once every 12 frames.

We model the composite I-B-P video traffic as follows:

Step 1: Isolate I frames only and model the I-frames process according to the previous sections. Assume $r_h(k)$ be its final autocorrelation function;

Step 2: Rescale the estimated autocorrelation of the I frames :

$$r(k) = r_I(k/K_I) \tag{2.42}$$

where, $r_I(k)$ is the autocorrelation and $K_I = 12$ is the period of I frames;

Step 3: By using Hosking's technique employing the autocorrelation $r(k)$ in the last step we generate the process $\mathbf{X}$;

Step 4: We then generate process $\mathbf{Y}$ using equation 2.23 where function $h(.)$ is replaced by functions $h_I(.)$,$h_B(.)$, $h_P(.)$ iteratively according to the GOP structure.

The similarity between the synthetic and real data trace is evaluated by means of the corresponding estimates of autocorrelation functions and marginal distribution histograms. Figures 2.11, 2.12, and 2.13 show the foreground autocorrelation of the synthetic trace in comparison to the autocorrelation of the original empirical trace from "Last Action Hero". Figure 2.14 compares the marginal distributions of the model process versus the empirical data trace, using a Q-Q plot. The agreement shown in the figures above supports the use of our approach for modeling complex traffic streams.

We have developed a software package that establishes an automatic search for the best background autocorrelation structure and the calculation of the attenuation factor a.

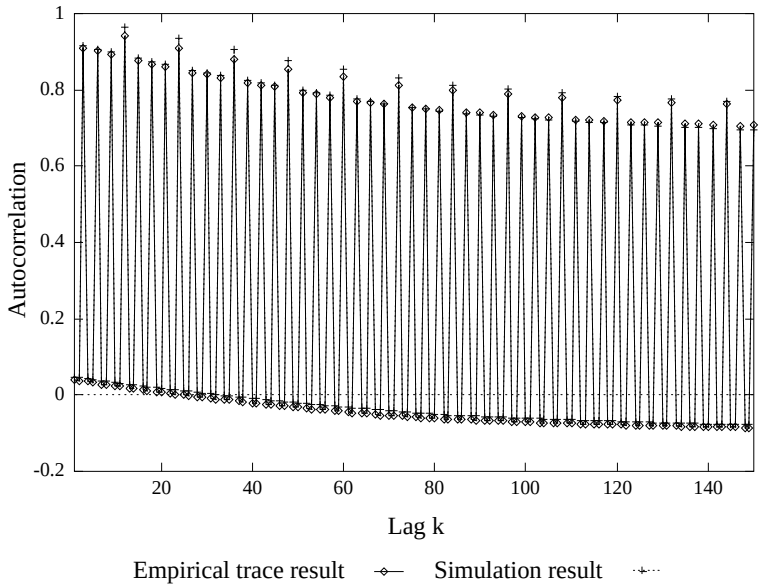

Figure 2.11: Comparison of autocorrelations of simulation process and empirical trace (lags 1 to 150).

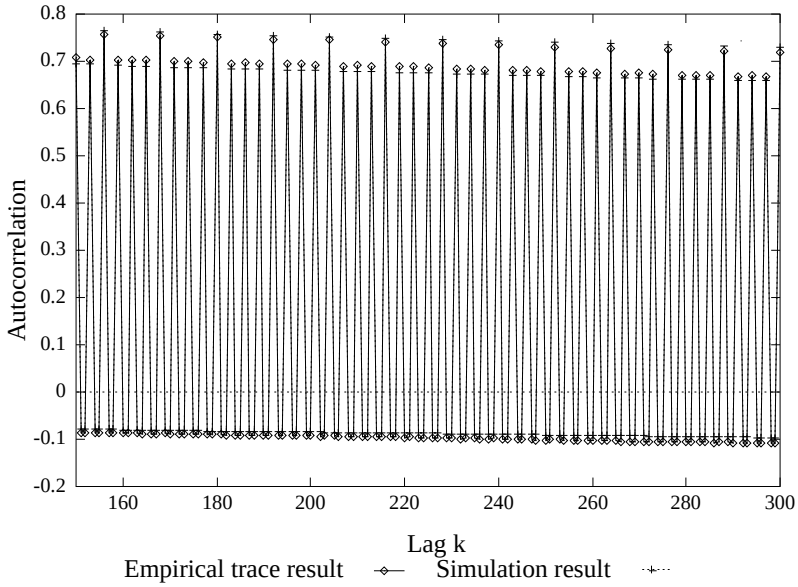

Figure 2.12: Comparison of autocorrelations of simulation process and empirical trace (lags 151 to 300).

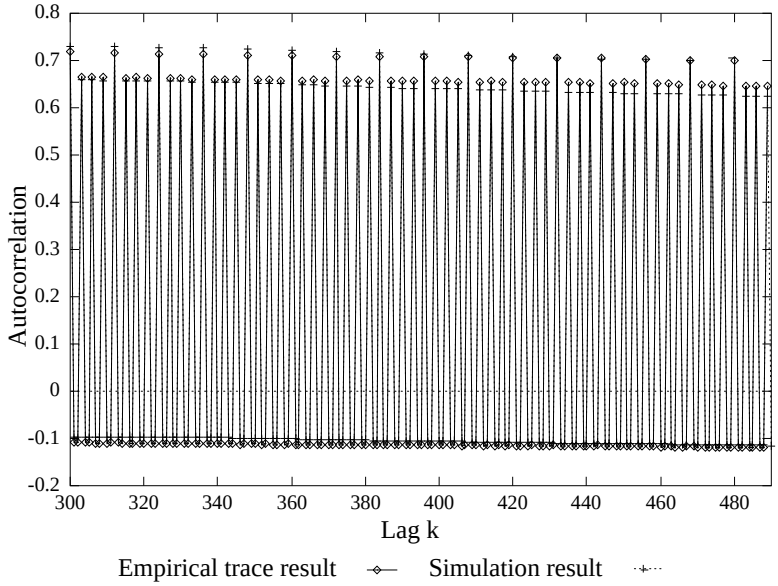

Figure 2.13: Comparison of autocorrelations of simulation process and empirical trace (lags 301 to 490).

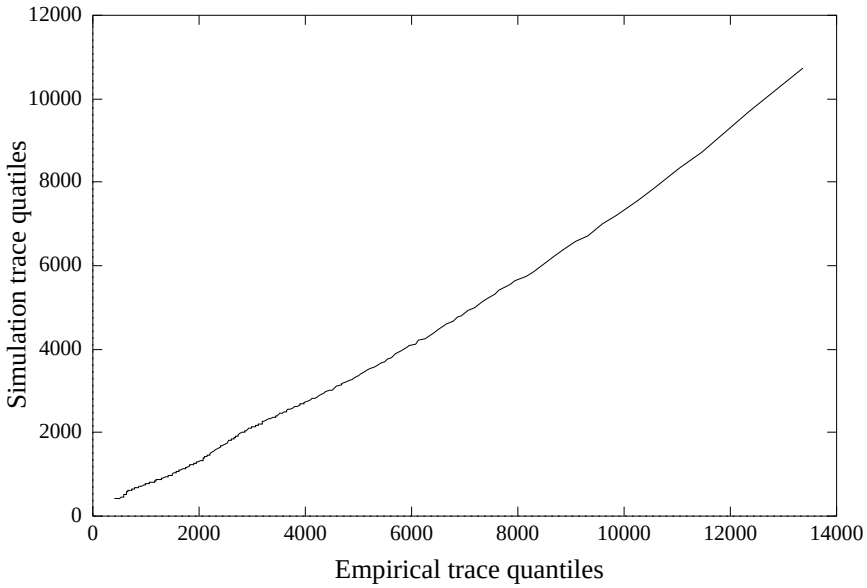

Figure 2.14: Q-Q plot comparing the marginal distributions of the simulation process and the empirical trace.

Chapter 3

Importance Sampling Techniques for MTP

3.1 Lindley Equation and Large Deviation Result

Consider a slotted-time single server queue with deterministic service rate μ and an arrival process $\mathbf{X} = \{X_k, i = 1, 2, \ldots\}$, with X_k representing the number of arriving cells within the kth time slot. Here, without loss of generality, we assume X_k can take any real value. Letting Q_k denote the size of the queue at time $k = 0, 1, \ldots$, we have the following Lindley equation [59]:

$$Q_k = \langle Q_{k-1} + X_k - \mu \rangle^+ = \langle Q_{k-1} + Y_k \rangle^+, \text{ for } k = 1, 2, \ldots \tag{3.1}$$

where we refer to the process $\mathbf{Y} = \{Y_k : Y_k = X_k - \mu, k = 1, \ldots\}$ as the *netput* process and $\langle x \rangle^+$ denotes $\max\{0, x\}$. If we define the *total netput process* $\mathbf{W}$ as $\{W_k : W_k = \sum_{i=1}^{k} Y_i,\ k = 1, 2, \ldots\ W_0 = 0\}$ and assume $Q_0 = 0$, we will have the following Reich formula [60]

$$Q_k = \max_{0 \le i \le k} (W_k - W_i) \text{ for } k = 1, 2, \ldots \tag{3.2}$$

Now we assume that the process $\mathbf{X}$ is stationary and ergodic. We can then claim that the limiting process of Q_k as $k \longrightarrow \infty$ exists. This follows from a well-known result that if $Q_0 = 0$ and the total netput process $\mathbf{W}$ has stationary increments with $\lim_{k\to\infty} W_k = -\infty$ a.s., then Q_i converges in distribution to a proper random variable, say Q_∞ [61]. In this case, since $\mathbf{X}$ is stationary and ergodic, it follows from the Ergodic Theorem [62] that $\lim_{i\to\infty} W_k/k = E[X_i - \mu] < 0$ a.s. if utilization is smaller than 1, which in turn implies that $\lim_{k\to\infty} W_i = -\infty$ a.s.. The above arguments follow [63]. The difference is that here the process $\mathbf{X}$ is a general stationary and ergodic process rather than FGN. Clearly the above conclusion can be generalized to the cases where Q_0 are finite.

If $\mathbf{X}$ is a FGN process with zero mean, unit variance and Hurst parameter H, $\mathbf{X}$ will be a stationary and ergodic Gaussian process [9, 35]. Duffield *et al.* [8] have shown the following steady-state, large deviation result:

$$\lim_{b\to\infty} b^{-2(1-H)} \log \Pr(Q_\infty > b) = -c^{-2(1-H)}(c+\mu)^2/2 \tag{3.3}$$

where $c = \mu/H - \mu, \mu > 0$. Therefore, in contrast to traditional SRD models, the steady-state queueing distribution decays asymptotically in a Weibull fashion rather than exponentially. Thus the performance (in terms of buffer overflow probability) predicted under FGN may be worse compared to the one derived using traditional models assumptions. This intuition stems from the fact that FGN models capture the burstiness of traffic at all time scales, contrary to traditional models which only capture burstiness at certain time scales. Under traditional (SRD) models, the burstiness at different time scales is typically exhibited by using a complex hierarchical structure which makes theoretical analysis and simulation increasingly difficult due to the large

class of system parameters to be selected. In contrast, self-similar models capture long range dependence in a *parsimonious manner* which makes them very attractive from the standpoint of modeling realistic LRD traffic [2].

Results in [8] deal with the steady-state asymptotics for a single-server queue under FGN. While the self-similar property captures the burstiness of traffic at all time scales, realistic ATM networks are expected to have a limiting time scale. Therefore, predicted performance based on a steady-state regime may not be accurate enough for practical applications. Furthermore, questions regarding the transient behavior, small buffer sizes, multiplexing effects, and, in general, the performance of ATM networks under LRD traffic, remain unanswered.

Due to the recursive nature of Hosking's method, the computational effort required for generating self-similar traffic increases approximately as $O(n^2)$ with the length of the trace, n. Therefore, although Hosking's method has some improvements over McLeod's method, it still requires a large computational effort, especially for long traces.

Given the computational cost of trace generation, the number of replications required becomes crucial, especially when studying ATM networks where one may want to simulate events that are *rare*, e.g., cell losses with probability $< 10^{-9}$, or extremely long cell waiting times. In such cases, using conventional Monte Carlo simulation, we may need to generate millions of traces using Hosking's method, which is practically infeasible. In the following, we develop a fast simulation approach based on importance sampling, that makes Hosking's method applicable to QoS evaluation in ATM networks.

3.2 Importance Sampling Theory

Let U be a random variable that has a probability density function $p(u)$ and consider estimating the probability $P = \Pr(U \in A)$ for some set A, we can write

$$P = \int_{-\infty}^{\infty} I_A(t)p(t)dt = \mathrm{E}_p[I_A(U)] \tag{3.4}$$

where $I_A(\cdot)$ is the indicator function of event A. Assume that $p'(u)$ is another density function. Assuming that $p(u) = 0$ whenever $p'(u) = 0$ (*absolute continuity* condition), we have

$$P = \int_{-\infty}^{\infty} I_A(t)\frac{p(t)}{p'(t)}p'(t)dx = \mathrm{E}_{p'}[I_A(U)\frac{p(U)}{p'(U)}] = \mathrm{E}_{p'}[I_A(U)L(U)] \tag{3.5}$$

where $L(u) = p(u)/p'(u)$ is a *likelihood ratio* (*weight function*) and the notation p' denotes sampling from the density $p'(u)$. This equation suggests the following variance reduction estimation scheme which is called *importance sampling* (IS) (see [64] and references within): Draw N samples $u_1, \ldots, u_N$ using the density p'. Then, by equation (3.5), an unbiased estimate of P is given by

$$\hat{P}_N = \frac{1}{N}\sum_{n=1}^{N} I_A(u_n)L(u_n) \tag{3.6}$$

i.e., P can be estimated by simulating a random variable with a different density and then unbiasing the output $I_A(u_n)$ by multiplying with the likelihood ratio. We call $p'(u)$ the *transformed density*. Since any density satisfying the absolute continuity condition can be used as the transformed density, the question arising is which is the *optimal* transformed density, i.e., which is the density that minimizes the variance of $\hat{P}$. The optimal density thus defined is not practically feasible because it implies knowledge of P [65].

Typically, the search for $p'(u)$ focuses on constrained or parametric solutions. A general rule for choosing a favorable transformed density is to make the likelihood ratio small on the set A. When A is a rare event under density $p(u)$, by appropriately choosing a density $p'(u)$ we can make the event A more likely to occur and at the same time achieve a reduction of the variance of the estimate $\hat{P}$. For more about the IS technique, the interested reader should consult [64] and [65]. Importance sampling has been successfully applied to the simulation of various SRD processes. A variety of approaches, namely analytical, large deviation-based, and statistical have been proposed for the selection of $p'(u)$ ([64, 65, 66, 67] and references within).

3.3 Transformed Density and Likelihood Ratio

In order to apply the theory of importance sampling to efficiently simulate rare buffer overflow in an ATM multiplexer under VBR video traffic, we need to construct an appropriate "transformed" arrival traffic stream, calculate the corresponding likelihood ratio, and choose optimal (or simply favorable) transforming parameter values. In [51] the transformed process and likelihood ratio were described for simulating FGN processes. Here, we extend those results for the case of a self-similar Gaussian process that serves as the *background* process for the generation of realistic VBR video traffic.

Let $\mathbf{X}$ be the background self-similar Gaussian process as defined in Section 3.2, with mean value $m = 0$. Define a new process $\mathbf{X}' = \{X'(k) : X'(k) = X(k) + m^*, k = 1, \ldots\}$. It is easy to see that the process $\mathbf{X}'$, which we call the *transformed background process*, is a Gaussian process with mean m^*, and that its variance and correlation function are the same as for $\mathbf{X}$. Given a realization $(x'_1, \ldots, x'_{k-1})$ of

process $\mathbf{X}'$, the corresponding realization of process $\mathbf{X}$ satisfies $x_j = x'_j - m^*$, for $j = 1, 2, \ldots, k-1$. From equations (2.17)–(2.18),

$$\begin{aligned} \mathrm{E}_{X'}(X'_k | x'_{k-1}, \ldots, x'_1) &= m^* \\ &+ \mathrm{E}_X(X_k | x'_{k-1} - m^*, \ldots, x'_1 - m^*) \\ &= m^* + \mathrm{E}_X(X_k | x_{k-1}, \ldots, x_1) \\ &= m^* + \sum_{j=2}^{k} \phi_{kj}(x_{k-j}) \\ &= m^* + \sum_{j=2}^{k} \phi_{kj}(x'_{k-j} - m^*) \\ &= m^* + m_{k,X'} \text{ for } k = 2, 3, \ldots \end{aligned} \tag{3.7}$$

where

$$m_{k,X'} \triangleq \sum_{j=2}^{k} \phi_{kj}(x'_{k-j} - m^*) \tag{3.8}$$

Also from equations (2.17)–(2.18)

$$\mathrm{var}_{X'}(X'_k | x'_{k-1}, \ldots, x'_1) = \mathrm{var}_X(X_k | x_{k-1}, \ldots, x_1) \tag{3.9}$$

In IS simulation, we simulate a transformed *foreground* arrival process $\mathbf{Y}'$ instead of the arrival process $\mathbf{Y}$, where $\mathbf{Y}$ is defined in equation (2.23) and $Y'_k = h(X'_k) = F_Y^{-1}(F_X(X'_k))$. It is straightforward to observe that, during the simulation we need only calculate the likelihood ratio of the *background* processes, $\mathbf{X}$ and $\mathbf{X}'$.

The likelihood ratio of the corresponding background processes, $\mathbf{X}$ and $\mathbf{X}'$ respectively, is calculated as follows: Let $(x'_1, \ldots, x'_{k-1})$ be also taken as a realization of the netput process $\mathbf{X}$. Then,

$$\begin{aligned} \mathrm{E}_X(X_k | x'_{k-1}, \ldots, x'_1) &= \sum_{j=2}^{k} \phi_{kj}(x'_{k-j}) \\ &= m_{k,X} \text{ for } k = 2, 3, \ldots \end{aligned} \tag{3.10}$$

$$\tag{3.11}$$

where

$$m_{k,X} \triangleq \sum_{j=2}^{k} \phi_{kj}(x'_{k-j}) \tag{3.12}$$

We also have

$$\mathrm{var}_X(X_k|x'_{k-1},\ldots,x'_1) = \mathrm{var}_{X'}(X'_k|x'_{k-1},\ldots,x'_1) \tag{3.13}$$

The likelihood ratio of the background processes up to time k is

$$\begin{aligned} L(k) &= \frac{f_X(x'_1,\ldots,x'_k)}{f_{X'}(x'_1,\ldots,x'_k)} \\ &= \frac{f_X(x'_1)f_X(x'_2|x'_1)\cdots f_X(x'_k|x'_{k-1},\ldots,x'_1)}{f_{X'}(x'_1)f_{X'}(x'_2|x'_1)\cdots f_{X'}(x'_k|x'_{k-1},\ldots,x'_1)} \\ &= \prod_{i=1}^{k} L_i \end{aligned} \tag{3.14}$$

where

$$L_1 = \frac{f_X(x'_1)}{f_{X'}(x'_1)} \tag{3.15}$$

$$L_i = \frac{f_X(x'_i|x'_{i-1},\ldots,x'_1)}{f_{X'}(x'_i|x'_{i-1},\ldots,x'_1)} \quad \text{for} \quad i = 2,3,\ldots,k \tag{3.16}$$

Then, from equations (3.7) to (3.11), we have

$$L_i = \frac{e^{\theta_i x'_i}}{M_i} \quad for\ i = 2,3,\ldots \tag{3.17}$$

where

$$\theta_i = -\frac{-m_{i,X} + m^* + m_{i,X'}}{\sigma^2 \prod_{j=2}^{i}(1-\phi_{jj}^2)} \tag{3.18}$$

$$M_i = e^{-\theta_i(-m_{i,X}-m^*-m_{i,X'})/2} \tag{3.19}$$

and

$$L_1 = e^{-\frac{2m^* x_1' - m^{*2}}{2\sigma^2}} \tag{3.20}$$

The probability $\Pr(Q_k > b)$ can be estimated by observing N iid replications of the realization $w_1^{(n)}, \ldots, w_k^{(n)}$ of $\mathbf{W}$, for $n = 1, \ldots, N$. Let $L^{(n)}$, $n = 1, \ldots, N$, denote the corresponding likelihood ratio for each replication. Then, we propose the following simulation procedure for estimating $\Pr(Q_k > b)$:

1. Initialize $i = 1, n = 1$;

2. Generate a sample point x_i by Hosking's method described in Section 3;

3. Generate a sample point y_i' by the equation $y_i' = x_i + m^*$;

4. Generate a sample point w_i by replacing the process $\mathbf{Y}$ with the process $\mathbf{Y}'$ in the definition of total netput process;

5. If $w_i \leq b$ and $i < k$, then repeat from step 2 with $i = i + 1$; otherwise continue with step 6;

6. If $w_i \leq b$ and $i = k$, set $I_n = 0$ and go to step 8; otherwise continue with step 7;

7. Set $I_n = 1$ and calculate $L^{(n)} = L(i)$ via equations (3.14) to (3.20);

8. If $n = N$ evaluate the estimate using $\hat{P} = \frac{1}{N} \sum_{n=1}^{N} I_n L^{(n)}$; otherwise set $n = n + 1$, $i = 1$ and goto step 2.

3.4 Optimal Transformed Mean Value

Based on the above description, we can apply IS by suitably modifying (transforming) the mean of the arrival process. However, an efficient method to obtain a favorable (or near-optimal) transformed mean remains to be devised. Here, we describe two such

methods, namely a heuristic search and an approximate analytical approach. While the analytical approach can be only applied to the Gaussian process, the heuristic search approach can be applied to any processes. It has been successfully applied to traditional (SRD) models (see [66] and references within), and will be briefly explained in Section 3.6.

We now focus our attention on the approximate analytical approach.If $\mathbf{X}$ is a FGN process with mean m, variance σ^2 and Hurst parameter H, $\mathbf{W}$ will be an stationary increment Gaussian process with mean $(m-\mu)k$ and variance $\sigma^2 k^{2H}$. Because stationary Gaussian processes are reversible, for $Q_0 = 0$,we have

$$\Pr(Q_k > b) = \Pr(\max_{0\le i\le k}(W_k - W_i) > b) = \Pr(\max_{0\le i\le k} W_i > b),\ for\ k = 0, 1, 2, \ldots \tag{3.21}$$

From equation (3.21), we have [9, 8]

$$\Pr(Q_k > b) > \max_{0\le i\le k} \Pr(W_i > b) \triangleq \mathrm{P}_{W,k} \tag{3.22}$$

This approximation, which is an optimistic bound for $\Pr(Q_k > b)$, can be quite accurate for any time k, when b is large. The heuristic behind this assumption is that rare events occur in the most likely way [68, 8]. Therefore we have

$$\Pr(Q_\infty > b) \simeq \sup_{i\ge 0} \Pr(W_i > b) \triangleq \mathrm{P}_{W,\infty} \ \text{ for large } b \tag{3.23}$$

By the assumptions in section 3.1 and 3.3, W_i has the distribution $N(-\mu i, \sigma^2 i^{2H})$. Therefore we have

$$\sup_{i\ge 0} \Pr(W_i > b) = 1 - \inf_{i\ge 0} \Phi(\frac{b+\mu i}{\sigma^2 i^{2H}}) \tag{3.24}$$

where $\Phi(.)$ is the cdf of a standard Gaussian distribution. And then we have

$$\arg\inf_{i\ge 0} \Phi(\frac{b+\mu i}{\sigma^2 i^{2H}}) = \arg\inf_{i\ge 0}(\frac{b+\mu i}{\sigma^2 i^{2H}}) = \frac{b}{c} \ \triangleq\ k_s \tag{3.25}$$

where $\arg\inf_{i\geq 0}$ denotes the value of the argument i for which the minimum is achieved and c is defined in equation (3.3)[8, 9]. Now by definition, we have

$$\mathrm{P}_{W,\infty} = \Pr(W_{k_s} > b) \tag{3.26}$$

Similar procedure can show that, for $k > k_s$, $\mathrm{P}_{W,k} = \Pr(W_{k_s} > b)$. Thus, loosely speaking, k_s is the time when the buffer overflow probability for the specific buffer size approximately equals to the buffer overflow probability of steady-state,i.e. $\Pr(Q_\infty > b) \simeq \Pr(W_{k_s} > b)$. A very accurate approximate formula for calculating $\Pr(W_{k_s} > b)$ (i.e., the tail of a Gaussian distribution) was recommended in [69]. The above approximation procedures lead to quite accurate results, as our results in Section 3.5 indicate.

Since $\Pr(Q_\infty > b) \simeq \Pr(W_{k_s} > b)$, our approximate analytical approach consists of finding a near-optimal transformed mean value for $\Pr(W_{k_s} > b)$ and then applying that same transformed value to the simulation of $\Pr(Q_\infty > b)$. Since W_{k_s} is normally distributed with mean $-\mu k_s$ and variance $\sigma^2 k_s^{2H}$, the likelihood ratio corresponding to experiments with buffer size b will be

$$L(k_s, b) = \frac{e^{-\frac{(b+\mu k_s)^2}{2\sigma^2 k_s^{2H}}}}{e^{-\frac{(b-m^*_W k_s)^2}{2\sigma^2 k_s^{2H}}}} \tag{3.27}$$

where m^*_W is the transformed mean value. By minimizing the above likelihood ratio as suggested in [67, 70], we can find a near-optimal transformed mean $m^*_{W,opt} \simeq b = ck_s$. Hence, a near-optimal transformed mean value for process $\mathbf{Y}$ can be found as follows

$$m^*_{opt} \simeq m^*_{W,opt}/k_s \simeq c = \mu/H - \mu \tag{3.28}$$

Furthermore, it is reasonable to assume and we will verify through simulations later that m^*_{opt} is also near-optimal for the estimation of the (transient) probability $\Pr(Q_k > b)$ when $k > k_s$.

In the next section we will show using numerical examples that the heuristic result and the approximate result described in the last paragraph are in very close agreement. Therefore, the above approximate value for m^*_{opt} can be used directly or provide a good initial estimate for the search of a near-optimal transformed mean value.

3.5 Numerical Results for FGN

For IS simulation, the estimator $\hat{P}$ of the unknown probability $\Pr(Q_k > b)$ is a function of $(m, m^*, \mu, H, k, b, N, \sigma^2)$. Since our set-up is translation-invariant with respect to m, we assume $m = 0$ without loss of generality. We let σ be fixed at $\sigma = 1$, since as shown in the Appendix, by changing the number of multiplexed homogeneous sources L, we can observe the same effect as if scaling σ.

We divide our simulation experiments into two cases, one with $H = 0.7$, which represents less bursty traffic, and one with $H = 0.9$ representing more bursty traffic. In each case, we consequently discuss dependence on the transformed mean value m^*, on service rate μ, on stopping time k, on the buffer size b, and on the number L of multiplexed homogeneous sources. By homogeneous sources we mean sources which have the same Hurst parameter. In the final part, we simulate multiplexing two heterogeneous sources, one with $H = 0.7$ and one with $H = 0.9$. We also provide example values of the improvement factor of our IS technique over conventional MC simulation.

3.5.1 Case I: $H = 0.7$

1. The dependence on m^*:

It is important to point out that the IS estimator of $\Pr(Q_k > b)$ is always *unbiased*, regardless of the value of m^*. However, the sample path properties as well as the variance of the IS estimator are dramatically affected by the choice of m^*. This is the basis for the heuristic search procedure for the optimal transformed mean value, described in [66]. Fig. 3.1 is an example of plotting the estimated $\Pr(Q_k > b)$, while Fig. 3.2 plots the normalized variance $\sigma^2_{\hat{P}}/\hat{P}^2$ of $\hat{P}$, both versus the transformed mean value m^*. The value corresponding to $m^* = -0.5$ is in fact the result of direct (conventional) Monte Carlo (MC) simulation. We can see that, as m^* increases, the normalized variance exhibits a clear "valley" around the most favorable values of m^*. This behavior, as well as the behavior of the estimated $\Pr(Q_k > b)$ versus m^*, is discussed in detail in [66] and the references therein. The minimum normalized variance appears around $m^* = 0.2$ which coincides with the approximate value m^*_{opt} from equation (3.28).

2. The dependence on μ:

Fig. 3.3 shows the estimated $\log \Pr(Q_\infty > b)$ versus the service rate μ. Each simulation is based on 1000 iid replications. In all simulations, we apply the IS technique using the near-optimal transformed mean value of equation (3.28). Our simulation result is compared with the optimistic bound obtained from equation (3.22).

3. The dependence on k:

Fig. 3.4 depicts the estimated $\log \Pr(Q_k > b)$ versus the stopping time k. Each simulation is based on 1000 iid replications. The dependence of $\log \Pr(Q_k > b)$ on k

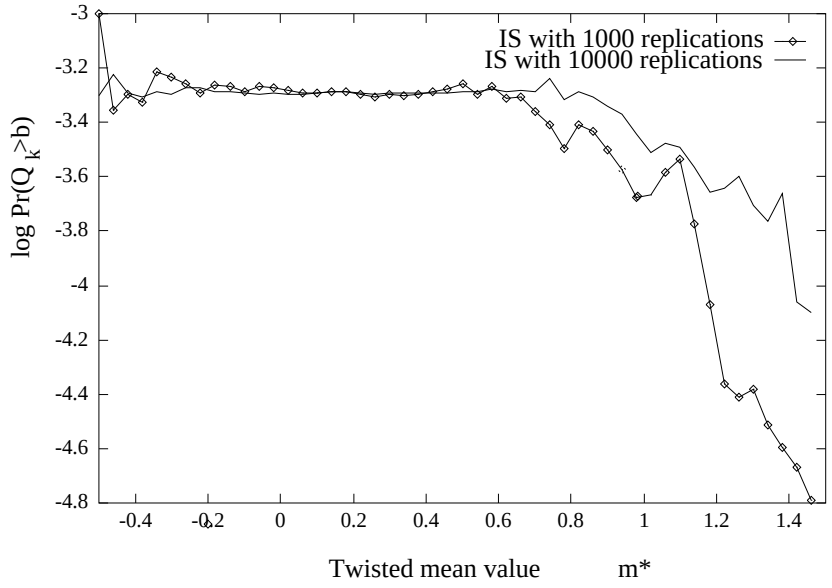

Figure 3.1: Estimated $\log \Pr(Q_\infty > b)$ versus the transformed mean value m^*. The Hurst parameter is $H = 0.7.b = 50$, $\mu = 0.5$.

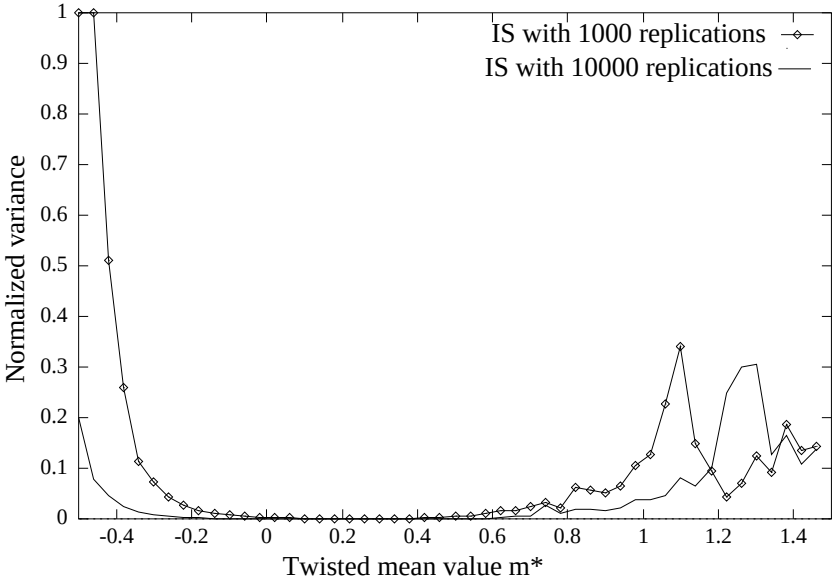

Figure 3.2: Normalized variance $\sigma^2_{\hat{P}}/\hat{P}^2$ of estimated $\log \Pr(Q_\infty > b)$ versus the transformed mean value m^*. The Hurst parameter is $H = 0.7.b = 50$, $\mu = 0.5$.

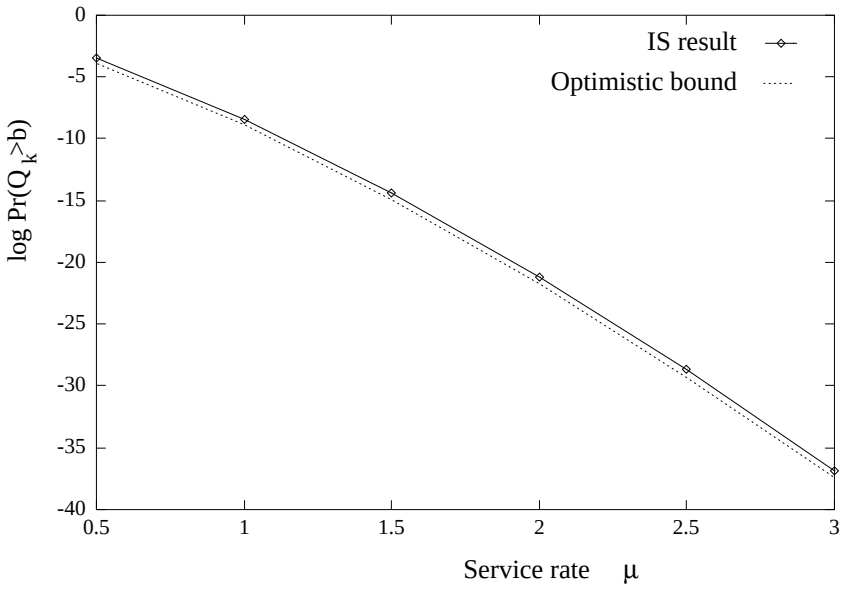

Figure 3.3: Estimated $\log \Pr(Q_\infty > b)$ versus the service rate μ. Each simulation is based on 1000 iid replications. The Hurst parameter is $H = 0.7.b = 50$.

reflects the transient nature of our experiments. The curves show how the queueing state approaches asymptotically the steady-state as k increases. In order to see how the time of entering steady-state depends on the buffer size b, in Fig. 3.4 we show results with different buffer sizes. For $b = 20$, we also show the direct MC simulation result in order to illustrate that the IS approach is in agreement with direct simulation. When b becomes larger, a direct simulation would have become impractically long, while IS simulation gives very good result even with only 1000 replications. Notice that the empirically observed times of entering steady-state are very close to the k_s predicted by equation (3.25), with $c = \mu/H - \mu$.

4. The dependence on b:

We simulate the dependence of $\log \Pr(Q_k > b)$ on b for two stopping times k: one is time k_s predicted by equation (3.25), and the other is $2 \times k_s$. We compare our simulation results with the large deviation result of equation (3.3) and the optimistic

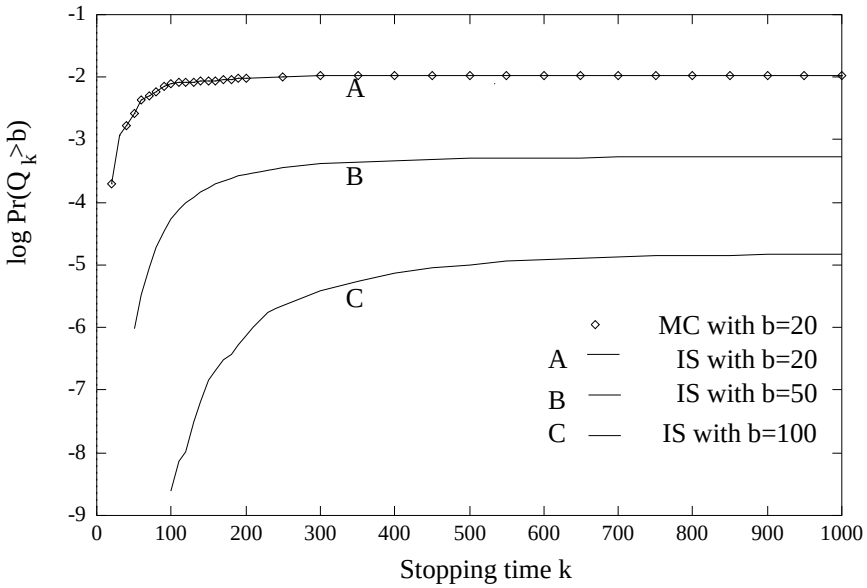

Figure 3.4: Estimated $\log \Pr(Q_k > b)$ versus stopping time k. Each simulation is based on 1000 iid replications. The Hurst parameter is $H = 0.7.\mu = 0.5$.

bound of equation (3.22) in Fig. 3.5. Each simulation is based on 1000 iid replications. It can be seen that, with increasing stopping time, the results are approaching the large deviation result. This was to be expected since the large deviation result is based on a steady-state regime while our simulation captures the transient behavior.

5. The dependence on L:

Fig. 3.6 shows the estimated $\log \Pr(Q_k > b)$ versus the number of homogeneous multiplexed sources L, for $H = 0.7$. Each simulation is based on 1000 iid replications. Fig. 3.6 also depicts the optimistic bound of equation (3.22). With changing L, both the buffer size b and the utilization are kept constant(i.e. The service rate is in fact $L\mu$ in order to maintain the same load on the queue). The multiplexing gain (i.e., reduction in $\Pr(Q_k > b)$ with increasing L) is evident.

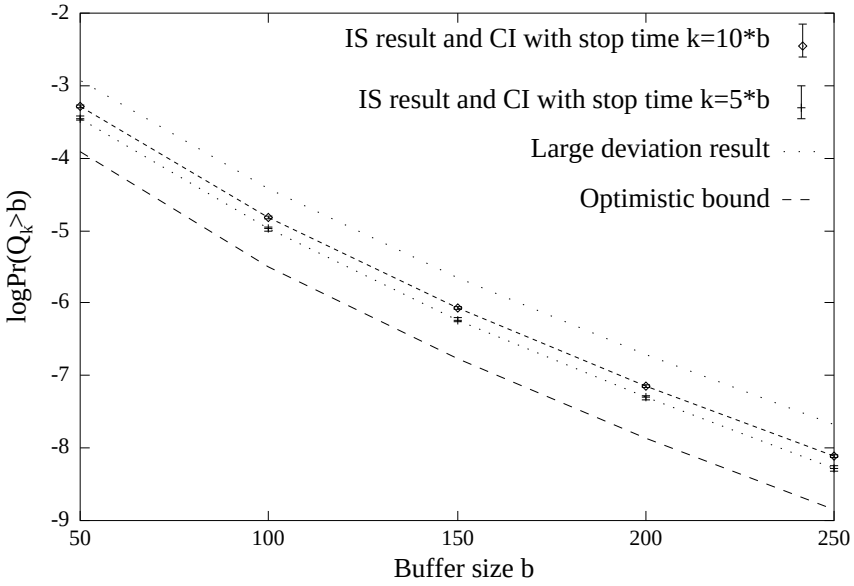

Figure 3.5: Estimated $\log \Pr(Q_k > b)$ versus the buffer size b with their corresponding 95% confidence intervals(CI). Each simulation is based on 1000 iid replications. The Hurst parameter is $H = 0.7.\mu = 0.5$.

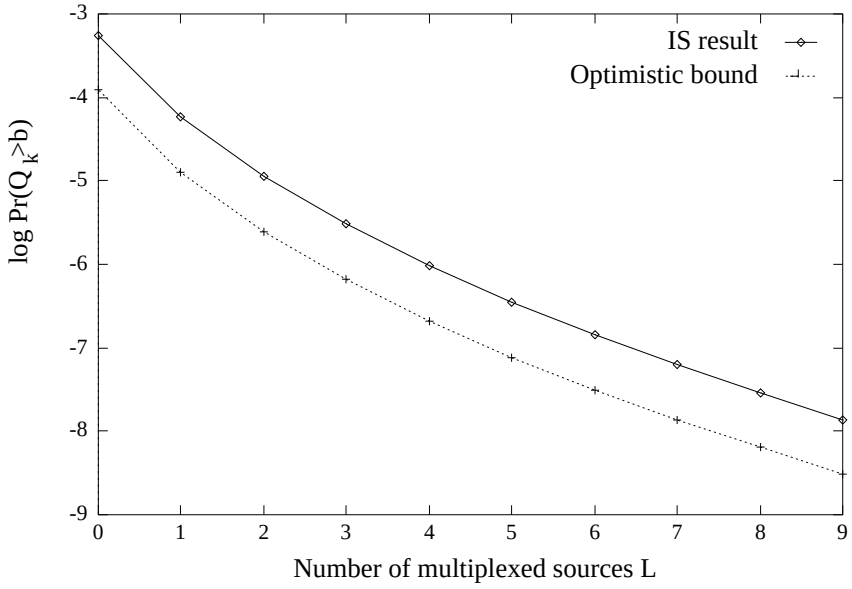

Figure 3.6: Estimated $\log \Pr(Q_k > b)$ versus the number of multiplexed sources L. Each simulation is based on 1000 iid replications. The Hurst parameter is $H = 0.7.b = 50$.

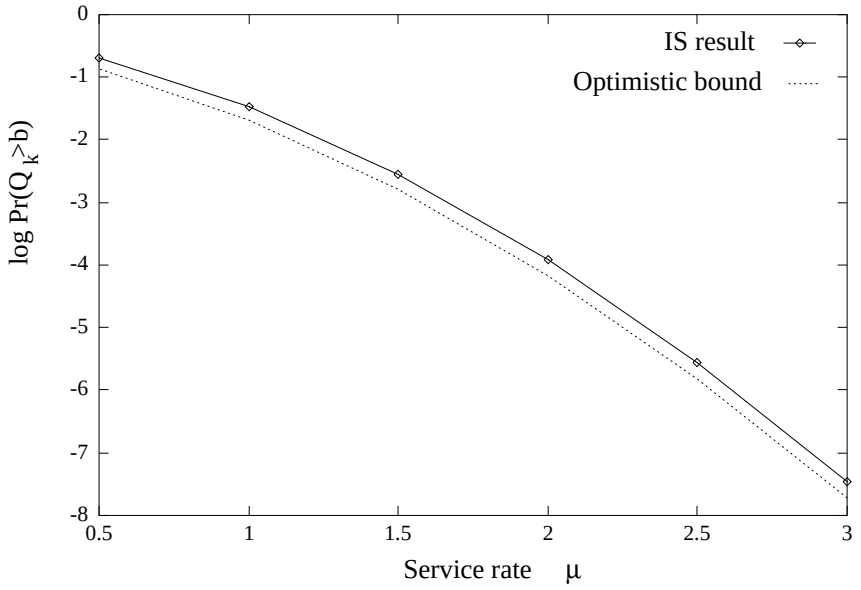

Figure 3.7: Estimated $\log \Pr(Q_k > b)$ versus the service rate μ. Each simulation is based on 1000 iid replications. The Hurst parameter is $H = 0.9.b = 50$.

3.5.2 Case II: $H = 0.9$

The simulation procedures are basically the same as for $H = 0.7$. Therefore, we only comment on those features which are different from previous experiments.

1. The dependence on μ:

Fig. 3.7 shows the estimated $\log \Pr(Q_k > b)$ versus the service rate μ, for $H = 0.9$. Each simulation is based on 1000 iid replications. Comparing this result with Fig. 3.3, we see that increasing μ is more efficient for burstier sources. Based both on the optimistic bound (3.22) and the large deviation result (3.3), it is easy to obtain $\Pr(Q_\infty > b) \approx de^{-a\mu^{2H}}$, where a, d are positive, slowly changing functions of μ. This result is in close agreement with our simulation results.

2. The dependence on b:

Fig. 3.8 depicts the dependence of the estimated $\log \Pr(Q_k > b)$ on b, for $H =$

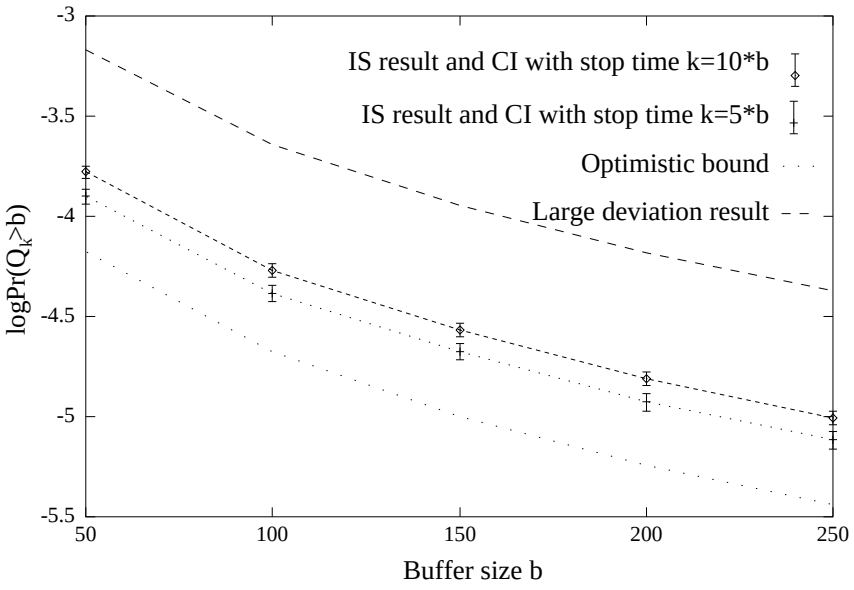

Figure 3.8: Estimated $\log \Pr(Q_k > b)$ versus the buffer size b with their corresponding 95% confidence intervals(CI). Each simulation is based on 1000 iid replications. The Hurst parameter is $H = 0.9.\mu = 2$

0.9. Each simulation is based on 1000 iid replications. Comparing this result with Fig. 3.5, we find that increasing the buffer size is less efficient in reducing the overflow probability than for less bursty sources ($H = 0.7$), while always less efficient than for SRD models (estimated $\log \Pr(Q_k > b)$ decays less than exponentially fast). This agrees with large deviation theory that predicts $\Pr(Q_\infty > b) \approx de^{-ab^{2(1-H)}}$, where a, d are positive, slowly changing functions of b.

3.The dependence on L:

Fig. 3.9 shows the estimated $\log \Pr(Q_k > b)$ versus the number of multiplexed sources L, for $H = 0.9$. Each simulation is based on 1000 iid replications. Also similar to Fig. 3.6, with changing L, both the buffer size b and the utilization are kept constant. Comparing Fig. 3.9 with Fig. 3.6, we see that increasing the number of multiplexed sources leads to higher gains (larger reductions in overflow probability)

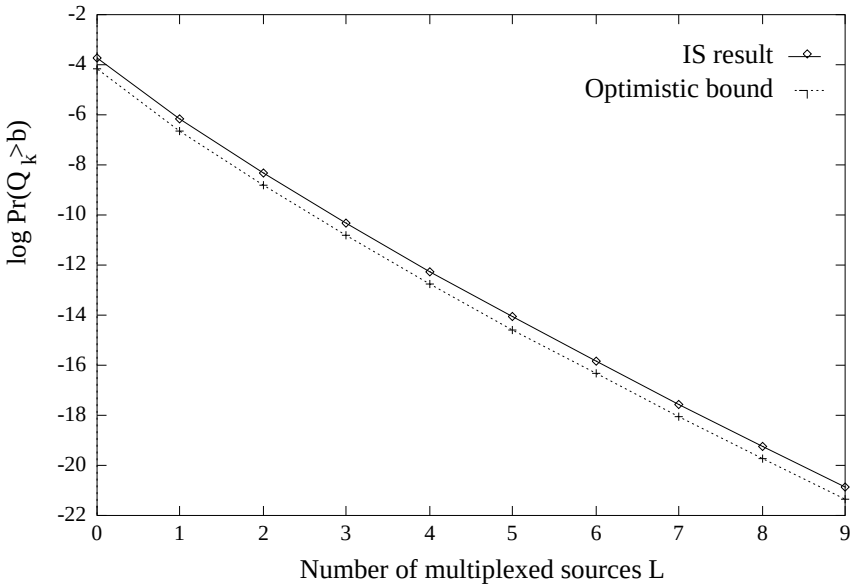

Figure 3.9: Estimated $\log \Pr(Q_k > b)$ versus the number of multiplexed sources L. Each simulation is based on 1000 iid replications. The Hurst parameter is $H = 0.9.b = 50$.

for burstier sources (higher values of H). Using large deviation theory (but also from the optimistic bound of Section 4) we obtain that the dependence with respect to L of $\Pr(Q_\infty > b) \approx de^{-aL^{2H-1}}$, where a, d are positive, slowly changing functions of L. This result is, again, in close agreement with our simulation results.

3.5.3 Case III: Multiplexing Heterogeneous Sources

We now consider the aggregation of two independent FGN processes $\mathbf{X}_1$ and $\mathbf{X}_2$. We assume that $\mathbf{X}_1$ and $\mathbf{X}_2$ have zero mean and unit variance. Their corresponding correlation functions are defined as in (2.9) with $H = H_1$ for $\mathbf{X}_1$ and $H = H_2$ for $\mathbf{X}_2$. We assume $H_1 > H_2$ and the service rate to be μ. Then the mean of total work load process $\mathbf{W}$ is $-\mu\ k$, $k = 1, 2, \ldots$, and the variance is $k^{2H_1} + k^{2H_2}$. We can show the following lemma:

Lemma 1. Let $\mathbf{X}_i$, $i = 1, 2$, be two FGN traffic processes with zero mean, variances σ_i^2, and Hurst parameters H_i, $i = 1, 2$, respectively. Let $H_1 > H_2$ and $1/2 < H_i < 1$, $i = 1, 2$. Then the queue length process resulting from the aggregate FGN traffic satisfies:

$$\lim_{b\to\infty} \sigma_1^2 b^{-2(1-H_1)} \log \Pr(Q_\infty > b) = -c^{-2(1-H_1)}(c+\mu)^2/2 \tag{3.29}$$

The proof is given in the Appendix.

Clearly, we have the same result as in equation (3.3) with $H = H_1$. This indicates that the steady-state tail distribution is dominated by the arrival process with the larger Hurst parameter.

We will find that the simulation procedures for multiplexing heterogeneous sources are similar to the steps for single source if we note that the aggregate process is still a Gaussian process and its mean, variance and autocorrelation function can be calculated from the corresponding values of individual sources.Fig. 3.10 shows the result of multiplexing two self-similar sources, one with $H = 0.7$ and another with $H = 0.9$. As we aggregate the two arrival sources, we also increase accordingly the total service rate in order to maintain constant load while buffer size is kept constant, and observe the gain from increased buffer capacity. As shown in Fig. 3.10, the burstier source ($H = 0.9$) will dominate the queueing tail distribution, which agrees with the large deviation result in the Appendix.

3.5.4 IS Improvement Factor

The speed-up or improvement factor of IS over conventional MC simulation denotes the relative decrease in the required number of replications in order to achieve the

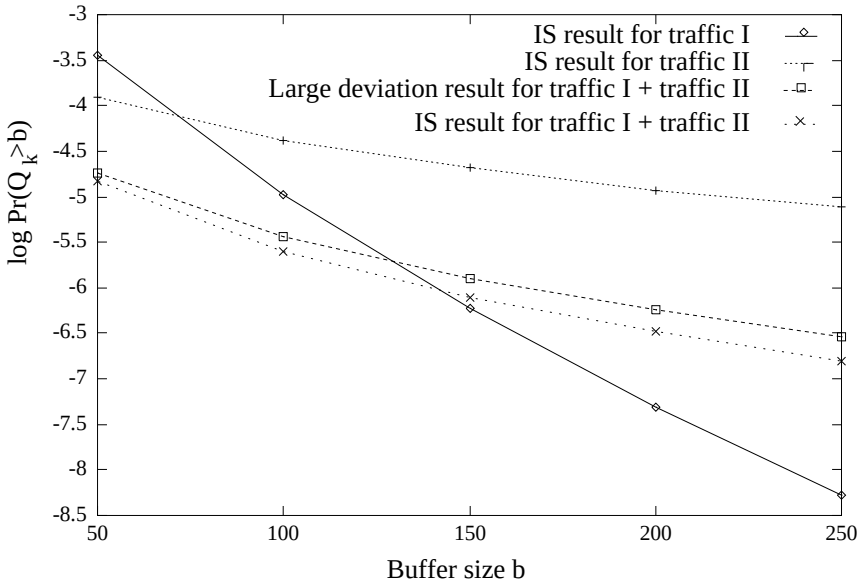

Figure 3.10: Estimated $\log \Pr(Q_k > b)$ versus the buffer size b (heterogeneous sources, traffic I with $H = 0.7$, traffic II with $H = 0.9$). Each simulation is based on 1000 iid replications.

same statistical accuracy. Let $\sigma^2_{MC}(N)$ denote the estimator variance after N replications using conventional MC simulation. Furthermore, let $\sigma^2_{IS}(N)$ denote the estimator variance after N replications using IS simulation. Then the improvement factor is defined as $\sigma^2_{MC}(N)/\sigma^2_{IS}(N)$.

Denote with P the probability to be estimated using N replications. Then, $\sigma^2_{MC}(N) = \mathrm{P}(1-\mathrm{P})/N$. Since only an estimate $\hat{\mathrm{P}}$ of P is known, we use the approximation $\sigma^2_{MC}(N) \simeq \hat{\mathrm{P}}(1-\hat{\mathrm{P}})/N$. We also approximate the true $\sigma^2_{IS}(N)$ with a sample variance estimate. Fig. 3.11 shows the estimated improvement factor versus buffer size, b, for Case I ($H = 0.7$), and Case II ($H = 0.9$), respectively.

We observe significant improvement factors for both cases. The improvement factor increases dramatically as the buffer size increases (i.e., as the overflow probability decreases), as is desirable.

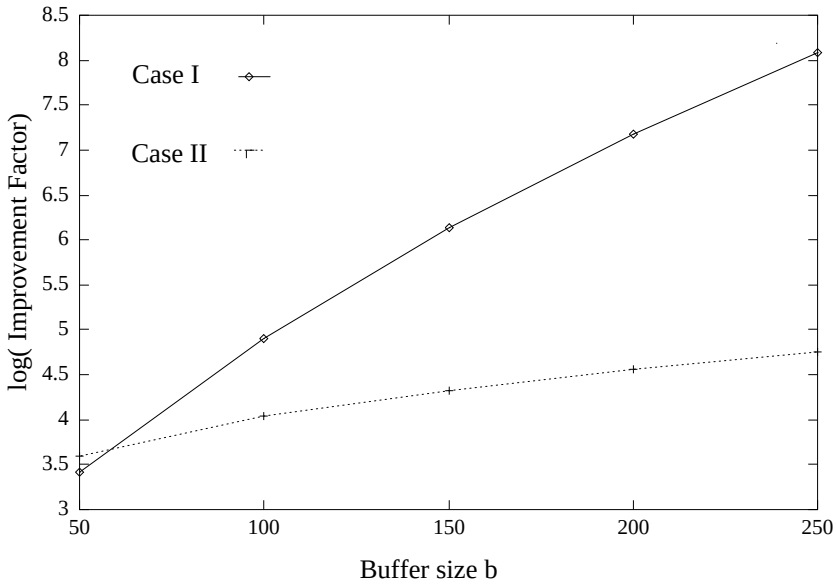

Figure 3.11: Estimated IS improvement factors over conventional MC simulation. Improvement factors denote the ratio of required number of replications for the same statistical accuracy, and are plotted here versus buffer size, b, for Case I ($H = 0.7$), and Case II ($H = 0.9$), respectively.

3.6 Buffer Overflow Studies in an ATM Environment

In this section, we use the video model we built in Section 2.3.3 to conduct the study of buffer overflow in an ATM environment. For the convenience of plotting, we normalized all buffer sizes with the mean source arrival rate(i.e. 6005.2 bytes/frame for "Last Action Hero") througout this section. Analytical approaches to optimizing the form and amount of transforming for SRD models have been investigated in [64, 65, 67]. For the case of FGN processes, analytical arguments for optimizing the transforming process were given in last section. However, after the transformation in (2.23), a closed-form optimization becomes intractable, therefore we resort here to the heuristic search approach, which is based on the fact that the IS estimator

of $\Pr(Q_k > b)$ is always *unbiased*, while the sample path properties as well as the variance of the IS estimator are dramatically affected by the choice of transforming parameter values. Typically, if we observe estimates of $\Pr(Q_k > b)$ and its normalized variance as the transforming parameters change, the normalized variance exhibits a clear "valley" around the most favorable parameter values, which can be thus, approximately identified. This approach has been successfully applied to traditional (SRD) models (see [66] and references within) and to FGN processes in [51].

An optimal selection of the (transformed) mean will result in a greatly reduced variance of the estimator for $P(Q_k > b)$. A favorable (near-optimal) background (transformed) mean value can be found from plots such as the one shown in Fig. 3.12. For our experiments, we found the value 3.2 to be a near-optimal transformed mean value for our simulation scenario shown later, resulting in a variance reduction of approximately 1000 (conversely, the required number of replications for the same accuracy is reduced by a factor of 1000). In the figures that follow, when we refer to buffer size we will essentially mean the normalized buffer size, i.e., the ratio of true buffer size to mean arrival rate.

All the simulations that we have described thus far have a transient nature in the sense that they provide an estimate of the probability of buffer overflow at a given time slot k with initial zero buffer occupation. It is of particular interest to decide on a simulation run length in order to achieve steady state results, i.e. how large should k be? Fig. 3.13 shows the transient buffer overflow probability for a given buffer size b, corresponding to two initial buffer occupancy conditions, namely empty and 200. From this figure we can see that the transient time in a simulation may be reduced if the initial conditions are chosen properly. Since the generation of the

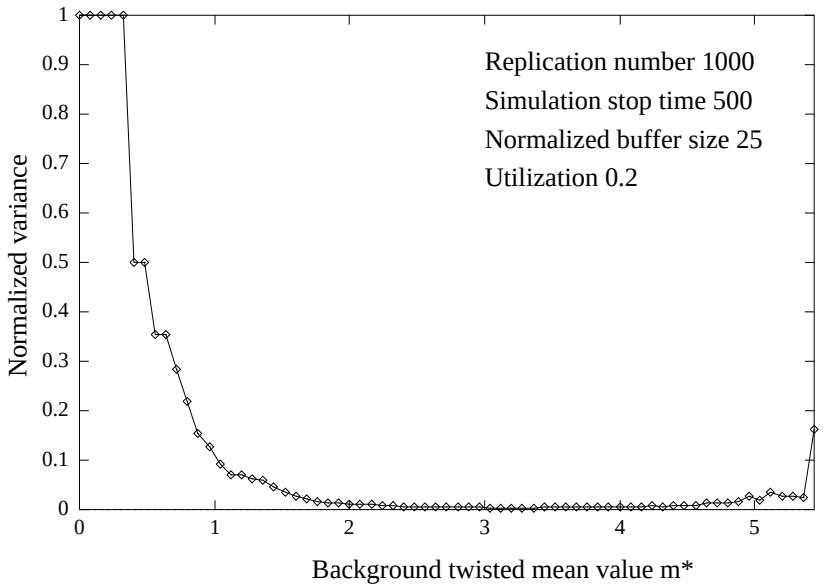

Figure 3.12: Plot of the estimated normalized variance of the estimator versus the mean value of background process transforming, m^*. Results correspond to a stopping time $k = 500$, utilization 0.2, buffer size $b = 25$, and 1000 replications.

background process $\mathbf{X}'$ may be computationally demanding, a small transient period may be highly desirable.

Fig. 3.14 shows approximately steady state results ($k = 2000$) for several service rates (corresponding to different system utilization values). Clearly enough, the decay of overflow probability is far from exponential, contrary to the prediction of traditional models. This is further illustrated in Fig. 3.15, where we compare three models. The first video model possesses only the SRD and includes only the exponentially decaying part of the autocorrelation as it was derived in Section 2.3.3. The second model is the one exhibiting both the SRD and LRD. The third model captures only the LRD structure, based on a single FGN background process (i.e., there is no short-term exponential component). It is easy to see that for small buffer size the difference in the probability of buffer overflow is not significant, but as the buffer size

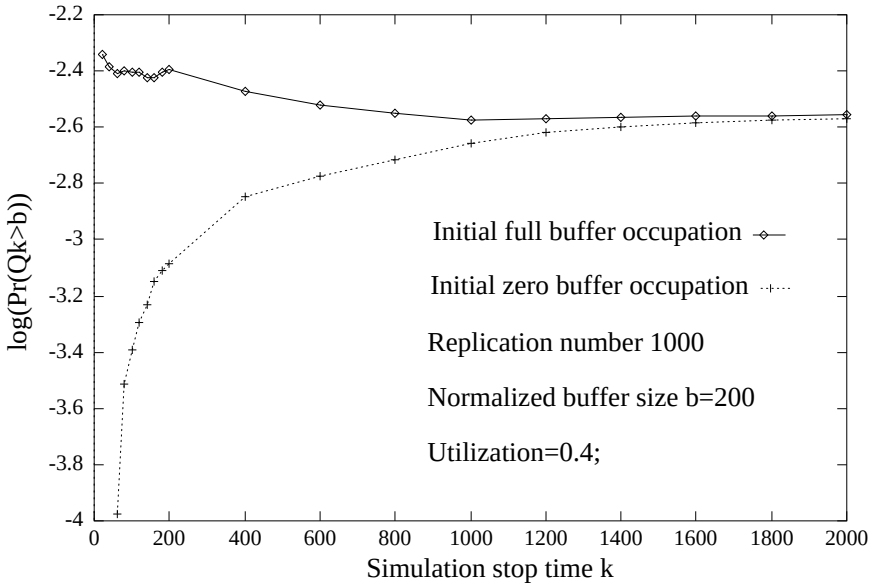

Figure 3.13: Transient buffer overflow probability, using 1000 replications, $b = 200$, and a utilization of 0.4.

increases the estimate based on the SRD model decays much faster than the one based on the model characterized by both LRD and SRD. Finally, as expected, although the third model exhibits the appropriate asymptotic behavior, the corresponding loss probability decays too fast for small buffer sizes.

In Fig. 3.16, we compare our approximate steady state results with the results using the empirical video trace. Different from the results in Fig. 3.14, the steady state results in Fig. 3.16 treat the Hurst parameter H as Gaussian random variable with parameters as measured by Frequency Domain MLE in the last chapter rather than a constant. The 95% confidence intervals for the steady state results are also displayed. From Fig. 3.16, we can see that although both results agree closely at high utilizations, they are significantly different at low utilizations. We believe that the reason behind these errors is that, for the steady state results, simulations were based on 1000 independent replications for each different utilization and buffer size.

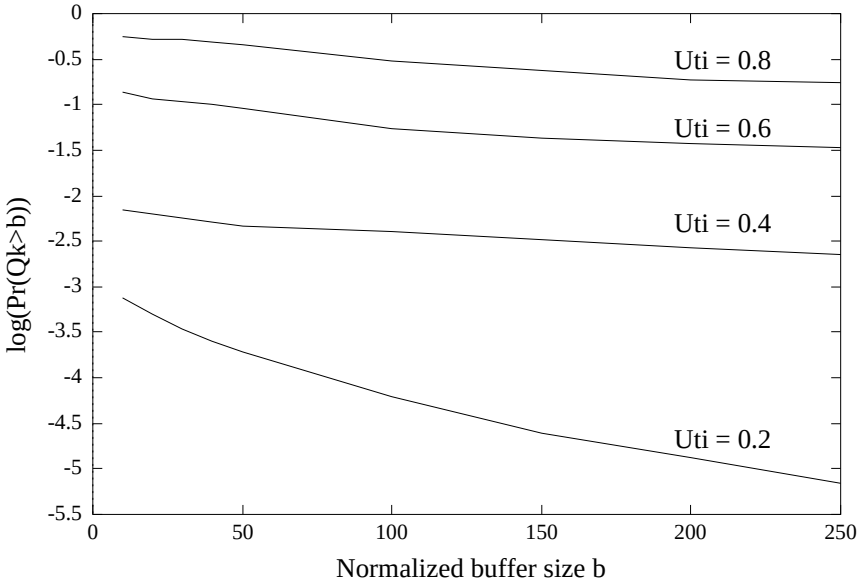

Figure 3.14: Overflow probability versus buffer size b, for different utilization values, using 1000 replications and $k = 10b$.

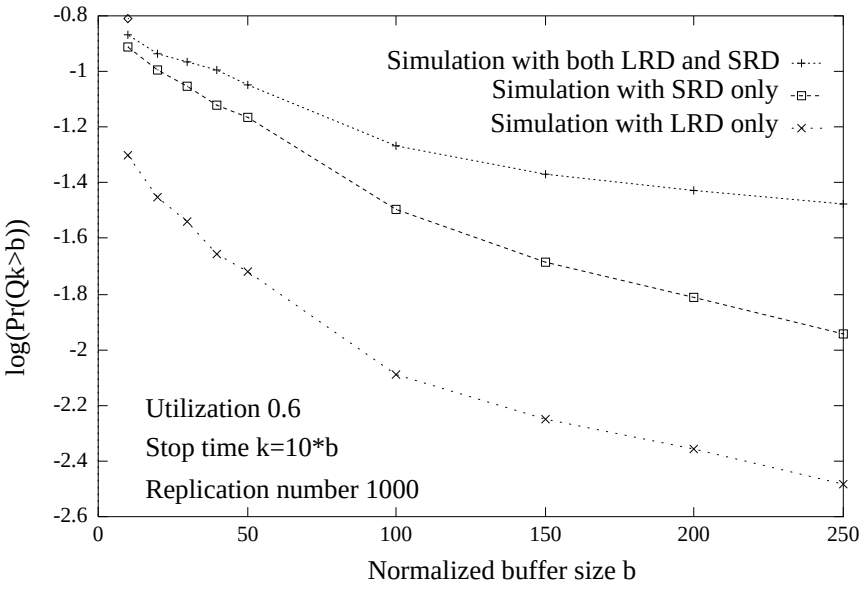

Figure 3.15: Overflow probability versus buffer size b for three cases: using the simulated model with both LRD and SRD, using a simulated model *without* LRD, and using a model with *only* LRD.

Since only one empirical trace was available, it was impossible to perform independent replications for each simulation involving real data. Even if the real data were split into batches we would expect significant correlations between batches due to the self similar nature of the traffic. Therefore, very few independent samples can be used to estimate buffer overflow rates causing significant variance. This argument is further illustrated in Fig. 3.17 where we try to compare the queueing deviation results of the model with both SRD and LRD structures and the model with SRD only. For the simulation cases with a single long synthetic trace, we deliberately choose the stop time equal to the length of the empirical video trace so that they may reflect possible deviations of real video traces. It is clearly shown in Fig. 3.17 that the model with both LRD and SRD structure results in a much larger deviation than the model with SRD only. This agrees with the conclusion in [12] where it is shown that SRD model can have a closer match than LRD if simulation is performed with one replication only and the buffer size in a queueing system is small.

For lower utilizations and larger buffer sizes however this disagreement is expected to be more profound as shown for the cases with utilizations 0.4 and 0.2 in Fig. 3.16 since in addition to the above mentioned reasons, a cell loss event now becomes a rare event making the number of meaningful independent samples even less.

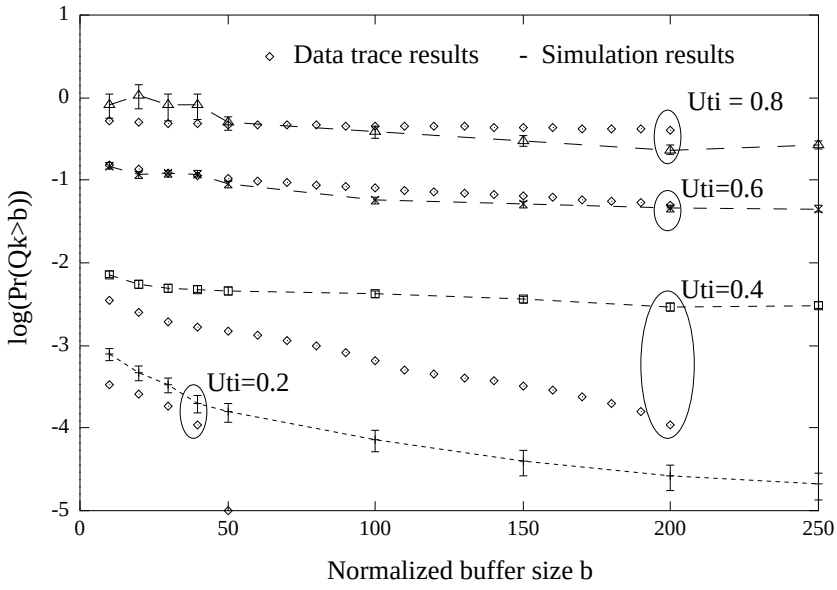

Figure 3.16: Overflow probability with its 95% CI versus buffer size b, for different utilization values, using 1000 replications ($k = 10b$) from model with both LRD and SRD, and using empirical data trace.

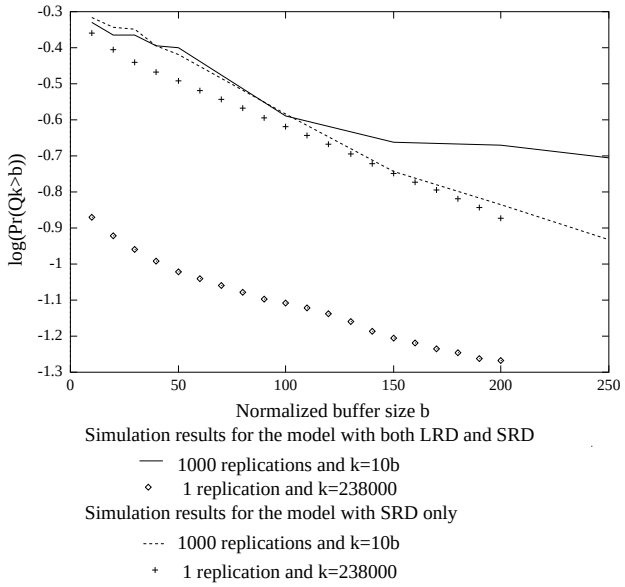

Figure 3.17: Overflow probability versus buffer size b for two models: using the simulated model with both LRD and SRD, using a simulated model *without* LRD. For each model, two cases are considered: simulation with 1000 replications (k=10b) and simulation with one replication (k=238000). For all cases, utilization value is 0.8.

Chapter 4

Network Design Issues

4.1 Implications of Self-similar Traffic

Integrated telecommunication networks carry traffic of several different classes, including real-time traffic, each with its own set of traffic characteristics and performance requirements. Two different types of solutions have been advanced to deal with this phenomenon: In *circuit-switched* networks, sufficient bandwidth is allocated to each call to handle its maximum bandwidth requirement; this guarantees that the call will receive the QoS it requires, but may be wasteful of system resources. In *packet-switched* networks, traffic from all sources is packetized, and statistical multiplexing techniques are used to combine all network traffic through a single switching fabric. This allows higher network utilization, but requires more sophisticated controls to ensure that the appropriate QoS is provided.

Different from circuit-switched networks, packet-switched networks derive their delay from several more causes. In addition to the transmission delay caused by the propagation of the packet at nearly the speed of light, there is the so called "stop-and-forward" delay at each switching point waiting for the entire packet to arrive before commencing the next stage of transmission. Also added to this fixed delay

is a variable amount of delay related to the time that each packet spends in service queues in the switches. This delay variation is what must be bounded or minimized if adequate real-time service is to be achieved. Unfortunately, this variation is highly dependent on traffic characteristics and network resources.

Earlier packet-switched networks (e.g., Internet) only provide datagram service to which the network makes no service commitments at all, except to promise not to delay or drop packets unnecessarily (this is sometimes called "best-effort" service). For this kind of network, traffic models have often been devised and selected for the analytical tractability they induce in the corresponding queueing systems.

The most striking feature of LRD traffic is that burstiness is displayed across several time scales (i.e., from milliseconds to years [2]). This burstiness can drive queueing systems into overflow state for a longer time than traditional models have predicted as shown in the last chapter(also in [8, 9]). Although significant multiplexing gain can be achieved for LRD traffic streams, the burstier stream will dominate the queueing tail distribution as it was also shown in the last chapter. This means that a traffic stream with a lower Hurst parameter may suffer the same mean delay as a traffic stream with a higher Hurst parameter. In extreme cases, a *starvation* problem may be introduced where an unfortunate call may have to wait an indefinitely long time before it receives service. This kind of problem cannot be solved by traditional priority strategies. As pointed out in [3], for self-similar traffic, high priority traffic may block low priority traffic for quite a long time, making it enter into starvation.

To solve these problems, congestion control must be implemented in such a way that the network is shared so that clients (1) receive better service than if there

were no sharing (as in a circuit switched or TDM network), and (2) are protected from the potentially negative effects of sharing [71]. The principle of *isolation* is fundamental for any real-time traffic control algorithm: The network cannot make any commitments if it cannot prevent the unexpected behavior of one source from disrupting others.

While there are numerous congestion control schemes proposed in the literature, they can be generally classified into three types [72, 71]: One kind of service commitment, which is called *guaranteed* service, depends on no other assumptions. That is, if the network hardware is functioning and the client is conforming to its traffic characterization, then the service commitment will be met. In recent years, several guaranteed service based algorithms have been developed. They are WFQ [73], Delay-EDD [16, 24], Virtual Clock [17], and PGPS [18, 19]. Similar to circuit-switched networks, these algorithms isolate each source from the others by providing it a specified share of the bandwidth under overload conditions through a proper scheduling strategy. Service is guaranteed under worst-case bounds. Other approaches such as Jitter-EDD [20], RCSP [21], Stop-and-Go [22] and Hierarchical Round Robin [23], attempt to solve the propagation of delay jitter existing in the approaches above and further decouple the interference between nodes by relaxing the work-conserving constraint.

The above guaranteed service-based approaches are very promising under traditional models because they provide a congestion-free and topology-independent solution to the high speed packet switch network. Unfortunately, they all require certain burstiness constraints [74] at the access node. A well-known such constraint is the so-called (σ, ρ) constraint which can be implemented by a leaky bucket. A leaky

bucket can be modeled by a virtual queue.

As shown in Fig. 4.1, where Q_1 is the user access queue and Q_2 is a virtual queue which we use to model the leaky bucket, if $Q_1 > 0$, we have $Q_1 = Q_2 - b$. Therefore, queueing results in [9, 6, 51] can be directly applied to estimate the loss probability at the access node regulated by a leaky bucket. The result is that, although the user can be guaranteed loss-free transport within the network using the approaches above, the user may suffer heavy losses at the access node. Without raising the bucket rate, losses may become intolerable. But increasing the bucket rate results in a loose control or no control at all over the burstiness of the user traffic. Therefore a network that uses guaranteed service must work in a low utilization region. While guaranteed service is appropriate for intolerant and rigid clients, since they need absolute assurance about the service they receive, it is not appropriate for tolerant and adaptive clients for which a small percentage of violation can be tolerated in exchange for higher network utilization.

The second congestion control approach does not provide for the worst-case scenario. Instead it guarantees a bound on the probability of lost packets based on statistical characterization of traffic [75, 76, 77, 78, 79, 80, 81, 82]. In such an approach, each flow is allotted an effective bandwidth that is larger than its average rate but less than its peak rate; network utilization is thus increased. Most of these approaches focus on finding statistical results for the steady state of the network through *a priori* characterization of flows based on a statistical model. For LRD traffic, steady state results may still play an important role in network planning or long term performance prediction. But as a tool to predict the QoS of a single user session, steady state results cannot be used as in traditional models. As our

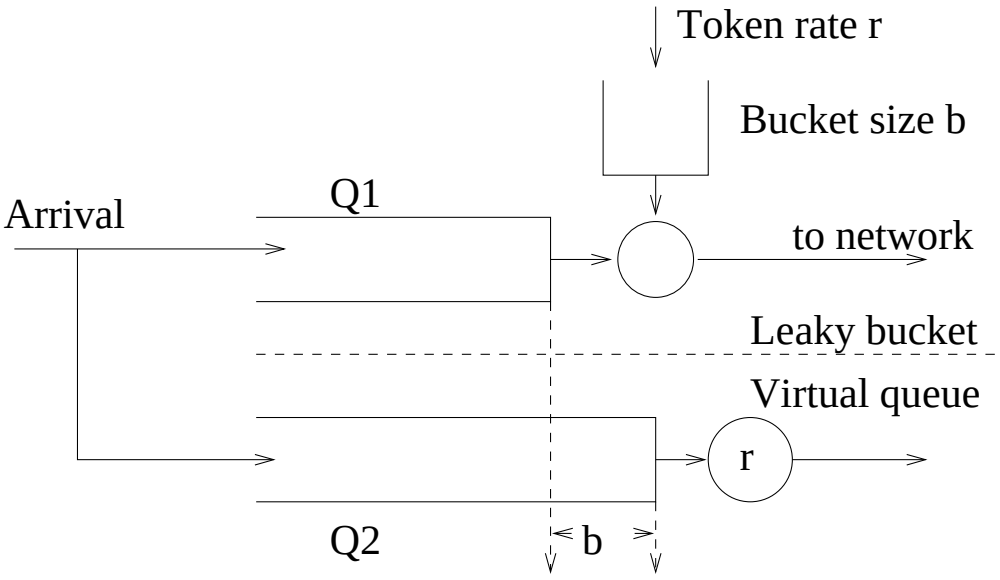

Figure 4.1: Equivalence of leaky bucket and virtual queue in terms of loss rate.

simulations indicated in the last chapter, the buffer overflow rate of a single user session may be far from the steady state results due to the inherent LRD structure. This poses significant challenges to the probabilistic service based approach [83]. In addition, analytical and simulation results in the last chapter have shown that LRD traffic shows strong subadditivity rather than linearity as effective bandwidth approach has assumed. Therefore, effective bandwidth solutions for LRD traffic can become a loose bound and result in low network utilization again.

To overcome the problems discussed above, applying a dynamic bandwidth allocation scheme seems unavoidable. *Predicted service* is therefore proposed in the literature. This approach has two components: First, the network commits to meeting service requirements under the assumption that past traffic is a guide to future behavior. This component embodies the fact that the network can take into account recent measurements of the traffic load in estimating what kind of service it can de-

liver reliably. Second, the network attempts to deliver a service that will allow an adaptive source to minimize the *post facto* delay bound and maximize throughput [71]. Several protocols to provide predicted service have been proposed (see, for example, [84, 71, 85, 86] and references therein). While these approaches differ from each other in how to react to congestion, they all require a well-behaved *traffic predictor*.

Traditional traffic predictors are based on heuristic assumptions or autoregressive (AR) filters. While heuristic assumptions typically cause over-reaction, AR filters may introduce long response times. Furthermore, an AR filter is quite difficult to implement in real-time.

In this chapter, we propose a new traffic predictor called the *Double Threshold Moving Window Detector* (DTMW). Our analytical results show that DTMW can detect and predict traffic streams with LRD robustly, in the sense that it is not sensitive to the marginal distributions and *short range dependence* (SRD) characteristics of individual sources. As an example, by integrating DTMW with the RCBR protocol [86], we establish a congestion control scheme which can be easily implemented in real-time. Simulations based on empirical video traces show that our scheme can get nearly optimal performance in terms of utilization as achieved through off-line approach. Because DTMW is not sensitive to the characteristics of individual video streams, users do not need to declare detailed traffic parameters before they are allowed to enter the network. This makes call admission control (CAC) simpler and easier to apply to realistic networks.

4.2 Predicted Service and Predictor

In this section, we will examine in detail two protocols which provide predicted service. They are the *Available Bit Rate* (ABR) service and the *Renegotiated Constant Bit Rate* (RCBR) service. The reason that we select ABR service as an example is because ABR is being standardized by the ATM Forum and will likely see wide usage in the near future [87].

The primary goal of the ABR service is the economical support of applications with vague requirements for throughputs and delays which are best expressed as ranges of acceptable values [85]. The congestion control approach chosen by the ATM Forum as the best match for the goals of the ABR service is to control the bandwidth of connections directly.

One of the earliest closed-loop schemes for rate control was proposed to the ATM Forum by Hluchyj and Yin [88]. Their proposed scheme used Explicit Forward Congestion Indication (EFCI), a code-point in the header of ATM data cells, as a single-bit indicator of congestion in the forward direction of the connection. A node on the connection path will set the EFCI bit upon the detection of a congestion state (typically determined by a buffer threshold crossing). At regular intervals, the destination for the connection would check whether EFCI was set in the most recently received data cell and, if not, would transmit an ATM RM cell back to the traffic source to increase its rate by a fixed increment. If, over an interval of the same length, the source did not receive permission to increase its rate, it would decrease its allowed rate instead by an amount proportional to its current rate. The allowed rate of a connection would adapt between a minimum and a maximum value.

The use of a linear increase and exponential decrease of rates(or recently the exponential increase and exponential decrease of rates [87]) is a heuristic approach to the prediction of traffic. It is more network protection-based than traffic prediction-based. This algorithm typically results in a slow-start and over-shooting iteratively and wastes a certain percentage of bandwidth. To solve this problem, some authors advocated rate-based schemes in which the network would provide the source explicitly with its allowed rate rather than with single-bit feedback. Under some circumstances, this could allow the rate of a source to adapt more rapidly and to oscillate less widely than with single-bit feedback.

Most of the proposed approaches to the calculation of the explicit rate are based on current available bandwidth. Due to the burstiness of traffic, the current available bandwidth may only reflect the available bandwidth in the short term. Direct usage of this information may still cause unnecessary oscillation. Another approach is to combine the explicit rate with EFCI and interpret explicit-rate feedback as a dynamic upper bound (see [85] and references therein). This reduces the oscillation but again is conservative in terms of network utilization.

While the ABR service is potentially useful for a wide variety of applications, the main motivation for its development has been the economical support of data traffic. It does not support different QoS requirements for each individual connection except for a guaranteed minimum bandwidth. In [86], a RCBR (Renegotiated CBR) service discipline is introduced. The basic idea of RCBR is to augment standard (static) CBR service with a renegotiation mechanism. In static CBR service, at the time of call setup, an end-system initiates a signaling message requesting a certain constant bandwidth from the network. In the forward path, each switch performs

an admission control test, and if this is successful, makes a tentative reservation and passes on the call setup message to the next switch along the path. On the reverse path, if all the switches have admitted the call, the tentative reservation is confirmed, and the call is allocated a VCI.

Users of RCBR service are given the option to renegotiate their service rate at any time. Renegotiation consists of sending a signaling message along the path, requesting an increase or decrease of the current service rate. If the request is feasible, the network allows the renegotiation, and upon completion of the request, the source is free to send data at the new CBR rate. During renegotiation, a switch controller does not need to compute routing, allocate VCI or acquire housekeeping records. This reduces the renegotiation overhead.

If a renegotiation fails, a trivial solution is that the source that failed renegotiation can try again. Of course, data will build up in the access queue while the second request proceeds, and there is the possibility of data loss. This may not be acceptable for some users. Such users might reserve resources at or close to the peak rate, so that the frequency of renegotiation is highly reduced, and so is the possibility of renegotiation failure. There is a clear trade-off between buffer size, requested rate and the frequency of renegotiation. In any case, note that even if the renegotiation fails, in contrast to ABR service, the source using RCBR service can keep whatever bandwidth it already has.

Second, because the high frequency of renegotiation failure explicitly expresses the congestion state, during admission control a switch controller might reject an incoming call in order to reduce the frequency of renegotiation failure. This allows

the network operator to trade off call blocking probability and renegotiation failure probability.

While similar to ABR service, since RCBR also provides best-effort bandwidth utilization through dynamic bandwidth allocation, it is different from ABR service in that it provides guaranteed bandwidth for users. With ABR service, there is much less protection among streams, since each user's bandwidth depends on the demand of the others. This feature makes RCBR service ideal for traffic streams with hard real-time requirements.

The two key mechanisms for RCBR are renegotiation and rate prediction. Stored (off-line) and interactive (on-line) applications may use RCBR services differently. Off-line sources can compute the desired series of CBR rates (the optimal renegotiation schedule) in advance, and so renegotiation to increase the service rate can be carried out before actually increasing the data rate. Based on this optimal schedule, analytical and simulation results in [86] have shown that, stored sources which use RCBR service can extract close to maximum multiplexing gain and at the same time keep the cell loss rate at a low value.

For interactive applications, the renegotiation schedule cannot be calculated in advance. Instead, the authors of [86] proposed a heuristic AR(1) filter to monitor the buffer occupancy. For this heuristic AR(1) approach, three parameters have to be tuned: a high and a low buffer threshold B_h and B_l, respectively, and a time constant T, which should reflect the long-term rate of change of the rate function. The rate predictor is

$$\hat{r}_{i+1} = (1 - T^{-1})\hat{r}_i + T^{-1}(r_i + \max\{b_i - B_h, 0\}) \tag{4.1}$$

where r_i is the actual incoming rate during slot i, and b_i is the buffer size at the end of slot i. The additional term $T^{-1}\max\{b_i - B_h, 0\}$ in the estimator adds the bandwidth necessary to flush the current buffer content within T. This is necessary to have a sufficiently fast reaction to sudden large buffer build-ups. Let

$$s_{new} = \lceil \frac{\hat{r}_{i+1}}{\Delta} \rceil \Delta \tag{4.2}$$

with Δ the bandwidth allocation granularity. The algorithm specifies that a new bandwidth s_{new} is requested if

$$(b_i > B_h \text{ and } s_{new} > s) \text{ or } (b_i \leq B_l \text{ and } s_{new} < s) \tag{4.3}$$

Similar approaches can also be found in [84, 89].

Although this AR(1)-based heuristic shows some improvement over static bandwidth allocation, it is still far from optimal as is pointed out in the paper. There are several problems with the heuristic AR(1)-based approach:

(1) AR(1) is a low-pass filter which introduces an inherent unbounded response delay;
(2) AR(1) has only one parameter to adjust for accommodating different traffic streams;
(3) AR(1) is not robust with respect to traffic streams with different characteristics;
(4) AR(1) requires a multiplier which is costly to implement.
(5) The performance of the AR(1) filter is difficult to estimate if the input traffic has non-Gaussian marginal distribution.

We will discuss the above points in detail in later sections. It should be noted that bounded response delay is especially important in real-time applications.

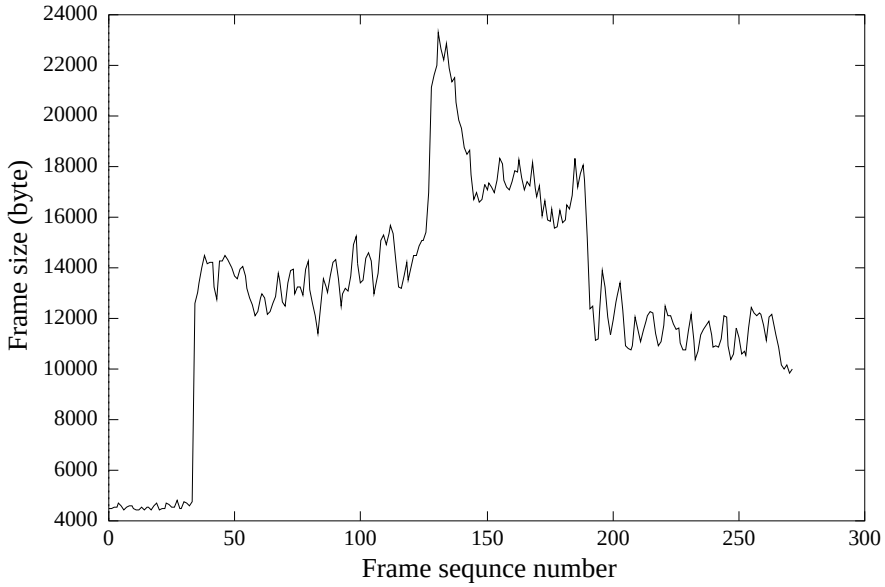

Figure 4.2: Segment of a VBR MPEG video sequence from "BBC News".

Fig. 4.2 shows a segment of a VBR MPEG video trace from the "BBC News". At frame number 35, there is a sharp increase in frame size. For the AR(1) approach, it is difficult to react to this kind of change in time due to the AR(1)'s "infinite average" nature.

In [90] a predictor for Fractal Gaussian Noise (FGN) is devised and shown to have good performance. Unfortunately, it requires a priori knowledge of the Hurst parameter and is limited to the FGN process only. As shown in [5, 6], real traffic has an arbitrary marginal distribution which may deviate far from the Gaussian distribution. Furthermore, in interactive applications, it is not possible to estimate the Hurst parameter beforehand.

In summary,for both ABR and RCBR services, the detection of significant and long-term changes in the characteristics of a connection are key in providing users with the type of performance they expect while allowing for efficient usage of network

resources. In the following parts, we will propose a new traffic prediction algorithm which can help solve the problems above. We will start in an intuitive way and then justify our solution using analytical and simulation results.

4.3 The DTMW Scheme

Intuitively speaking, in the sample path of a process with LRD, high values are more likely to be followed by high values and low values are more likely to be followed by low values. Therefore, in a sense, it is possible to predict future sample values based on past values. Because self-similar traffic displays burstiness at all time scales and because the buffer at the access node can smooth efficiently the burstiness at small time scales, our goal is to predict the burstiness at relatively large time scales. Therefore, the predictor should not be too sensitive so as to respond to short-term burstiness. Based on these observations, we introduce the *Double Threshold Moving Window* (DTMW) detector in the following.

Let N denote the window length, and T_1 and T_2 denote values of the first and second threshold, respectively. Define the detection quantity $S_N(i)$, $i = N, N+1, \ldots$ as follows:

$$S_N(i) = \sum_{j=0}^{N-1} I(Y(i-j) \geq T_1) \tag{4.4}$$

where $I(.)$ is an indicator function and $\mathbf{Y} = \{Y(i),\, i = 1, 2, \ldots\}$ is the input process. If $S_N(i) \geq T_2$, we set the detector output to 1, meaning that congestion is about to occur in the near future, otherwise we set the detector output to 0.

It is easy to see that only a comparator, a single-bit shift register and a counter are needed. An illustrative diagram is shown in Fig. 4.3. The first threshold T_1 is set

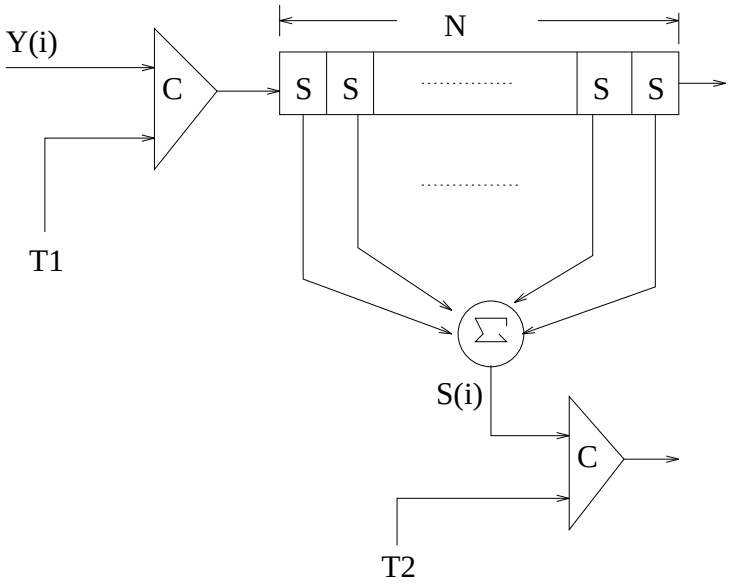

Figure 4.3: Implementation of DTMW.

to detect those high values that are larger than T_1. The T_2-out-of-N criterion is for smoothing burstiness at small time scales and detecting the existence of burstiness at large time scales. In the following paragraphs, we justify the above intuition using an analytical approach.

A general analytical solution for the performance of the DTMW under arbitrary input processes can be difficult if not impossible. But, under certain types of source models, it is possible to give analytical results. An example is the CBR model. It is easy to see that, if the input process $\mathbf{Y}$ is a CBR process, then DTMW will work as predicted in the last section with a maximum initial response delay of N. In the following parts, we analyze the performance of the DTMW under the MTP model we proposed in chapter 2. It has been shown in chapter 2 that this model is general enough to include most of the existing models (e.g., FGN, FARIMA) and can match empirical data up to second-order statistics.

Lemma 2. Let $\mathbf{X} = \{X_i, i = 0, 1, \ldots\}$ be a zero mean, unit variance Gaussian process defined on a probability space $(\Omega, \mathcal{F}, P)$ with autocorrelation function $(r_X(k) : k \geq 0)$. Suppose $r_X(k) \sim k^{2H-2}L(k)$ when $k \to \infty$, where $L(k)$ is a slowly varying function of k and $1/2 < H < 1$. Let $Y = g(X)$, where $g : \mathbf{R} \longmapsto \mathbf{R}$ is a nondecreasing function. If $0 < Pr(Y \geq T_1) < 1$ and $T_1 > 0$, then the process $\mathbf{Z_N} = \{Z_N(j) = (S_{Nj}(Nj) - Pr(Y \geq T_1)Nj)/d_N : j = 0, 1, \ldots\}$ with $d_N^2 \sim \frac{2}{(2H-1)2H}N^{2H}L(N)$ converges weakly as $N \to \infty$ to $J(1)B_H(j)$ where $J(1) = Pr(Y \geq T_1)$ and process $\mathbf{B_H} = \{B_H(t) : t \geq 0\}$ is a Fractional Brownian Motion (FBM) process with parameter H.

Proof: Define $h : \mathbf{R} \longmapsto \mathbf{R}$ such that

$$h(X) = I(Y \geq T_1) = I(g(X) \geq T_1) \tag{4.5}$$

Then we have

$$E(h(X)) = E(I(Y \geq T_1)) = Pr(Y \geq T_1) \tag{4.6}$$

and

$$VAR(h(X)) \leq 1 \tag{4.7}$$

Given that $g(.)$ is a nondecreasing function, we have

$$Pr(Y \geq T_1) = Pr(g(X) \geq T_1) = Pr(X \geq g^{-1}(T_1)) \tag{4.8}$$

Because $0 < Pr(Y \geq T_1) < 1$ and X is a zero mean Gaussian distributed random variable, we have $|g^{-1}(T_1)| < \infty$ and therefore we have

$$\begin{aligned} E(h(X)X) &= E(I(g(X) \geq T_1)X) \\ &= \int_{\{x:g(x)\geq T_1\}}^{\infty} x dP(x) > 0 \end{aligned} \tag{4.9}$$

Therefore we have $E(h(X)X) \neq 0$. Consequently, the Hermite rank of $h(.)$ is one.(See Section 3 of [37] for the definition.) Applying Corollary 5.1. in [37], the proof of the Lemma is complete.□

In the modeling approach of Chapter 2, the function $g(.)$ will be the marginal transformation function which will always be nondecreasing. Other conditions in Lemma 2 will be generally satisfied except in degenerate cases.

From the above Lemma, we have the following conclusions:

(1) For N large enough, the quantity $(S_N(i) - NPr(Y \geq T_1))/d_N$, $i \geq N$, will converge to the Gaussian distribution with variance equal to $J(1)$. This means that, the detection quantity $S_N(i), i \geq N$ is totally decided by the LRD structure and $Pr(Y \geq T_1)$. The influences of SRD structure and marginal distribution are removed by the summation procedure in DTMW.

(2) For N and T_2 large enough, the higher the Hurst parameter H, the larger the probability with which DTMW outputs a value of 1.

These two conclusions justify our intuition in the previous section. While we will demonstrate the usage of DTMW through RCBR service and real-time video applications in the following sections, it should be noted that DTMW can be applied to other types of LRD traffic as well as protocols (e.g., ABR) which also provide predictive services.

4.4 Integration of the DTMW into the Access Node

The selection of the measure of resource congestion has broad implications for the implementation complexity, stability and performance of the corresponding system. The simplest approach, and the one most commonly implemented, consists in observing the instantaneous length of the queue of cells waiting to be transmitted out of a switch port. The switch port is then considered congested whenever the queue length is found to be larger than a given threshold.

Beyond this basic approach, there exist a multitude of alternatives available [85]. A few illustrative examples proposed in the literature include: 1) the use of multiple thresholds on the queue length; 2) the use of the derivative (differential) of the queue length, where an increasing queue reflects more directly an instantaneous bandwidth demand exceeding the available port bandwidth; 3) the explicit estimation of the aggregate bandwidth demand at a port; and 4) the estimation of the variation of the delay perceived in the service of successive cells from the same connection at a switch port.

An interesting theoretical insight on the stability of rate-based congestion control systems is presented by Altman *et al.* [91], who show that queue length information must supplement bandwidth information for stable rate-based control of systems achieving 100% utilization of the available bandwidth.

While our conclusions in the last section are based on monitoring the traffic rate process, they are also valid for monitoring the *queueing* increment process based on the following facts:

Consider a slotted-time single server queue with deterministic service rate μ and an arrival rate process $\mathbf{A}$. Let us define $\delta\mathbf{Q} = \{\delta Q(i) = Q(i) - Q(i-1), i = 1, 2, \ldots\}$ as the increment process of the queueing process which is denoted by $\mathbf{Q} = \{Q(i) : i = 1, 2, \ldots\}$,then by equation 3.1 we have

$$\begin{aligned} Y(i) = A(i) - \mu = \delta Q(i) \geq 0 \ \text{ when } \ A(i) \geq \mu \\ Y(i) = A(i) - \mu \leq \delta Q(i) \leq 0 \ \text{ when } \ A(i) \leq \mu \end{aligned} \tag{4.10}$$

where the process $\mathbf{Y} = \{Y_i : Y_i = A_i - \mu, i = 1, \ldots\}$ is the *netput* process.

So when $T_1 > 0$, $\delta Q(i) \geq T_1$, we must have $\delta Q(i) \geq 0$, and by equation 4.10, we have $\delta Q(i) \geq T_1$ *if and only if* $A(i) - \mu \geq T_1$. Therefore monitoring $\delta Q(i)$ using DTMW is the same as monitoring $A(i)$ differing only by a constant. Thus, our conclusion in the last section can be applied to the queueing increment process.

When we apply the DTMW to the queueing increment process $\delta\mathbf{Q}$, we are in fact monitoring the increasing rate of buffer occupancy. When DTMW outputs a value of 1, it indicates that buffer occupancy is increasing at a rate larger than T_1 per slot. To interpret this in another way, it means that the server needs at least T_1 more bandwidth to keep the buffer occupancy constant. This allows us to integrate DTMW smoothly with RCBR where T_1 can be used as the granularity of the RCBR bandwidth reallocation.

Similarly, we can build a detector which can detect the *decreasing* rate of buffer occupancy as follows:

$$S_N(i) = \sum_{j=0}^{N-1} I(Y(i-j) \leq T_1) \tag{4.11}$$

where T_1 must be a negative value. If $S_N(i) \geq T_2$, we set the detector output to be 1. We will call this detector the *Inverse Double Threshold Moving Window* (IDTMW) detector.

While DTMW can monitor increasing rates which are larger than T_1, rates smaller than T_1 can still drive the buffer into overflow although in a slower fashion. To further control overflow, we can set a "high buffer" boundary. When buffer occupancy reaches this high buffer boundary, we will always request a bandwidth increase by the quantity T_1.

Similarly, to increase the bandwidth utilization, we need to monitor a "low buffer" boundary (which is typically zero), therefore when buffer occupancy reaches the low boundary we will request a bandwidth decrease by the quantity T_1.

The final integrated structure is shown in Fig. 4.4, where HLBC refers to high/low buffer occupancy control. The flow chart for the whole algorithm is shown in Fig. 4.5, where HBC refers to "high buffer boundary check". When buffer occupancy is equal or larger than the high buffer boundary, then $HBC = 1$, otherwise $HBC = 0$. Similarly, LBC refers to "low buffer boundary check". When buffer occupancy is equal or smaller than the low buffer boundary, then $LBC = 1$, otherwise $LBC = 0$.

While bandwidth decrease requests can always be granted without delay, bandwidth increase requests always suffer from a round trip delay. We have taken these factors into account in Fig. 4.5. While the last bandwidth increase request has not been acknowledged, a new bandwidth increase request is not permitted. This prevents excessive bandwidth increase requests being generated due to the round trip renegotiation delay.

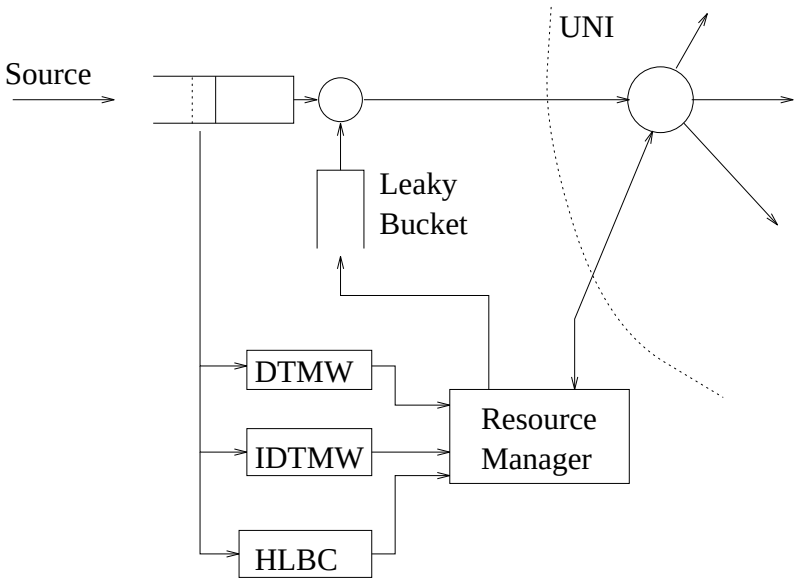

Figure 4.4: Integration of DTMW with RCBR.

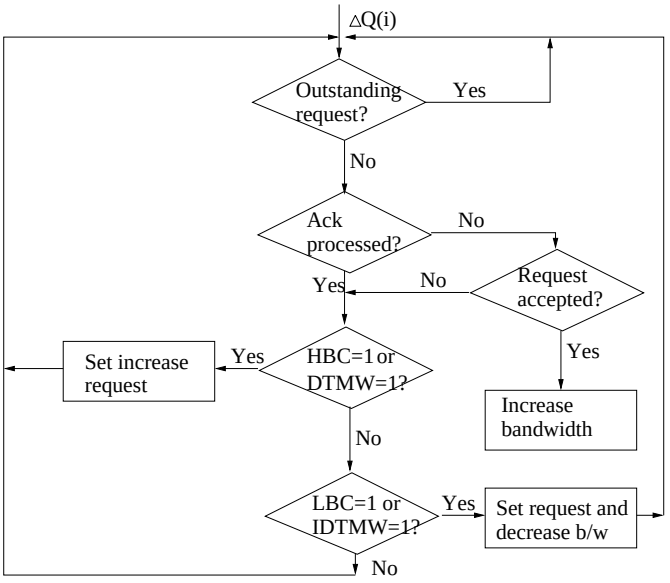

Figure 4.5: Flow chart of integration of DTMW with RCBR.

4.5 Simulation Results

To test our scheme described in the last section, we simulate our algorithm using four empirical video traces, namely, "Last Action Hero", "Ghost", "Star War", and a segment from BBC News that were digitized in our Laboratory. All empirical traces are collected at the frame level. In the following, a slot is equal to a frame interval, i.e., approximately 33ms. Buffer occupancy values are always normalized by the mean arrival rates of corresponding sources(e.g. 6005.2 bytes/frame for I frame only "Last Action Hero"). We consider separately video segments with intraframe compression (I frames) only and video segments with interframe compression (I, B, P frames). For segments with I, B, and P frames, only the "Last Action Hero" and "Ghost" traces are available.

4.5.1 Video with Intraframe Compression Only

As a scenario, in all cases, we will try to control the normalized buffer occupancy to a value below 100. Assume the round trip delay for bandwidth increase renegotiations to be equal to 20 frame intervals (approximately 0.66s) and that bandwidth reallocation requests are always granted.

First we discuss how to select the control parameters in DTMW. Unlike the AR(1) heuristic, DTMW results in a bounded response delay which is equal to the window size N. To control the normalized buffer occupancy within 100, we need to select the window size N plus the round trip delay to a value smaller than 100 so that the response will not be too late. But an excessively small N will reduce the effect of smoothing out the SRD structure. As a compromise, we set $N = 15$.

Because N cannot be set too large, we have to use T_2 to reduce the influence of short term fluctuation. As shown in Lemma 2, the higher the value T_2, the less frequent the renegotiations. So we set $T_2 = N = 15$. For controlling the buffer occupancy below 100 and accommodating a round trip delay at 20 frame periods, the high buffer boundary is set to 80. The parameters of IDTMW are set exactly the same except that the T_1 is negative. The low buffer boundary is set to zero.

The first threshold T_1 of DTMW is more difficult to set. A small T_1 will cause the algorithm to react more frequently, therefore introducing more overhead in terms of bandwidth renegotiations. A large T_1 will make DTMW ineffective most of the time and leave the control burden to the high buffer check which may overreact due to the large T_1. In Fig. 4.6, we plot the frequency of renegotiations versus T_1, for the movie "Last Action Hero". Fig. 4.6 shows that there is a low value around 4000. Notice, however, that, around $T_1 = 4000$, the frequency of renegotiations is in general not sensitive to the value of T_1 over wide ranges. We will further illustrate this conclusion later by using different movies. Based on the results in Fig. 4.6, we set $T_1 = 4000$. To simplify the simulations, the bucket size of the Leaky Bucket in Fig. 4.4 is set to zero.

Fig. 4.7 to Fig. 4.9 depict the results for the movie "Last Action Hero". Fig. 4.7 shows the bandwidth increase/decrease requests versus the queueing process. When the buffer occupancy increases very fast, DTMW requests a bandwidth increase earlier than the time that the buffer occupancy reaches its high boundary. Fig. 4.8 shows the corresponding arrival and departure processes. It can be clearly seen that the service rate tracks the long term arrival rate closely. The histogram of the queueing process is shown in Fig. 4.9. Comparing with Fig. 4.10 which uses CBR service and no control

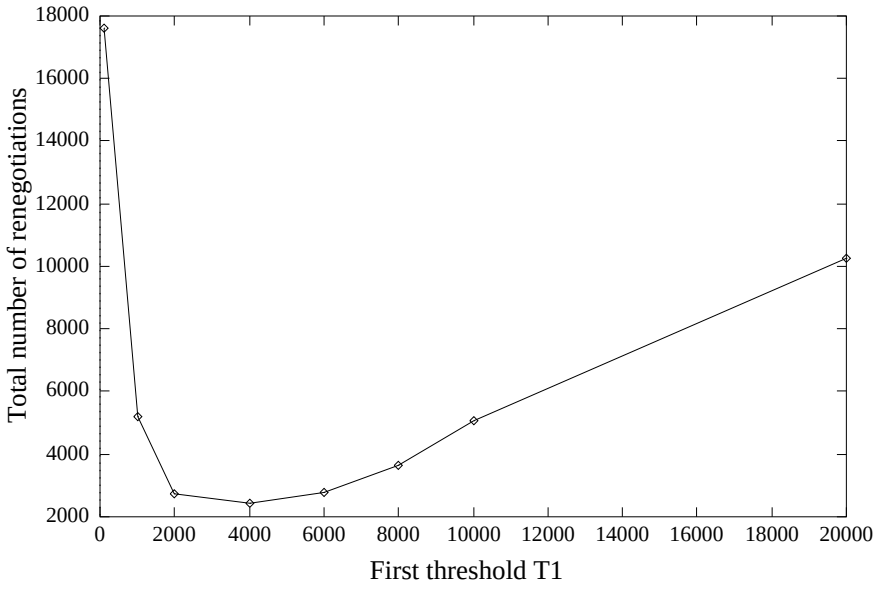

Figure 4.6: Frequency of renegotiations versus threshold T_1 for the video trace "Last Action Hero".

mechanism, the improvement is significant. In Fig. 4.9, most of the normalized buffer occupancy values are within 100, according to our target and the maximum buffer size without any loss is only 350, which is in stark contrast to 18000 under static CBR in Fig. 4.10. Because the probability of the buffer occupancy larger than 100 now is much smaller than in Fig. 4.10, for a cell-loss tolerant application, the overflow traffic can be dropped.

We applied the same set of control parameters to the other three video traces and list all the results in Table 4.1. From Table 4.1, we can see that, for all video clips, the mean utilization is very close to 1 and the maximum buffer occupancy without loss is very close to our target. This shows that DTMW is robust with respect to different traffic streams.

To compare the performance of DTMW with the heuristic AR(1) approach, we apply the AR(1) approach also to the above video traces. As for the simulation of

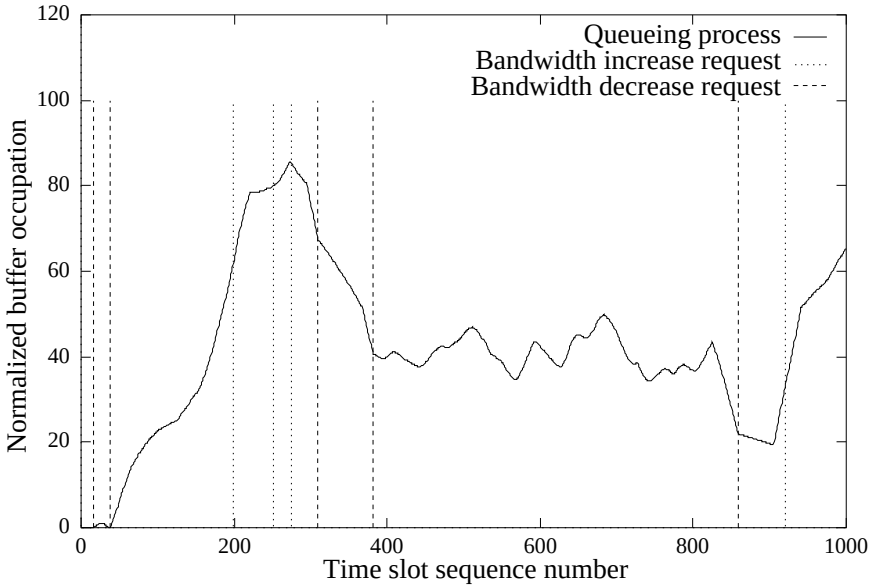

Figure 4.7: Queueing process and bandwidth increase/decrease requests: Utilization=1, RCBR service, DTMW/IDTMW control.

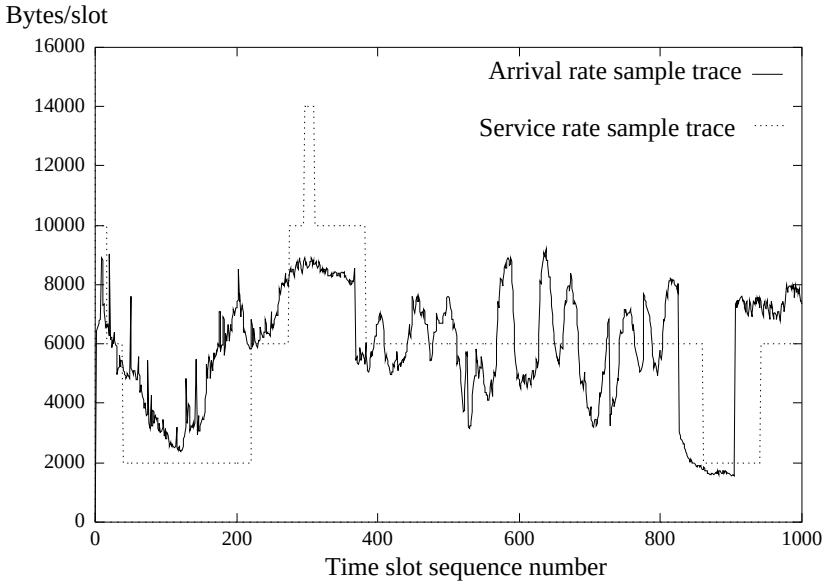

Figure 4.8: Arrival rate process and service rate process: Utilization=1, RCBR service, DTMW/IDTMW control.

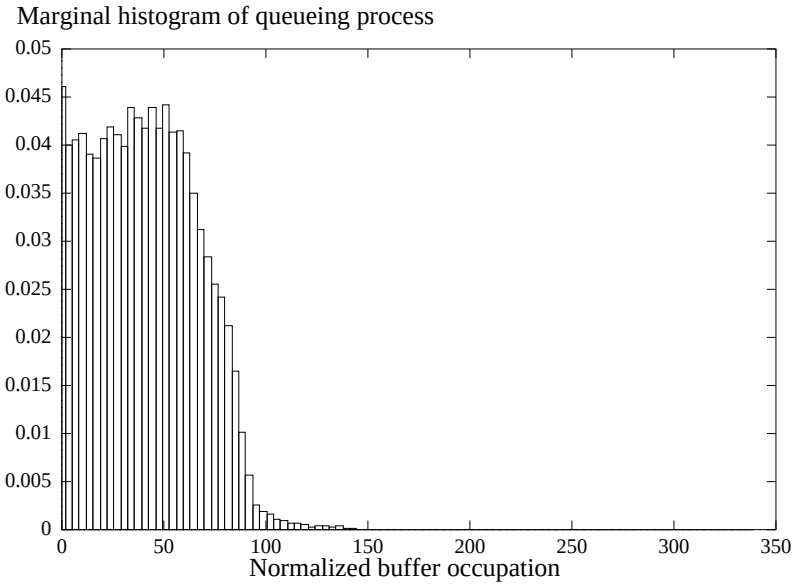

Figure 4.9: Histogram of queueing process: Utilization=1, RCBR service, DTMW/IDTMW control.

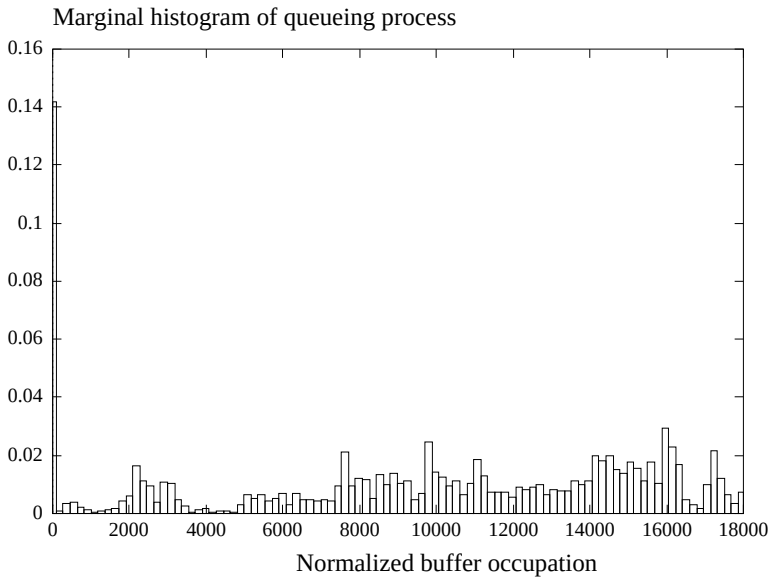

Figure 4.10: Histogram of queueing process: Utilization=1, CBR service, no control.

Video name	Total frames	Mean utilization	Max buffer occupation	Frequency of renegotiations(1/s)	Mean Source rate(b/frame)
LAH	238000	0.9997	340	0.31	6005.2
Ghost	217000	0.9978	200	0.26	9909.0
Star War	170000	1.000	140	0.54	27791.0
BBC News	26000	0.9993	103	0.26	12709.0

Table 4.1: Simulation results for intraframe compression videos using DTMW.

DTMW approach, we chose the parameters for AR(1) approach based on the video trace of "Last Action Hero" and applied the same set of parameters to the other video traces in order to examine the robustness of the AR(1) approach. Similar to the case of DTMW, we set $B_h = 80$ and $B_l = 0$ where B_h and B_l are all normalized by the mean source arrival rate. While the maximum bandwidth increase range for DTMW is the same as the granularity of bandwidth allocation (i.e., the first threshold T_1), they can be significantly different for the AR(1) approach. Higher bandwidth increase ranges are more likely to be rejected by the network.

To provide a fair comparison, we set the maximum bandwidth increase range of the AR(1) approach to be the same as for the DTMW approach by appropriately tuning the value of T. This results in choosing $T = 5000$. A comparison of a source arrival sample trace with the corresponding service rate sample trace is shown in Fig. 4.11 where the slow response nature of the heuristic AR(1) approach is clearly shown.

The final results are shown in Table 4.2. From Table 4.2 we can see that the heuristic AR(1) approach performs much worse than DTMW in terms of utilization (more than 20% lower) and is less robust with respect to different traffic streams.

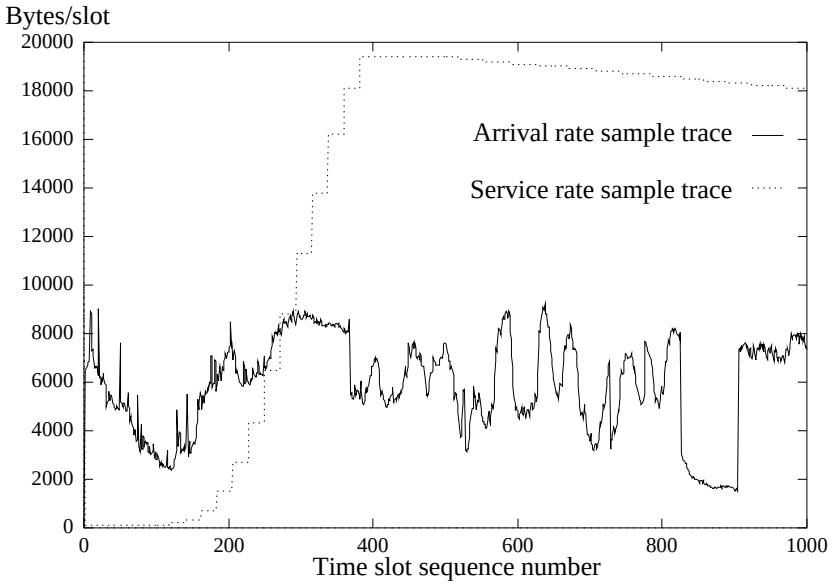

Figure 4.11: Arrival rate process and service rate process: Utilization=0.6, RCBR service, heuristic AR(1) control.

Video name	Max Bandwidth increase range	Mean utilization	Max buffer occupation	Frequency of renegotiations(1/s)
LAH	4000	0.6019	248	0.27
Ghost	4700	0.6386	190	0.36
Star War	11200	0.7743	176	0.51
BBC News	3300	0.6644	140	0.42

Table 4.2: Simulation results for intraframe compression videos using heuristic AR(1).

4.5.2 Video with Intraframe and Interframe Compression

Video traces with both intraframe and interframe compression typically exhibit a strong periodic structure associated with their group of picture (GOP) structure. While this may indicate a non-stationary property, we can still treat it as a stationary process with strong short term burstiness in most cases where GOP sizes are small. DTMW is designed to predict fluctuations in long-range dependent streams, therefore it should also work for video streams with both intraframe and interframe compression. To test this conclusion, we apply DTMW to video streams with both intraframe and interframe compression. While keeping all other parameters the same as in the last section, we only optimize again the parameter T_1 through a similar search based on the video trace of "Last Action Hero" as in the last section. The resulting service rate process is shown together with the source arrival rate process in Fig. 4.12. From Fig. 4.12 we can see that the service rate process tracks the arrival rate process closely.

In [86], assuming a mean renegotiation interval of 1 sec, it is shown that current technology can already handle up to 40,000 RCBR sources. The fact that all our results above require less than one renegotiation per sec on the average, indicates that the overhead of our DTMW scheme is acceptable.

We only have two available video traces with both intraframe and interframe compression. The results for both movies are listed in Table 4.3 where the control parameters are the same for both movies. From Table 4.3, we can see that all utilization values are very close to 1 and the robustness of DTMW is satisfactory.

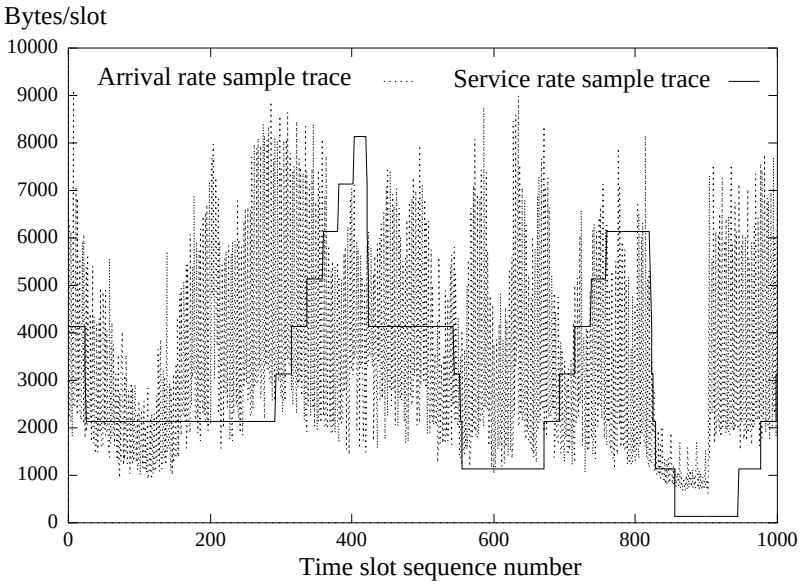

Figure 4.12: Arrival rate process and service rate process: Utilization=1, RCBR service, DTMW/IDTMW control, $T_1 = 1000$.

Video name	Total frames	Mean utilization	Max buffer occupation	Frequency of renegotiations(1/s)	Mean source rate(b/frame)
LAH	238000	0.9940	495	0.60	2481
Ghost	217000	0.9944	420	0.50	4077

Table 4.3: Simulation results for videos with both intraframe and interframe compressions using DTMW.

Chapter 5

Conclusions and Recommendations for Future Research

5.1 Conclusions

Recently extensive measurements have shown that the LRD traffic model is more adequate to capture the natures of various traffic streams than are traditional traffic models. In this thesis, we firstly propose a new traffic modeling approach which combines a direct modeling of both LRD and SRD autocorrelation structures with marginal inversion and Hosking's technique. Through an analytical approach we have shown that, in addition to capturing the marginal distribution of empirical data traces, this approach can also preserve both the SRD and LRD autocorrelation structures. The large class of transformations which have the above transform-invariant nature forms a new class of models (MTP). We then proceed to develop a fast simulation approach for the MTP models based on importance sampling technique. This allows us to efficiently simulate the performance of network with LRD traffic. Extensive simulation results in this thesis have shown that traditional congestion control

schemes cannot handle congestion caused by LRD traffic and result in low network utilization. Instead, we propose in this thesis a new congestion control scheme which fits into a predicted service principle smoothly.

Predicted service provides dynamic bandwidth allocation for traffic streams with tolerant delay and loss requirements. The crucial part of realizing predicted service is a good predictor that can measure traffic and predict future bandwidth in real time. The DTMW scheme we proposed in this thesis can predict bandwidth requirements of LRD traffic through on-line measurement. Analytical and simulation studies employing real video traces have led us to the conclusion that DTMW is flexible and tolerant to different traffic streams in terms of the setting of control parameters. Using DTMW, users do not need to declare detailed traffic parameters, which in most cases is impossible. Typically DTMW gives close to 100% bandwidth utilization and buffer occupancy values that are significantly lower compared to the static CBR case.

5.2 Recommendations for Future Research

Traditional CAC approaches typically use a priori knowledge of different traffic streams and make decisions based on a static rule. In real application, it may be very difficult to get accurate a priori traffic characteristics and rough traffic characterization typically leads to overly conservative decisions. As discussed above, DTMW is robust to different traffic streams. Users of DTMW do not need to declare their detailed traffic behaviors. A high rejection rate for bandwidth renegotiation requests of DTMW is a clear indication of the network congestion state. This information can be used to

make a CAC decision on a dynamic basis. The detailed CAC mechanism based on DTMW and its performance study in a network environment are left for future study.

Appendix A

Proof of Lemma 1

First, we briefly summarize some important results that appear in [8] which are necessary for our results. Due to space restrictions, we restrict our presentation to the very essentials leaving most of the algebraic manipulations to be checked by the interested reader. We start by the following two assumptions:

Hypothesis A [8]: *(i) There exist functions* $a, v : \mathbf{Z}_+ \to \mathbf{R}_+$ *that increase to infinity, such that for each* $\theta \in \mathbf{R}$*, the cumulant generating function defined as the limit*

$$\lambda(\theta) \stackrel{\triangle}{=} \lim_{k\to\infty} v_k^{-1} \log E e^{\theta v_k W_k / a_k} \tag{A.1}$$

exists as an extended real number.

(ii) $\lambda(.)$ *is essentially smooth, lower semi-continuous and there exists* $\theta > 0$ *for which* $\lambda(\theta) < 0$*. Note that* λ *is automatically convex.*

(iii) There exists an increasing function $h : \mathbf{Z}_+ \to \mathbf{R}_+$ *such that the limit*

$$g(c) \stackrel{\triangle}{=} \lim_{k\to\infty} \frac{v(a^{-1}(k/c))}{h_k} \tag{A.2}$$

exists for which $c > 0$*, where*

$$a^{-1}(x) \stackrel{\triangle}{=} \sup\{s \in \mathbf{R}_+ : a(s) \leq x\} \tag{A.3}$$

Hypothesis B [8]: *There exists $d > 0$ such that*

(i)

$$\inf_{c>0} g(c)\lambda^*(c) = \inf_{c>d} g(c)\lambda^*(c) < \infty \tag{A.4}$$

(ii)

$$\lim_{k\to\infty} \inf_{c>d} \frac{\lambda^*(c)v_k}{h(ca_k)} = \inf_{c>d} \lambda^*(c)g(c) \tag{A.5}$$

(iii) for each $\gamma > 0$

$$\limsup_{b\to\infty} h_b^{-1} \log \sum_{k=[a^{-1}(b/d)]}^{\infty} e^{-\gamma v_k} \leq -\inf_{c>0} g(c)\lambda^*(c) \tag{A.6}$$

(iv)

$$\limsup_{b\to\infty} h_b^{-1} \log a^{-1}(b/d) = 0 \tag{A.7}$$

where

$$\lambda^*(x) \stackrel{\triangle}{=} \sup_{\theta\in\mathbf{R}} \{\theta x - \lambda(\theta)\} \tag{A.8}$$

Now, we have the following theorem [8]:

Theorem 2. Suppose that Hypotheses A and B are satisfied, then

$$\limsup_{b\to\infty} h_b^{-1} \log \Pr(Q > b) = -\inf_{c>0} g(c)\lambda^*(c) \tag{A.9}$$

Proof of Lemma 1: Define

$$a_k \stackrel{\triangle}{=} k \tag{A.10}$$

$$v_k \stackrel{\triangle}{=} \frac{k^2}{\sigma_1^2 k^{2H_1} + \sigma_2^2 k^{2H_2}} \tag{A.11}$$

$$h_k \stackrel{\triangle}{=} \frac{k^{2(1-H_1)}}{\sigma_1^2} \tag{A.12}$$

We first check Hypothesis A:

(i) It is easy to see that both a_k and v_k increase to infinity, and

$$\lambda(\theta) = \lim_{k\to\infty} v_k^{-1} \log E e^{\theta v_k W_k/a_k} \tag{A.13}$$

$$= \frac{\theta^2}{2} - \theta\mu \quad for \quad all \quad \theta \in \mathbf{R} \tag{A.14}$$

(ii) It is also easy to check that $\lambda(\theta)$ is a smooth function and there exists $\theta > 0$ for which $\lambda(\theta) < 0$.

(iii) For each $c > 0$, we can show

$$g(c) = \lim_{k\to\infty} \frac{v(a^{-1}(k/c))}{h_k} \tag{A.15}$$

$$= c^{2H_1-2} \tag{A.16}$$

Therefore Hypothesis A is satisfied, and we can easily get

$$\lambda^*(x) = \sup_{\theta\in\mathbf{R}} \{\theta x - \lambda(\theta)\} \tag{A.17}$$

$$= \frac{(x+\mu)^2}{2} \tag{A.18}$$

We now check Hypothesis B: Conditions (i) and (ii) can be checked in a straightforward manner. To check conditions (iii) we note that $\exists K > 0$ such that $\forall k > K$

$$v_k > \frac{k^{2-2H_1}}{2\sigma_1^2} \tag{A.19}$$

Hence,

$$e^{-\gamma v_k} < e^{-\frac{\gamma k^{(2-2H_1)}}{2\sigma_1^2}} \quad for \quad \gamma > 0 \tag{A.20}$$

Conditions (iii) and (iv) follow after some algebra. Then by Theorem 2, our conclusion is proved. □

References

[1] D.R. Cox. Long-Range Dependence: A Review. In H. A. David and H. T. David, editors, *Statistics: An appraisal*. The Iowa State University Press, 1984.

[2] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the Self-Similar Nature of Ethernet Traffic (Extended Version). *ACM/IEEE Transactions on Networking*, 2(1):1–15, Feb. 1994.

[3] Vern Paxson and Sally Floyd. Wide Area Traffic: The Failure of Poisson Modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, 1995.

[4] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger. Long-Range Dependence in Variable-Bit-Rate Video Traffic. To appear on *IEEE Transactions on Communications*, 1994.

[5] M. W. Garrett and W. Willinger. Analysis, Modeling and Generation of Self-Similar VBR Video Traffic. In *Proc. ACM SIGCOMM '94*, London, U. K., Aug. 1994.

[6] C. Huang, M. Devetsikiotis, I. Lambadaris, and A. R. Kaye. Modeling and Simulation of Self-Similar VBR Compressed Video:A Unified Approach. In *Proc. ACM SIGCOMM'95*, Boston, USA, August 1995.

[7] J. R. M. Hosking. Modeling Persistence in Hydrological Time Series Using Fractional Differencing. *Water Resources Research*, 20(12):1898–1908, 1984.

[8] N. G. Duffield and N. O'Connell. Large Deviations and Overflow Probabilities for the General Single-Server queue, with Appplications. *DIAS-STP-93-30*, 1993.

[9] I. Norros. A Storage Model with Self-Similar Input. *Queueing Systems*, 16:387 – 396, Aug. 1994.

[10] R. G. Addie and M. Zukerman. An Approximation for Performance Evaluation of Stationary Single Server Queues. In *Proc. IEEE INFOCOM '93*, 1993.

[11] N. Likhanov, B. Tsybakov, and N. D. Georganas. Analysis of an ATM Buffer with Self-Similar("Fractal") Input Traffic. In *Proc. IEEE Infocom'95*, April 1995.

[12] D. P. Heyman and T. V. Lakshman. What Are the Implications of Long-Range Dependence for VBR-Video Traffic Engineering? *IEEE/ACM Transactions on Networking*, June 1996.

[13] B. B. Mandelbrot. A Fast Fractional Gaussian Noise Generator. *Water Resources Research*, 7:543–553, 1971.

[14] A. Erramilli and R. P. Singh. The Application of Deterministic Chaotic Maps to Characterize Traffic in Broadband Packet Networks. In *Proc. 7th ITC Specialists Seminar*, 1990.

[15] H. E. Hurst. Long-Term Storage Capacity of Reservoirs. *Trans. of the Am. Soc. of Civil Eng.*, 116:770–799, 1951.

[16] D. Ferrari and D. Verma. A Scheme for Real-Time Channel Establishment in Wide-Area Networks. *IEEE J. Select. Areas Commun.*, 8(4):368–379, April 1990.

[17] L. Zhang. VirtualClock: A New Traffic Control Algorithm for Packet Switching Networks. *ACM Transactions on Computer Systems*, 9(2):101–124, May 1991.

[18] A. K. Parekh and G. Gallager. A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks–The Single Node Case. In *Proc. IEEE INFOCOM '92*, pages 915–924, 1992.

[19] A. K. Parekh and G. Gallager. A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks–The Multiple Node Case. In *Proc. IEEE INFOCOM '93*, pages 521–530, 1993.

[20] D. Verma, H. Zhang, and D. Ferrari. Delay Jitter Control for Real-Time Communication in a Packet Switch Network. In *Proc. IEEE TriCOM '91*, pages 35–43, 1991.

[21] H. Zhang and D. Ferrari. Rate-Controlled Static-Priority Queueing. In *Proc. IEEE INFOCOM '93*, pages 227–236, 1992.

[22] S. J. Golestani. Duration-Limited Statistical Multiplexing of Delay-Sensitive Traffic in Packet Networks. In *Proc. IEEE INFOCOM '91*, pages 323–332, 1991.

[23] C. Kalmanek, H. Kanakia, and S. Keshav. Rate Controlled Servers for Very High-Speed Networks. In *Proc. IEEE GlobeCom '90*, pages 300.3.1–300.3.9, 1990.

[24] S. Guo and N. D. Georganas. Resource and Connection Admission Control in Real-Time Transport Protocal with Deterministic QoS Guarantees. In *Proc. IEEE Infocom'95*, April 1995.

[25] Jan Beran. *Statistics for Long-Memory Processes*. Chapman and Hall, 1994.

[26] B. B. Mandelbrot. *The Fractal Geometry of Nature*. Freeman, 1983.

[27] B. B. Mandelbrot and J. R. Walls. Computer Experiments with Fractional Gaussian Noises. *Water Resources Research*, 5:228–267, 1969.

[28] B. B. Mandelbrot and J. W. Van Ness. Fractional Brownian Motions, Fractional Noises and Applications. *SIAM Review*, 10(4):422–437, 1968.

[29] C. W. J. Granger and R. Joyeux. An Introduction to Long-Memory Time Series Models and Fractional Differencing. *J. Time Series Anal.*, 1:15–29, 1980.

[30] J. R. M. Hosking. Fractional Differencing. *Biometrika*, 68(1):165–176, 1981.

[31] B. B. Mandelbrot and J. R. Walls. Some Long-Run Properties of Geophysical Records. *Water Resources Research*, 5:321–340, 1969.

[32] H. E. Hurst. Methods of Using Long-Term Storage in Reservoirs. In *Proc. Institution Civil Engineers*, volume Part I, pages 519–577, 1955.

[33] V. Klemes. The Hurst Phenomenon: A Puzzle? *Water Resources Research*, 10:675–688, 1974.

[34] P. Whittle. Estimation and Information in stationary Time Series. *Ark. Mat.*, 2:423–434, 1953.

[35] R. Fox and M. S. Taqqu. Large-Sample Properties of Parameter Estimates for Strongly Dependent Stationary Gaussian Time Series. *Ann. Statist.*, 14:517–532, 1986.

[36] R. Dahlhaus. Efficient Parameter Estimation for Self-Similar Processes. *Ann. Statist.*, 17:1749–1766, 1989.

[37] M. S. Taqqu. Weak Convergence to Fractional Brownian Motion and to the Rosenblatt Process. *Z. Wahrscheinlichkeitstheorie verw.*, 1975.

[38] D. Heyman, T. V. Lakshman, A. Tabatabai, and H. Heeke. Modeling Teleconference Traffic from VBR Video Coders. In *Proc. IEEE ICC '94*, New Orleans, 1994.

[39] P. Pancha and M. El Zarki. Bandwidth Allocation Schemes for Variable Bit Rate MPEG Sources in ATM Networks". *IEEE Trans. Circ. Syst. Video Tech.*, Vol. 3(3), June 1993.

[40] A. R. Reibman and B. G. Haskell. Constraints on Variable Bit Rate Video for ATM Networks". *IEEE Trans. Circ. Syst. Video Tech.*, Vol. 2(4), Dec. 1992.

[41] P. Skelly, M. Schwartz, and S. Dixit. A Histogram-Based Model for Video Traffic Behavior in an ATM Multiplexer. *IEEE/ACM Trans. on Networking*, 1(4), Aug. 1993.

[42] F. Yegenoglu, B. Jabbari, and Ya-Qin Zhang. Motion-Classified Autoregressive Modeling of Variable Bit Rate Video. *IEEE Trans. Circ. Syst. Video Tech.*, Vol. 3(1), Feb. 1993.

[43] D. Reininger, D. Raychaudhuri, B. Melamed, B. Sengupta, and J. Hill. Statistical Multiplexing of VBR MPEG Compressed Video on ATM Networks. In *Proc. IEEE INFOCOM '93*, San Fransisco, Mar. 1993.

[44] B. Melamed, D. Raychaudhuri, B. Sengupta, and J. Zdepski. TES-Based Video Source Modeling For Performance Evaluation of Integrated Networks. *IEEE Trans. Commun.*, 42(10), Oct. 1994.

[45] M. R. Ismail, I. Lambadaris, M. Devetsikiotis, and A. R. Kaye. Modeling Prioritized MPEG Video Using TES and a Frame Spreading Strategy for Transmission in ATM Networks. In *Proc. IEEE INFOCOM '95*, Boston, April 1995.

[46] C. M. Sharon, M. Devetsikiotis, I. Lambadaris, and A. R. Kaye. Rate Control of VBR H.261 Video on Frame Relay Networks. In *Proc. IEEE ICC '95*, Seattle, June 1995.

[47] B. Melamed and D. Pendarakis. A TES-Based Model for Compressed "Star Wars" Video. In *Proc. Comm. Theory Mini-Conf., IEEE Globecom '94*, San Fransisco, November 1994.

[48] B. Melamed, J. R. Hill, and D. Goldsman. The TES methodology: Modeling empirical stationary time series. In *Proc. of the 1992 Winter Simulation Conference*, New Jersey, 1992.

[49] M. C. Cario and B. L. Nelson. Autoregressive to anything: Time-series input processes for simulation. In *http://primal.iems.nmu.edu/ nelsonb/*, 1996.

[50] R. Addie, M. Zukerman, and T. Neame. Performance of a Single Server Queue with Self Similar Input. In *Proc. IEEE ICC '95*, Seattle, June 1995.

[51] C. Huang, M. Devetsikiotis, I. Lambadaris, and A. R. Kaye. Fast Simulation for Self-Similar Traffic in ATM Network. In *Proc. IEEE ICC'95*, Seattle, USA, June 1995.

[52] D. LeGall. MPEG: A Video Compression Standard for Multimedia Applications. *Communications of the ACM*, 34(4), Apr. 1991.

[53] ISO. *MPEG-1 Specification.* CD 11172.

[54] Sun Microsystems Computer Corporation. *SunVideo 1.0 User's Guide*, Oct. 1993.

[55] Portable Video Research Group, Stanford University. *PVRG-MPEG Codec 1.1*, June 1993.

[56] A. I. McLeod and K. W. Hipel. Preservation of the Rescaled Adjusted Range: 1. A Reassessment of the Hurst Phenomenon. *Water Resources Research*, 14(3):491–508, 1978.

[57] F. L. Ramsey. Characterization of the Partial Autocorrelation Function. *The Annals of Statistics*, 2(6):1296–1301, 1974.

[58] William Feller. *An Introduction to Probability Theory and Its Applications.* John Wiley & Sons, 1971.

[59] J. W. Cohen. *The Single Server Queue.* North-Holland, 1982.

[60] Vaclav E. Benes. *General Stochastic Processes in the Theory of Queues.* Addison-Wesley, 1963.

[61] A. A. Borovkov. *Stochastic Processes in Queueing Theory.* Springer-Verlag, 1976.

[62] E. Wong and B. Hajek. *Stochastic processes in engineering systems.* Springer-Verlag, 1985.

[63] Nol Rananand and Prakash Narayan. Upper-bounds on the Performance of a Queue with Long Range Dependent Input. *Preprint*, 1994.

[64] P. W. Glynn and D. L. Iglehart. Importance Sampling for Stochastic Simulations. *Management Science*, 35(11):1367–1392, Nov. 1989.

[65] J. A. Bucklew. *Large Deviation Techniques in Decision, Simulation, and Estimation.* John Wiley & Sons, 1990.

[66] M. Devetsikiotis and J. K. Townsend. Statistical Optimization of Dynamic Importance Sampling Parameters for Efficient Simulation of Communication Networks. *IEEE/ACM Trans. Networking*, 1(3), June 1993.

[67] P. Heidelberger. Fast Simulation of Rare Events in Queueing and Reliability Models. In *Proc. of Performance '93*, Rome, Italy, October 1993.

[68] A. Dembo and O. Zeitouni. *Large Deviation Techniques and Applications.* Jones and Bartlett, 1993.

[69] P. O. Borjesson and C. E. W. Sundberg. Simple Approximations of the Error Function $Q(x)$ for Communications Applications. *IEEE Transactions on Communications*, pages 639–643, Mar. 1979.

[70] D. Lu and K. Yao. Estimation Variance Bounds of Importance Sampling Simulations in Digital Communication Systems. *IEEE Trans. Commun.*, COM-39(10):1413–1417, Oct. 1991.

[71] D. D. Clark, S.Shenker, and L. Zhang. Supporting Real-Time Applicatiopns in an Integrated Services Packet Network: Architecture and Mechanism. In *Proc. ACM SIGCOMM'92*, 1992.

[72] S. Jamin, P. B. Danzig, S.Shenker, and L. Zhang. A Measurement-based Admission Control Algorithm for Integrated Services Packet Networks. In *Proc. ACM SIGCOMM'95*, 1995.

[73] A. Demers, S. Keshav, and S.Shenker. Analysis and Simulation of a Fair Queueing Algorithm. In *Proc. ACM SIGCOMM'89*, 1989.

[74] R. L. Cruz. A Calculus for Network Delay, Part I: Network Elements in Isolation. *IEEE Trans. Inform. Theory*, 37(1):114–131, Jan. 1991.

[75] C. S. Chang. Stability, Queue Length and Delay, Part II: Stochastic Queueing Networks. In *Proc. IEEE Conf. on Dec. and Control, CDC '92*, pages 1005–1010, Tucson, Arizona, 1992.

[76] J. Y. Hui. Network,transport, and switching integration for broadband communication. *IEEE Network*, Mar. 1988.

[77] F. P. Kelly. Effective bandwidths of multi-class queues. *Queueing Systems*, Vol. 9(1), 1991.

[78] R. J. Gibbens and P. J. Hunt. Effective bandwidths for multitype UAS channel. *Queueing Systems*, Vol. 9(1), 1991.

[79] A. I. Elwalid and D. Mitra. Effective bandwidth of general Markovian traffic sources and admission control of high speed networks. *IEEE/ACM Trans. Networking*, Vol. 1(3), June 1993.

[80] G. Kesidis, J. Walrand, and C.-S. Chang. Effective bandwidths for multiclass Markov fluids and other ATM sources. *IEEE/ACM Trans. Networking*, Vol. 1(4), Aug. 1993.

[81] P. W. Glynn and W. Whitt. Logarithmic asymptotics for steady-state tail probabilities in a single-server queue. *J. Appl. Prob.*, 31, 1994.

[82] D. Anick, D. Mitra, and M. M. Sondhi. Stochastic Theory of Data-Handling Systems with Multiple Sources. *Bell Sys. Tech. Journal*, Vol. 61(8), Oct. 1982.

[83] J. Kurose. On Computing Per-session Performance Bounds in High-Speed Multihop Computer Networks. In *Proc. ACM SIGMETRICS'92*, 1992.

[84] A. A. Lazar and Giovanni Pacifici. Control of Resources in Broadband Networks with Quality of Service Gurantees. *IEEE Communications Magazine*, Oct. 1991.

[85] F. Bonomi and K. W. Fendick. The Rate-Based Flow Control Framework for the Available Bit Rate ATM Service. *IEEE Transaction on Networks*, Mar. 1995.

[86] M. Grossglausser, S. Keshav, and D. Tse. RCBR: A Simple and efficient Service for Multiple Time-Scale Traffic. In *Proc. ACM SIGCOMM '95*, pages 219–230, 1995.

[87] The ATM Forum Technical Committee. Traffic management Specification Version 4.0. Apr. 1996.

[88] M. Hluchyj and N. Yin. On Closed-Loop Rate Control for ATM Networks. In *Proc. INFOCOM'94*, 1994.

[89] I. Cidon, I. Gopal, and R. Guerin. Bandwidth Management and Congestion Control in plaNET. *IEEE Communications Magazine*, Oct. 1991.

[90] I. Norros. On the Use of Fractional Brownian Motion in the Theory of Connectionless Networks. *IEEE JSAC*, 13(6):953 – 962, 1995.

[91] E. Altman, F. Baccelli, and J.C. Bolot. Discrete-Time Analysis of Adaptive Rate Control Mechanisms. In *Proc. 5th Int. Conference on Data Communications.*, 1993.