TES MODELING OF VIDEO TRAFFIC

BENJAMIN MELAMED and BHASKAR SENGUPTA

NEC USA, Inc.
C&C Research Laboratories
4 Independence Way
Princeton, NJ 08540

December 16, 1993

Abstract

Video service is slated to be a major application of emerging high-speed communications networks of the future. In particular, full-motion video is designed to take advantage of the high bandwidths that will become affordably available with the advent of B-ISDN. A salient feature of compressed video sources is that they give rise to autocorrelated traffic streams, which are difficult to model with traditional modeling techniques. In this paper, we describe a new methodology, called TES (Transform-Expand-Sample), for modeling general autocorrelated time series, and we apply it to traffic modeling of compressed video. The main characteristic of this methodology is that it can model an arbitrary marginal distribution and approximate the autocorrelation structure of an empirical sample such as traffic measurements. Furthermore, the empirical marginal (histogram) and leading autocorrelations are captured simultaneously. Practical TES modeling is computationally intensive and is effectively carried out with software support. A computerized modeling environment, called TEStool, is briefly reviewed. TEStool supports a heuristic search approach for fitting a TES model to empirical time series. Finally, we exemplify our approach by two examples of TES video source models, constructed from empirical codec bitrate measurements: one at the frame level and the other at the group-of-block level. The examples demonstrate the efficacy of the TES modeling methodology and the TEStool modeling environment.

Keywords and Phrases: Modeling and Simulation, Software Systems, TES Processes, Stochastic Processes, Autocorrelation Function, Autocorrelated Processes, Video Modeling.

1 INTRODUCTION

Video is an important new service, slated to be supported by emerging high-speed communication networks. VBR (variable bit rate) compressed video is expected to be a major bandwidth consumer, giving rise to autocorrelated traffic streams of bursty nature [12, 15]. Various studies [2, 5, 10, 13] have shown that when autocorrelated traffic is offered to a queuing system, the resulting performance measures are much worse than those corresponding to renewal traffic (which ignores autocorrelations in interarrival times). The growing realization of the impact of dependence in traffic streams on system performance provides the motivation for devising Monte Carlo methods [1] for generating autocorrelated variates with a wide variety of marginal distributions and autocorrelation structures.

In practice, we aim to devise a model having: (i) a marginal distribution that matches its empirical counterpart; (ii) an autocorrelation function that approximates its empirical counterpart; and (iii) sample paths that "resemble" the empirical data. See [9, 14] for some representative references.

In this paper we focus on a class of methods called TES (*Transform-Expand-Sample*) [11, 6, 7, 8]. TES methods can generate a variety of sample paths, and autocorrelation functions with a variety of functional forms (monotone, oscillating and alternating), while simultaneously guaranteeing an exact match of the marginal distribution to its empirical counterpart. In addition, the TES methodology possesses a large degree of freedom in approximating the empirical autocorrelations, even as it maintains the matching of the marginal distributions. Furthermore, the autocorrelation function of a TES model can be computed from fast numerical formulas without requiring simulation, allowing data modeling to be carried out interactively and visually. This observation motivated the implementation of a TES-based modeling package, dubbed TEStool (TE-SU-TSU-RU in Japanese transliteration). TEStool makes heavy use of visualization in order to provide a pleasant interactive environment supporting modeling of general dependent stationary time series. The environment speeds up the modeling search process, cuts down on modeling errors and relieves the tedium of repetitive search. The search is easy to carry out, and can be conducted by experts and non-experts alike.

This paper is organized as follows. Section 2 contains a brief survey of TES processes, and an outline of the TES modeling methodology. Section 3 overviews the TEStool modeling environment. Section 4 demonstrates the efficacy of the TES modeling methodology and the TEStool modeling environment by an example. Finally, Section 5 contains the conclusion of this paper.

2 TES PROCESSES AND MODELING METHODOLOGY

TES is a class of models for generating autocorrelated random sequences with arbitrary marginals [11, 6, 7, 8]. TES processes are a variation on classical autoregressive schemes, but with modulo-1 arithmetic. TES generation methods are easy to implement on a computer, enjoying a low computational complexity.

2.1 TES Processes

For any real x, let $\lfloor x \rfloor = \max\{n \text{ integer} : \ n \leq x\}$ be the integral part of x, and $\langle x \rangle = x - \lfloor x \rfloor$ be the fractional part of x. Let $\{V_n\}$ denote a sequence of i.i.d. (independent identically distributed) random variables with a common, though arbitrary, density f_V. Further, let U_0 be uniform on [0,1) and independent of each element of the sequence $\{V_n\}$. The random variables V_n are referred to as *innovations*.

Pure TES processes are stationary Markovian and come in two flavors: TES$^+$ giving rise to a sequence $\{U_n^+\}$, and TES$^-$ giving rise to a sequence $\{U_n^-\}$. The definition of $\{U_n^+\}$ is given recursively by

$$U_n^+ \ = \begin{cases} U_0, & \text{if } n = 0 \\ \\ \langle U_{n-1}^+ + V_n \rangle, & \text{if } n > 0 \end{cases} \tag{1}$$

and the sequence $\{U_n^-\}$ is defined in terms of $\{U_n^+\}$ by

$$U_n^- \ = \begin{cases} U_n^+, & n \text{ even} \\ \\ 1 - U_n^+, & n \text{ odd} \end{cases} \tag{2}$$

The superscripts in (1) and (2) are suggestive of the fact that TES methods achieve coverage of the full range of feasible lag-1 autocorrelation [6]; TES$^+$ methods cover the positive range $[0, 1]$, while TES$^-$ methods cover the negative range $[-1, 0]$. It can also be shown [6] that Eqs. (1) and (2) each give rise to a sequence of stationary random variables with uniform marginals on $[0, 1)$ and a wide variety of autocorrelation structures. However, in practice one is usually interested in transformed TES processes $\{X_n^+\}$ and $\{X_n^-\}$ obtained from Eq. (1) or Eq. (2) by some transformation D (called a *distortion*), i.e.,

$$X_n^+ \ = \ D(U_n^+), \quad X_n^- = D(U_n^-). \tag{3}$$

Uniform TES sequences $\{U_n^+\}$ and $\{U_n^-\}$ of the form (1) and (2) are called *background* sequences, whereas their distorted counterparts $\{X_n^+\}$ and $\{X_n^-\}$ of the form (3) are called *foreground* sequences.

The transformation D, and the fact that the $\{U_n\}$ all have a uniform marginal distribution on [0,1), allow us to generate random sequences with essentially arbitrary marginals by using the inversion method [1]; for a given distribution function F, the inversion method uses $D = F^{-1}$ (appropriately defined if F is not strictly monotone). The resultant sequences $\{X_n^+\}$ and $\{X_n^-\}$ are both stationary with marginal distribution F. In particular, this allows us to match the marginal distribution of empirical data, modeled by a histogram.

Our goal is to construct a model whose autocorrelation function approximates that of the empirical data. Naturally, a Monte Carlo simulation of Eqs. (1) and (2) can always provide an estimate of the model's autocorrelation function, if a sufficient sample size is generated. This approach, however, can be costly in terms of time complexity, especially when high autocorrelations necessitate large sample sizes for adequate statistical reliability. It is, therefore, essential to develop fast and efficient computational techniques for these calculations. Reference [6] provides numerically computable formulas for the autocovariance functions of $\{X_n^+\}$ and $\{X_n^-\}$: for a given lag τ, the autocorrelation functions $\rho_X^+(\tau)$ of $\{X_n^+\}$ and $\rho_X^-(\tau)$ of $\{X_n^-\}$ are given respectively by

$$\rho_X^+(\tau) \;=\; \frac{2}{\sigma_X^2} \sum_{\nu=1}^{\infty} \Re[\tilde{f}_V^{\tau}(i2\pi\nu)]\, |\tilde{D}(i2\pi\nu)|^2 \tag{4}$$

and

$$\rho_X^-(\tau) \;=\; \begin{cases} \rho_X^+(\tau), & \tau \text{ even} \\ \\ \dfrac{2}{\sigma_X^2} \displaystyle\sum_{\nu=1}^{\infty} \Re[\tilde{f}_V^{\tau}(i2\pi\nu)]\, \Re[\tilde{D}(i2\pi\nu)^2], & \tau \text{ odd} \end{cases} \tag{5}$$

where σ_X^2 is the common variance of $\{X_n^+\}$ and $\{X_n^-\}$, tilde denotes the Laplace Transform, f_V^{τ} is the τ-fold convolution of the innovation density f_V, $\Re$ denotes the real part operator, and $i = \sqrt{-1}$. It is worth noting that the effects of the innovation and distortion are conveniently separated in Eqs. (4) and (5). More importantly, the series converge rapidly, since the convolution factors decay rapidly in the lag.

Since the marginal distributions obtained from empirical data are often modeled by an empirical histogram H, one frequently uses the *histogram distortion*

$$D_H(x) \;=\; \sum_{k=1}^{N} 1_{[C_{k-1},C_k)}(x)\, [\ell_k + (x - C_{k-1})\, \frac{w_k}{p_k}], \quad 0 \le x \le 1, \tag{6}$$

where N is the number of histogram cells of the form $[\ell_k, r_k)$, $w_k = r_k - \ell_k$ is the width of cell k, 1_A is the indicator function of set A, p_k is the probability of cell k, and $C_k = \sum_{j=1}^{k} p_k$ is the cumulative distribution of $\{p_k\}$ ($C_0 = 0$ and $C_N = 1$). Recall that a histogram H is, mathematically, a step function density, i.e., a probabilistic mixture of uniform densities. Thus, H represents the density of a uniform random variable over $[C_{k-1}, C_k)$ with probability $p_k,\ k = 1, \ldots, N$. Reference [7] contains formulas for the Laplace transform of various distortions, including the quantities $\tilde{D}_H(i2\pi\nu)$ corresponding to the histogram distortion (6); these are needed to calculate numerically the autocorrelation functions (4) and (5).

Intuitively, the modulo-1 arithmetic used in defining TES processes in Eq. (1) has a simple geometric interpretation as a random walk on the unit circle (circumference 1), with random step size V_n. To fix the ideas, consider the important class of step function innovation densities f_V, given by

$$f_V(x) \;=\; \sum_{k=1}^{K} 1_{[L_k,R_k)}(x)\, \frac{P_k}{R_k - L_k}, \tag{7}$$

corresponding to mixtures of uniform densities whose support is contained in [-0.5,0.5). These densities can be specified by triplet sets of the form $\{(L_k, R_k, P_k)\}_{k=1}^{K}$, where K is the number of triplets, and L_k and R_k are the boundaries of cell k, satisfying $-0.5 \le L_k < R_k \le 0.5$, and $0 < P_k \le 1$ is the mixing probability of triplet k ($\sum_{k=1}^{K} P_k = 1$). We point out that any interval of length one can be used to support the innovation densities, due to the modulo-1 arithmetic, inherent in the definition of TES processes. However,

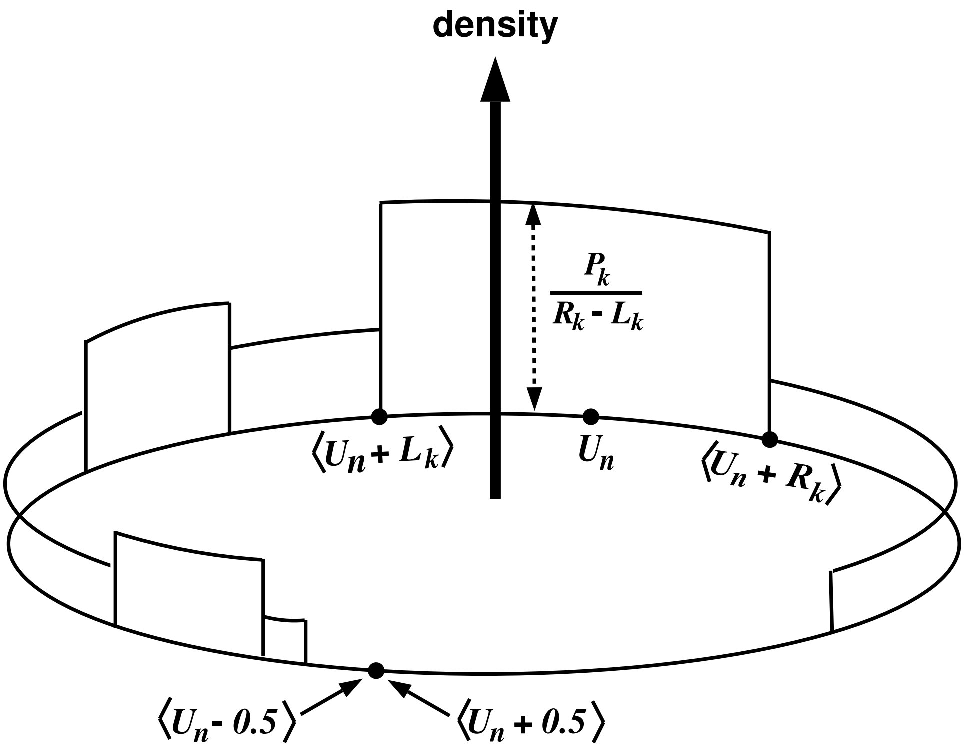

Figure 1: A 3-Dimensional Representation of a Step Function Density on the Unit Circle.

the symmetry of the interval $[-0.5, 0.5)$ about zero is intuitively compatible with the interpretation of the innovation variates as modulo-1 increments of the TES$^+$ process on the unit circle. For convenience, we also require that the triplets do not overlap (i.e., $R_k \le L_{k+1},\ k = 1, \ldots, K-1$). Figure 1 depicts a 3-dimensional representation of this parameterization on the unit circle. Here the steps are arranged on the surface of an imaginary cylinder resting on the unit circle. Observe that step function densities can approximate arbitrarily closely any density, while retaining an ease of specification via the triplets (L_k, R_k, P_k).

The sample paths of $\{X_n^+\}$ and $\{X_n^-\}$ (shown in [11, 6], for example) often exhibit a visual "discontinuity" whenever the corresponding uniform random walks $\{U_n^+\}$ and $\{U_n^-\}$ "cross" point 0 on the unit circle in either direction. In the cyclical case (see ibid.), sample path cycles are skewed to a sawtooth shape (either uptrending or downtrending, depending on the sign of $E[V_n]$). In order to "smooth" TES sample paths, or to skew sample path cycles variously, one can use the so-called family of *stitching transformations* $S_\xi,\ 0 \le \xi \le 1$, defined by

$$S_\xi(y) = \begin{cases} \dfrac{y}{\xi}, & \text{if } 0 \le y < \xi \\[2ex] \dfrac{1-y}{1-\xi}, & \text{if } \xi \le y < 1 \end{cases} \tag{8}$$

Processes of the form $\{S_\xi(U_n^+)\}$ and $\{S_\xi(U_n^-)\}$ are called *stitched* TES processes. Note that for $0 < \xi < 1$, the S_ξ are continuous on the unit circle; in particular $S_\xi(0) = S_\xi(1)$. It can be shown [6] that all $S_\xi,\ 0 \le \xi \le 1$, preserve uniformity; consequently, stitched TES processes can still be distorted to arbitrary marginals.

A great deal is known about the qualitative behavior of sample paths and autocorrelation functions generated by TES processes, as a function of the stitching parameter and innovation density parameters. These are discussed in some detail in references [11, 6].

2.2 The TES Modeling Methodology

The TES modeling methodology is geared towards fitting TES models to empirical data [3]. Typically, a modeler would start out with some empirical sample path data representing a partial process history (e.g., bitrates on some link in a communications network). A complete specification of a TES model requires the modeler to specify a flavor of TES process (TES$^+$ or TES$^-$), a distortion D, a stitching parameter ξ and an innovation density f_V. The interaction process comprising the TES modeling methodology and leading to these choices is now briefly outlined.

Selecting the TES Process Sign. The selection of the TES process sign is based on the modeler's experience and familiarity with fundamental sample path behavior of TES processes. Most real-life data call for a TES$^+$ model. In contrast, TES$^-$ models are rarely used; they are called for when the sample path has an alternating (zigzag) nature.

Selecting the distortion. As mentioned in Section 2.1, the distortion transformation D takes a TES process, which is marginally uniform on $[0, 1)$, and transforms each element in the time series via a distortion D. When modeling empirical data, D corresponds to the inverse distribution function of the empirical data, obtained from the empirical density function (itself modeled as a histogram); in other words, $D = D_H$ as given in Eq. (6). Reference [7] provides formulas for the Laplace transform of the histogram distortion (6), as well as for a variety of other distortions corresponding to distortions, such as uniform, exponential and geometric.

Selecting the Stitching Parameter and innovation Density. The core activity of the TES modeling process is a heuristic search for a suitable stitching parameter and innovation density. Recall that $\{X_n^+\}$ and $\{X_n^-\}$ always have the requisite distribution F *regardless* of the innovation density f_V and stitching parameter ξ selected. However, different innovation densities and/or stitching parameters usually give rise to different autocorrelation functions, thereby providing a large degree of freedom in fitting autocorrelation functions. The modeler searches through stitching parameters in the range $[0, 1]$ and innovation densities in the space of step function densities whose support is contained in $[-0.5, 0.5)$.

The search activity outlined above is supported by the TEStool modeling environment to be described next. See [3] for more details.

3 THE TEStool MODELING ENVIRONMENT

TEStool is a visual interactive software environment for fitting TES models to empirical data. It provides services to generate and modify TES models, and to examine their statistics, mainly in the context of a heuristic search for approximate autocorrelations and "similar" sample paths.

TEStool classifies data into three types. An *empirical* dataset contains data whose source is external to TEStool. The data consist of an empirical time series or statistics computed from it (histogram, autocorrelation function or spectral density). A *simulated* dataset contains data whose source is a Monte Carlo simulation of a TES model. The data consist of a sample path realization or statistics computed from it (histogram, autocorrelation function or spectral density). A *numerical* dataset contains the results of a numerical computation of the autocorrelation function, or the spectral density of a TES model, calculated via Eqs. (4) and (5).

Figure 2 displays a typical TEStool screen comprised of four subwindows. The upper-left subwindow contains sample paths, the upper-right subwindow contains histograms, the lower-left subwindow contains autocorrelation functions, and the lower-right subwindow contains a joint specification of a TES sign, stitching parameter and an innovation density. The buttons at the top screen and at the bottom of each subwindow control various modeling services. These include reading and writing datasets, subdividing the screen real estate, opening a TES specification window or menu, performing various computations and quitting the system. The most important service is the visual specification of a TES model and the interactive computations associated with it.

A TES model can be specified in TEStool either in standard text mode (by filling out text fields in a form) or graphically. The advantage of a graphical specification is that it is more intuitive; more importantly, it enables the user to interact flexibly with the environment, significantly increasing modeling efficiency and substantially decreasing modeling tedium. Figure 3 displays a graphical model specification subwindow. In the top part, the TES sign is selected by a button, the stitching parameter ξ is selected by a slider, and the distortion is selected from a menu, including the histogram distortion (6), constructed from the empirical data histogram. The bottom part contains a graphical specification of a step function density, drawn essentially as a 2-dimensional version of the representation in Figure 1. To this end, the user uses the mouse to draw non-overlapping rectangles (representing the steps of a step function density) on a square planar grid whose X values are between -0.5 and 0.5, and whose Y values are between 0 and 1. Each cell represents an (L_k, R_k, P_k) triplet in the following manner: The base of a step k always rests on the Y axis. The left and right step edges correspond to L_k and R_k, respectively, while the height is P_k. Note that technically speaking, this representation is not a mathematical density, since the area under the curve does not sum to one. It is, however, a consistent representation of a step function density with one distinct advantage: it is easier to manipulate graphically, since its height is bounded by one, whereas an ordinary density is unbounded. TEStool allows the user to add and modify as many steps as desired, as long as the sum of the P_k does not exceed one.

An important feature of TEStool is the way it reacts to a change in the model specification subwindow. Once a model is specified, the user can set selected subwindows to *interactive mode*. Thereafter, any change in the model specification subwindow, resulting in a legal

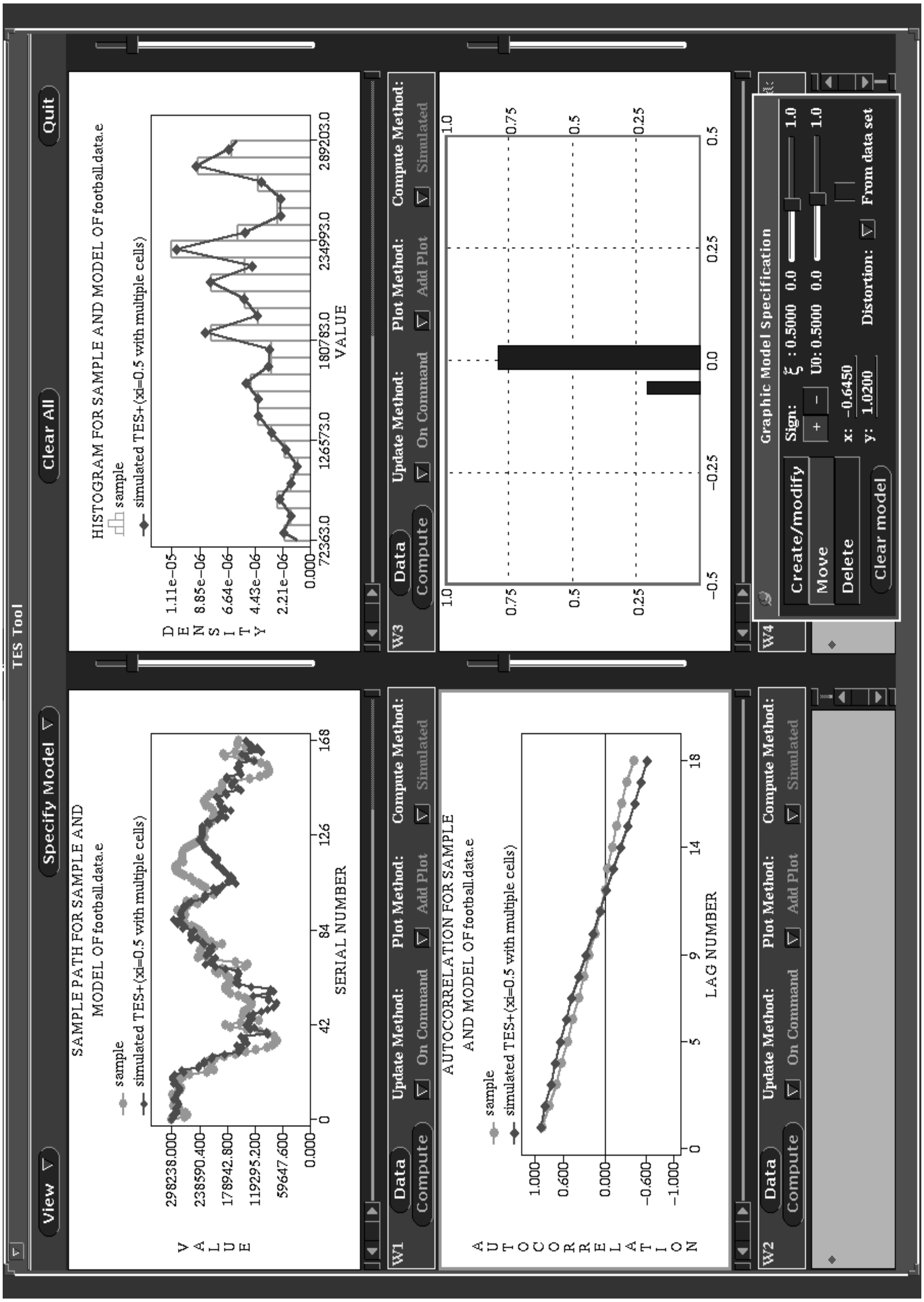

Figure 2: A Typical TEStool Screen.

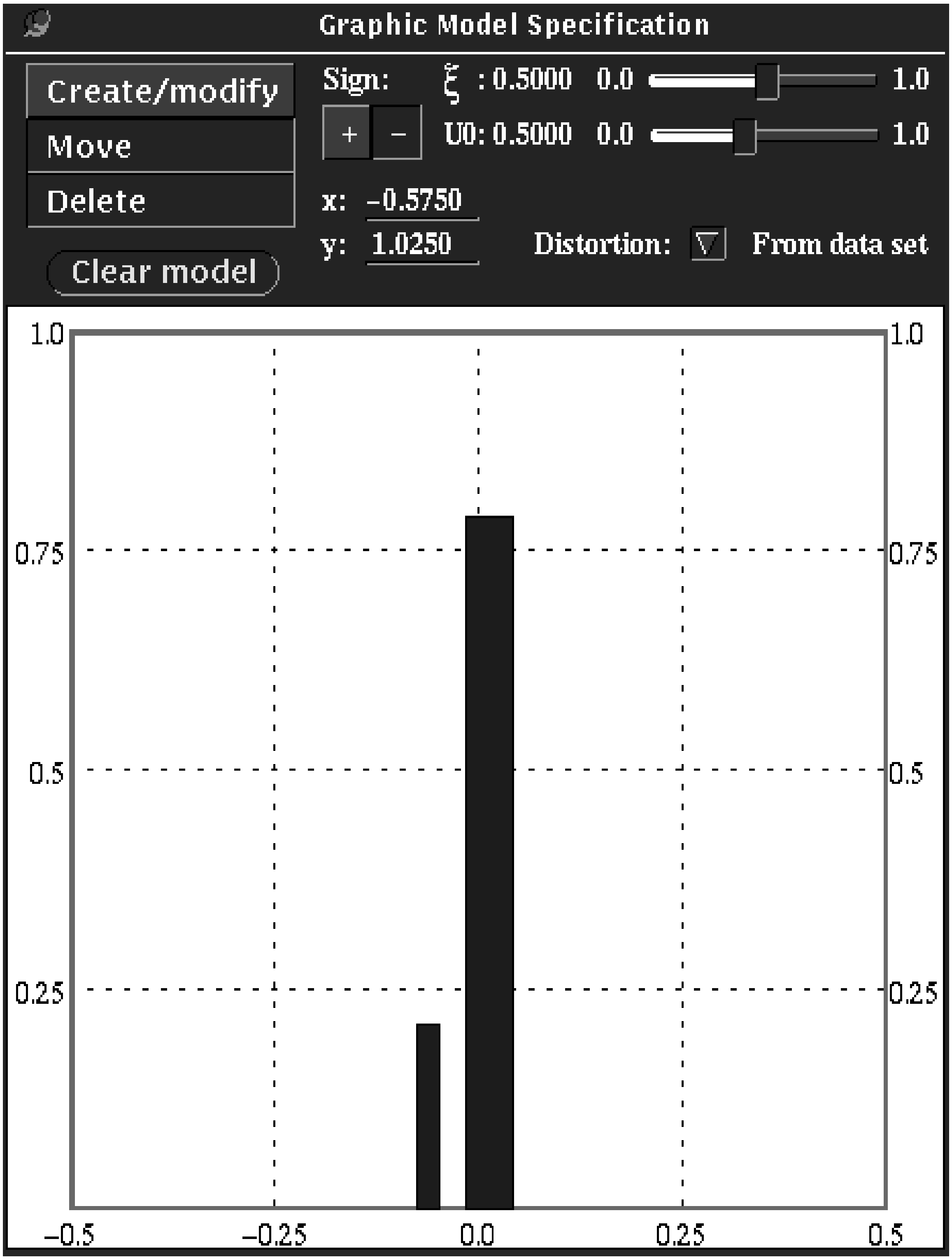

Figure 3: A TEStool Graphical Model Specification Subwindow.

innovation (i.e., $\sum P_k = 1$), will trigger an immediate recomputation and redisplay of the corresponding subwindow. This feature is typically used to recalculate statisics and compare them to their empirical counterparts for goodness of fit. The ability to continually change a TES model and the immediate feedback that follows serve to facilitate the assessment of progress in the search, and aid in deciding on the next perturbation. The user can quickly acquire an understanding of how a change in innovation parameters affects the model's sample paths and statistics; this understanding is both qualitative and quantitative. Furthermore, the search task can be easily delegated to a non-expert, since it largely eliminates the need to understand the mathematical details in Section 2.1. It also lets an expert user experiment with TES processes and possibly extract rules to guide the search process.

4 EXAMPLES OF TES SOURCE MODELS FOR VBR VIDEO

In this section, we demonstrate the efficacy of our approach via examples of source modeling of compressed VBR video. The following subsections briefly describe two case studies which employed the TES modeling methodology: the first example models a VBR video stream at the frame level, and the second at the group-of-block level. For more details see [15] and [12], respectively.

4.1 Frame Source Model for Compressed VBR Video

In the first example, we applied the TES modeling methodology to frame bitrate data, generated by a video coding algorithm which employs a layering technique [17]. The layered algorithm considered there uses motion compensation on low-frequency coefficients of the Discrete Cosine Transform (DCT) to improve data compression. The high-frequency coefficients are transmitted intraframe at low priority. The algorithm separates the video data into high and low priorities such that the loss of low-priority data will not generate errors in the received high-priority data. The main purpose of the layering scheme is to allow the network to drop low-priority bits in the event of congestion. Such a selective dropping scheme would cause less damage to the reconstructed video image when compared to a scheme in which the network is allowed to drop *any* packets it chooses. In this algorithm, high-frequency video data is assigned low priority, and may be lost in the network. Low-frequency video data is essential, and is assigned high priority.

We now outline the TES methodology employed in constructing the source model of the video bitrate stream generated by this encoder. Figure 4 depicts the empirical bitrate sample path (corresponding to individual frames), which was used to construct the TES source model (light shade); superimposed on it is a typical sample path generated by a Monte Carlo simulation of this TES model (dark shade). The corresponding histogram and autocorrelation function are plotted against their empirical counterparts in the two subwindows shown in Figure 5. Having extracted the empirical histogram (see the light-shaded histogram in the left subwindow of Figure 5), the distortion of Eq. (6) was constructed in a natural way. The empirical bitrate sequence in Figure 4 suggested a foreground distorted

Figure 4: A TEStool Sample Path Subwindow.

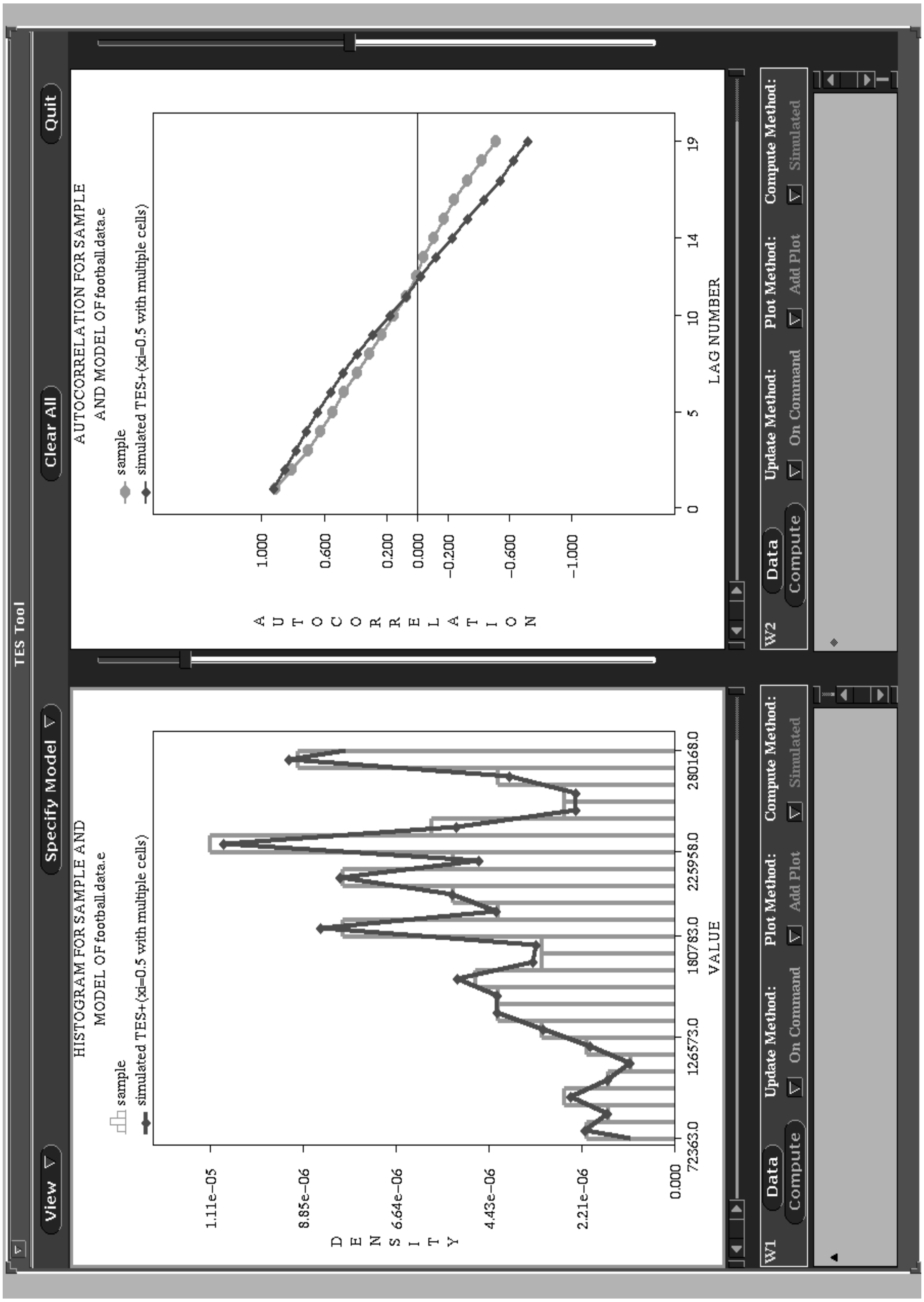

Figure 5: A TEStool Histogram and Autocorrelation Function Subwindows.

TES$^+$ process $\{X_n^+\}$ of the form (3). The corresponding background TES process $\{U_n^+\}$ used the innovation density shown in Figure 3, consisting of a mixture of two uniform distributions. Finally, the symmetric shape of the empirical sample path in Figure 4 led to the choice of a stitching transformation S_ξ with $\xi = 0.5$. This completes the specification of $\{X_n\}$ as a model for the compressed video bitrate traffic.

The particular selection of the innovation density parameters was carried out through a heuristic search for an acceptable fit of a TES model autocorrelation function and acceptable resemblance of its sample path, when compared against their respective empirical counterparts. The final results are shown in Figures 4 and 5. Recall that the TES methodology always guarantees that a TES model marginal distribution will automatically have an exact match with its empirical counterpart for any innovation density, as evidenced by the left subwindow in Figure 5. This is a distinct advantage over first order autoregressive schemes in which it may be impossible to match an arbitrary marginal distribution. We systematically varied the parameters K, L_k, R_k and ξ, guided by the qualitative behavior summarized earlier. The search could proceed rapidly, because the model autocorrelation function was computed numerically from Eq. (4), with the specializations of the Laplace transform of the histogram distortion (6) and the step function density (7), both of which were computed in [7]. Since an average scene lasts for only a few seconds and the data available consisted of 210 frames, it only made sense to match the first 10 autocorrelations or so. Naturally, determining the resemblance of sample paths is a highly heuristic decision based on subjective judgment. Nevertheless, it is worth trying, since similarity of sample paths would increase the confidence in the model. We emphasize that this is a qualitative matching which is done in addition to, not instead of, the quantitative matching of marginal distributions and autocorrelation functions.

We remark that the two-step innovation density reported here is an improvement on the single-step innovation density reported in the original study [15]. This may be verified by noting that both the simulated sample path and autocorrelation function depicted in Figures 4 and 5, respectively, exhibit a better fit to the empirical data than their counterparts in [15]. Indeed, the original model was created painstakingly from repeated batch runs without the benefit of the visual facilities of TEStool. Furthermore, the innovation density search for the original model took about a week, whereas the model reported here was constructed in under an hour.

4.2 Group-Of-Block Source Model for Compressed VBR Video

In the second example, we applied a variant of the TES methodology to VBR video encoded in accordance with the preliminary H.261 recommendations of the CCITT Study Group XV [16]. The encoding is based on a motion compensated intra/interframe discrete cosine transform (DCT) algorithm and the frame rate is 30 frames/sec. See [12] for details. Each coded frame is divided into 12 group-of-block (GOB) coded subscreens, and each GOB consists of 33 Macro Blocks (MB) which are scanned line by line. Each resulting encoded GOB is a suitable unit for packet transport, since it represents a self-contained block of data. The VBR codec considered in this study was configured to nominally generate an average of 128 kbps for a typical picturephone image. At this level, a GOB varies between about 500 and 5000 bits (with an average of some 2000 bits) - generally within the range of acceptable packet size for the IEEE 802.3 LAN. Occasionally, a GOB will exceed the maximum link level packet size limit of the LAN, requiring transport level segmentation and reassembly. Each compressed video stream was modeled as a GOB bitrate sequence.

We next describe the procedure used to fit a model to the empirical measurements of coded GOB bitrate. Figure 6 displays the raw empirical data $\{\hat{X}_n\}$, where each observation is the bitrate of a coded block within a frame. The bullets in Figure 7 represent the empirical autocorrelation function $\hat{\rho}_X(\tau)$ computed from the empirical data for lags $1 \le \tau \le 100$. The cyclical nature of the empirical sample path and the attendant autocorrelation function is clearly evident; each cycle corresponds to a run of twelve observations comprising the coded blocks of a single frame. Since every twelfth block codes the same screen portion of a frame, we expect to see a strong deterministic periodicity of twelve in the data. This is indeed the case, as evidenced by the same periodicity of the empirical autocorrelation function and its sustained amplitude. Since TES autocorrelation functions decay much faster, it became clear that a pure TES model could not be properly fitted to the raw empirical data. Rather, the raw data would have to be transformed first, so as to remove the deterministic periodic components. To this end, we performed a periodogram analysis [4] and estimated from the raw data a set of parameters J, ω_i, A_i and B_i, $1 \le i \le J$, such that the trigonometric sums $\{\hat{T}_n\}$, given by

$$\hat{T}_n = \sum_{i=1}^{J} \left[A_i \cos(\omega_i n) + B_i \sin(\omega_i n) \right], \tag{9}$$

tracked well the oscillations of the raw data $\{\hat{X}_n\}$. We then transformed $\{\hat{X}_n\}$ into a new sequence $\{\hat{R}_n\}$ (called the *residuals sequence*), where

$$\hat{R}_n = \hat{X}_n - \hat{T}_n. \tag{10}$$

The residuals sequence had a much faster decaying autocorrelation function as compared to $\hat{\rho}_X(\tau)$, and a considerably reduced magnitude. Based on the appearance of the residuals sample path, we fixed ξ at 0.5 and performed a search of the parameter space $\{(L, R) : -0.5 \le L < R < 0.5\}$ for a good TES model $\{U_n^+\}$, such that

$$Y_n = D_H(S_{0.5}(U_n^+)) \tag{11}$$

had the same marginal distribution as $\{\hat{R}_n\}$, and the first few autocorrelations of $\{Y_n\}$ were a good match to their $\{\hat{R}_n\}$ counterparts (we found that a single-step innovation was quite adequate). The final step was to add back the trigonometric components $\{\hat{T}_n\}$ to $\{Y_n\}$, leading to the final traffic model of coded video GOB bitrate

$$X_n = D_H(S_{0.5}(U_n^+)) + \hat{T}_n, \tag{12}$$

where D_H is given by Eq. (6), S_ξ by Eq. (8), U_n^+ by Eq. (1) and $\hat{T}_n$ by Eq. (9).

Figure 8 depicts a sample path generated by a Monte Carlo simulation of the TES model $\{X_n\}$ as defined by Eq. (12). Notice the similarity of the simulated sample path in Figure 8 to its empirical counterpart in Figure 6. A comparison of the corresponding autocorrelation functions appears in Figure 7, revealing a remarkable match at least up to

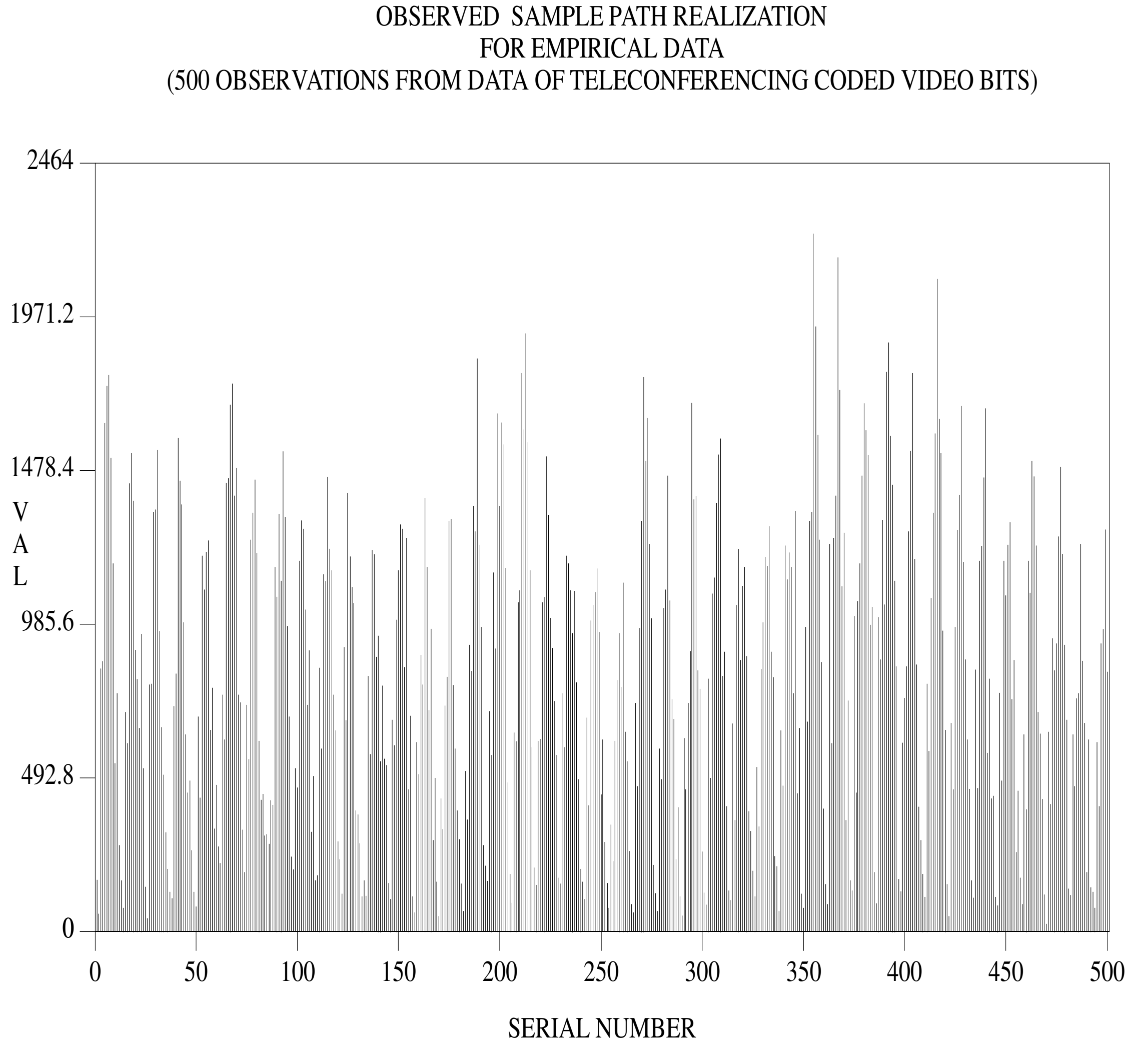

Figure 6: A Teleconferencing Empirical Sample Path of Bitrate Data Generated by GOB Compressed Video.

COMPOSITE(2) SIMULATED TES^{+}(α= 0.50, ϕ= 0.30, ξ= 0.50) AND OBSERVED AUTOCORRELATION FUNCTION
FOR EMPIRICAL DATA
(100 LAGS FROM DATA OF TELECONFERENCING CODED VIDEO BITS)

SIMULATED: bullets
OBSERVED: bars

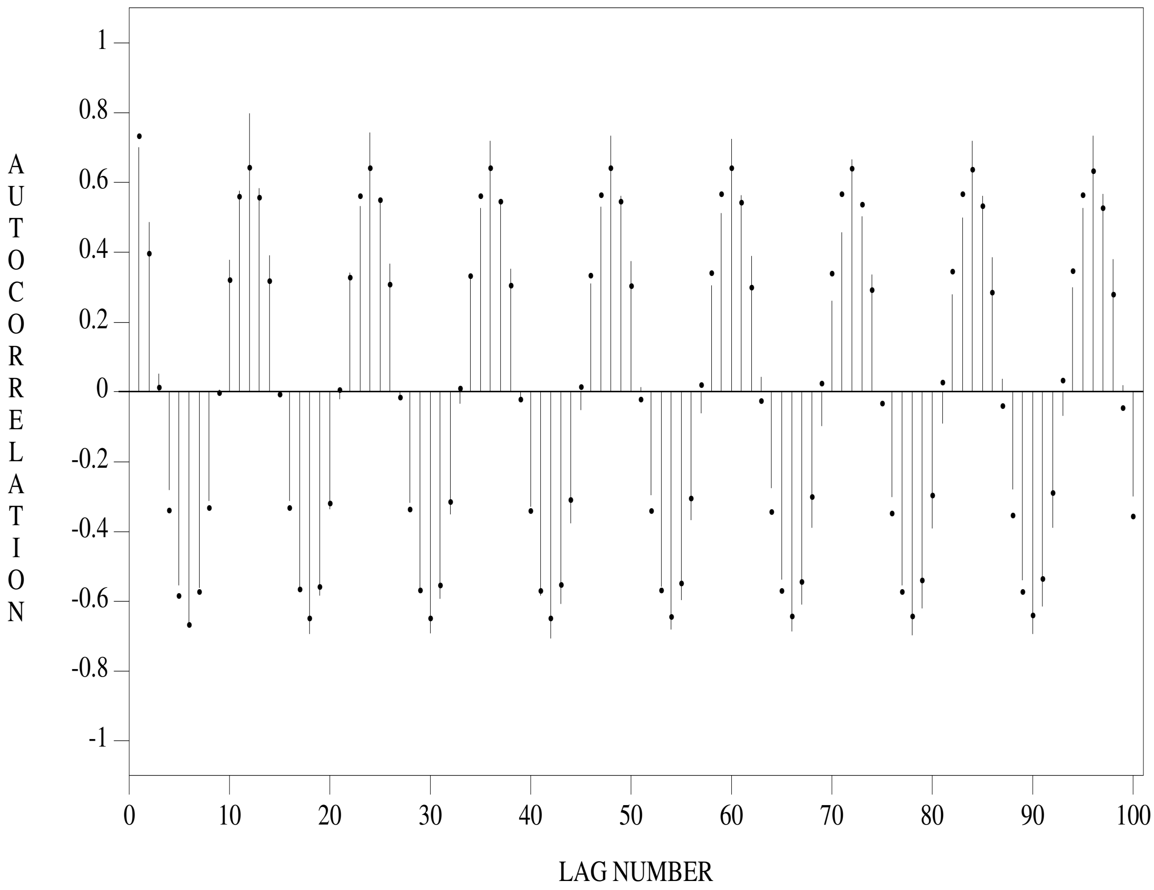

Figure 7: A Teleconferencing Autocorrelation Function of Bitrate Data Generated by a TES Model of GOB Compressed Video and Its Empirical Counterpart.

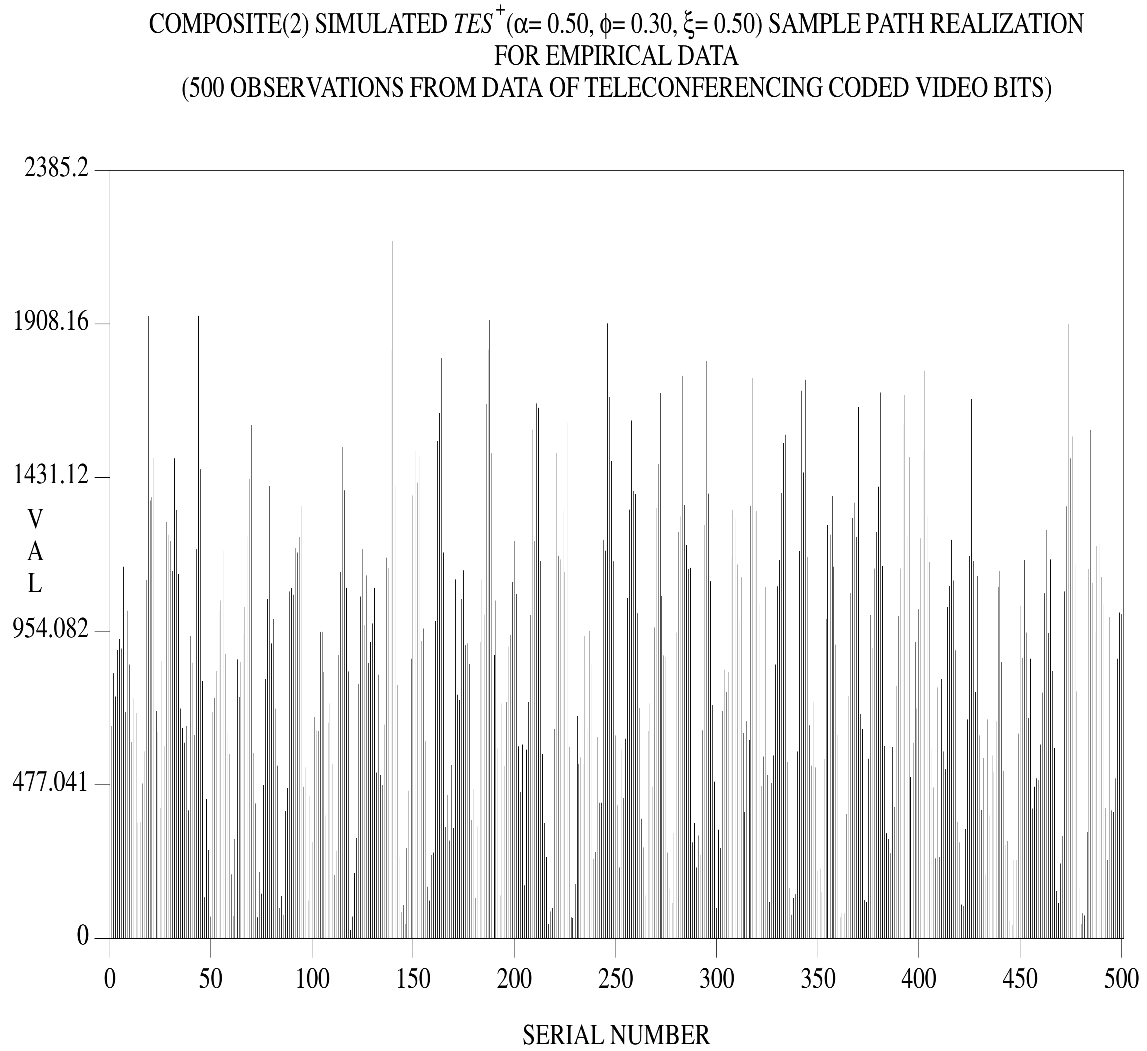

Figure 8: A Teleconferencing Simulated Sample Path of Bitrate Data Generated by a TES Model of GOB Compressed Video.

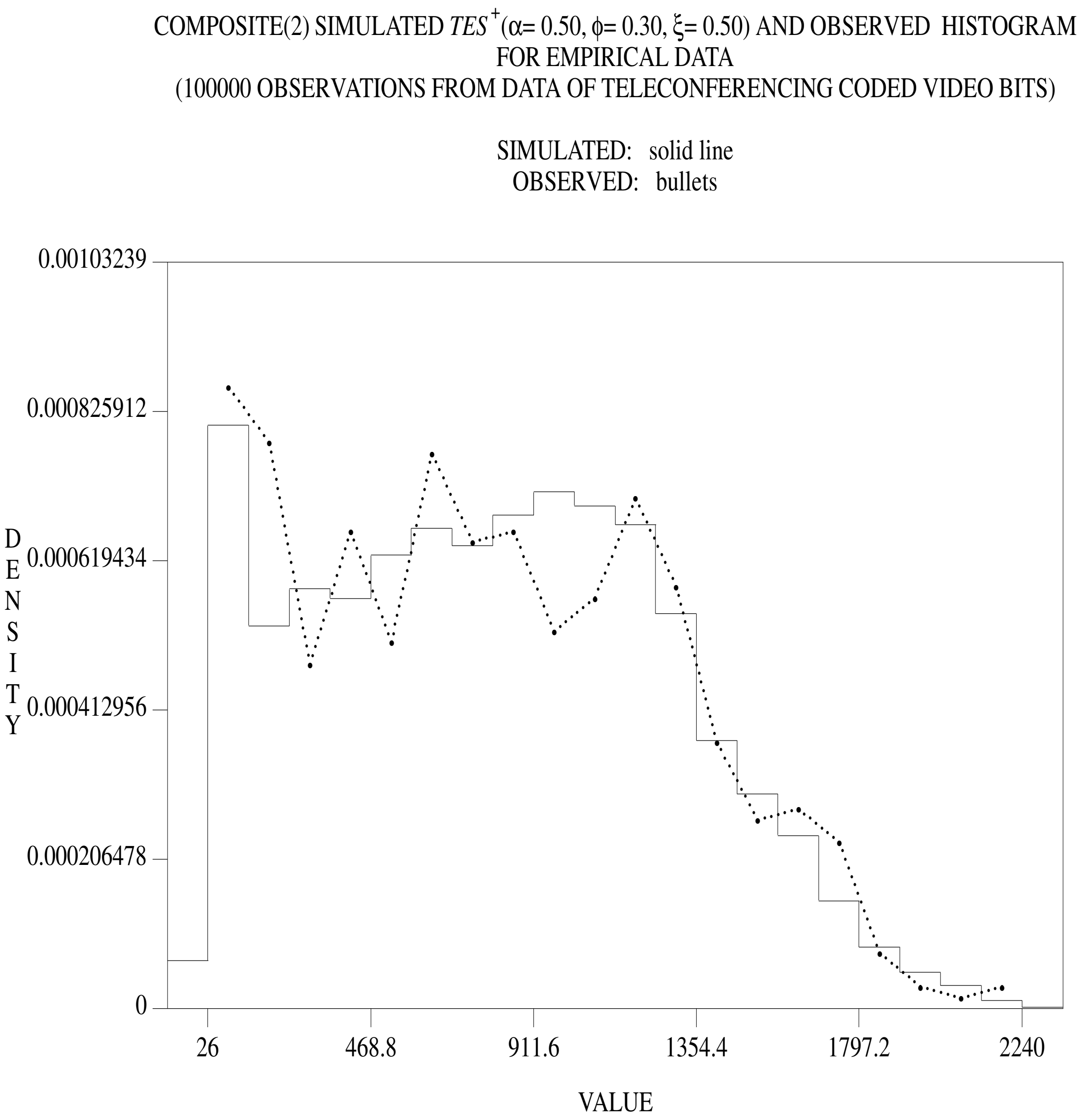

Figure 9: A Teleconferencing Histogram of Bitrate Data Generated by a TES Model of GOB Compressed Video and Its Empirical Counterpart.

lag 100. Finally, Figure 9 compares the histograms of $\{\hat{X}_n\}$ and $\{X_n\}$, again with good agreement. We remark that the TES variant used here does not guarantee a match to the empirical histogram of the raw data; rather, it gurantees a match to the empirical histogram of the residuals (11). Nevertheless, our experience shows that a reasonable match to the empirical raw data histogram is usually attained.

5 CONCLUSION

Offering autocorrelated traffic to queueing systems can result in far worse performance measures, as compared to those predicted by classical queueing models which ignore autocorrelation. Consequently, we need a new modeling approach which can capture autocorrelations of such traffic, and which can simultaneously model arbitrary marginal distributions (histograms) of empirical time series. VBR compressed Video traffic in high-speed communications networks is a case in point, since it exhibits a highly autocorrelated behavior and a variety of marginal distributions, depending on the video content and coding algorithm used.

This paper described TES processes and suggested the TES modeling methodlogy as a versatile approach to source modeling of autocorrelated traffic, of which the random bitrate of VBR coded video is a major representative. The paper outlined this methodology and how it is supported by a visual interactive modeling environment, called TEStool, that supports a heuristic search approach which can simultaneously capture the marginal distribution and autocorrelation function of empirical traffic data. The user interface of TEStool casts the search into an intuitive activity of modifying a visual parametric representation in small incremental stages.

The efficacy of our approach was demonstrated by two examples, where TES models were constructed from empirical data, obtained from measurements of the output bitrate of video codecs. In particular, we demonstrated the versatility of the TES approach in modeling both frame-level and group-of-block bitrate streams, resulting in monotone and cyclical autocorrelation structure, respectively. We found that using the TEStool modeling environment yielded remarkably accurate TES models, speeded up the modeling process and alleviated the tedium of repetitive search.

References

[1] P. Bratley, B.L. Fox and L.E. Schrage. *A Guide to Simulation*. Springer-Verlag, New York, NY, 1987.

[2] K.W. Fendick, V.R. Saksena and W. Whitt. Dependence in Packet Queues. *IEEE Trans. on Comm.*, 37:1173-1183, 1989.

[3] D. Geist and B. Melamed, "TEStool: An Environment for Visual Interactive Modeling of Autocorrelated Traffic", *Proceedings of ICC '92*, Vol. 3, 1285-1289, 1992, Chicago, Illinois.

[4] U. Grenander and M. Rosenblatt, *Statistical Analysis of Stationary Time Series*, Wiley, 1957.

[5] H. Heffes and D.M. Lucantoni. A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance. *IEEE Journal on Selected Areas in Communications*, Vol. SAC-4, 856-868, 1986.

[6] D. Jagerman and B. Melamed. The Transition and Autocorrelation Structure of TES Processes Part I: General Theory. *Stochastic Models*, Vol. 8, No. 2, 193-219, 1992.

[7] D. Jagerman and B. Melamed. The Transition and Autocorrelation Structure of TES Processes Part II: Special Cases. *Stochastic Models*, Vol. 8, No. 3, 499-527, 1992.

[8] D. Jagerman and B. Melamed. Spectral Analysis of Basic TES Processes. *Preprint*, NEC USA, Inc., Princeton, New Jersey, 1992.

[9] P.A.W. Lewis and E. McKenzie. Minification Processes and Their Transformations. *J. Appl. Prob.*, Vol. 28, 45-57, 1991.

[10] M. Livny, B. Melamed and A.K. Tsiolis. The Impact of Autocorrelation on Queuing Systems. *Management Science*, 1993 (to appear).

[11] B. Melamed. TES: A Class of Methods for Generating Autocorrelated Uniform Variates. *ORSA J. on Computing*, Vol 3, 317-329, 1991.

[12] B. Melamed, D. Raychaudhuri, B. Sengupta and J. Zdepski. TES-Based Traffic Modeling for Performance Evaluation of Integrated Networks. *Proceedings of INFOCOM '92*, Vol. 1, 75-84, Florence, Italy, 1992.

[13] B.E. Patuwo, R.L. Disney and D.C. McNickle. The Effects of Correlated Arrivals on Queues. *IIE Tranactions*, 1993 (to appear).

[14] B. W. Schmeiser. Simulation Experiments. In *Handbook of Operations Research and Management Science: Vol. 2* (D. Heyman and M. Sobel, editors). North Holland, New York, 1990.

[15] D.-S. Lee, B. Melamed, A. Reibman and B. Sengupta. Analysis of a Video Multiplexor using TES as a Modeling Methodology. *Performance Evaluation*, to appear.

[16] Plompen, et. al., "Motion Video Coding in CCITT SG XV - The Video Source Coding," *Proceedings of GLOBECOM '88*, 31.2.1-31.2.8, Hollywood, Florida, 1988.

[17] A. R. Reibman, "DCT-based Embedded Coding for Packet Video,", *Signal Processing: Image Communication*, Vol. 3, 231-237, 1991.