

MPEGTool: An X Window Based MPEG Encoder and Statistics Tool¹

Toshiyuki Urabe Hassan Afzal Grace Ho
Pramod Pancha Magda El Zarki

Department of Electrical Engineering
University of Pennsylvania
Philadelphia, PA 19104

Abstract

In this paper, we describe MPEGTool, an X window based tool which can be used to generate an MPEG² encoded bit stream for video sequences and to study the statistical properties of the encoded data. It is a very versatile tool that was designed to study the characteristics of variable bit rate video sources for transmission over ATM³ based BISDN⁴. The tool, which has a window based graphical user interface, allows a user to specify several of the MPEG parameters such as the intraframe to interframe ratio, and the quantizer scale. The tool also includes a statistical package which allows the user to plot graphs of various statistics including bit distributions, ATM cell distributions, time series, autocorrelation functions and cell interarrival times.

1.0 Introduction

The demand for increased bandwidth to support new services which integrate diverse media such as video, voice, graphics and text has lead to the introduction of BISDN. ATM, a packet switched approach based on fixed cell sizes, has been proposed as the switching and multiplexing scheme that can provide a unified transport structure for services in BISDN. Among the variety of services that will be supported by BISDN, video will be an important component because of its large bandwidth utilization and its stringent quality of service requirements.

ATM based networks are especially well suited for variable bit rate (VBR) video transport because of their ability to allocate bandwidth on demand to these services. VBR video is potentially superior to constant bit rate (CBR) video since it can provide constant image quality for all scenes, as well as efficient bandwidth utilization. However, many aspects of the nature of the relation between

VBR video services and network performance are still open issues[5].

One potential problem in ATM networks, caused by the bursty nature of traffic and statistical multiplexing, is cell loss. When several sources transmit at their peak rates simultaneously, the buffers available at some switches may be inadequate causing overflow. The congestion at these switch buffers and the subsequent dropping of cells due to overflow will most likely be the major component of cell loss in BISDN. For VBR video sources this can lead to severe degradation in service quality, especially if cells are discarded indiscriminately at the switch buffers. Priority mechanisms are important in these situations as they can be used to flag cells which are essential for some minimum quality of service to be maintained.

The traffic characteristics of a video source are determined both by the properties of the video sequence, i.e. dimensions, motion content etc. and the source coding technique. With the widespread acceptance of the MPEG coding standard, it is important that we understand the basic characteristics of MPEG encoded video data and its behavior in ATM based BISDN. Since hardware based MPEG coders are not available right now (only a few vendors have started to ship sample versions of the MPEG chip set), the only viable method for performing MPEG coding is in software. The versatility and user friendly nature of the MPEGTool can aid the task of studying issues in video transmission in BISDN.

In this paper we describe the design of the MPEGTool that we are currently using for our studies of MPEG video. It consists of an MPEG encoder that encodes digital video data, and a statistics package to study the characteristics of the encoded bit stream. Our MPEGTool is based on the current MPEG standard. This will be upgraded to MPEG II once it has been released. The tool consists of the following:

1. An X-window based graphical user interface
2. The MPEG encoder which includes:
 - a. Intraframe (I), Predictive (P) and Interpolative (B) frame coding
 - b. Layering scheme which separates the encoded bit stream into a high priority (HP) and a low priority (LP) stream
3. A GNUPLOT based graphical statistics package.

The outline of this paper is as follows: in section 2 we present a brief overview of the MPEG algorithm, in section 3, we describe the MPEGTool, and in section 4, we display some of the statistical results generated by the tool.

1. This work has been supported in part by grant NCR 90-16165 from the National Science Foundation.

2. Motion Picture Expert Group

3. Asynchronous Transfer Mode

4. Broadband Integrated Services Digital Network

2.0 MPEG Coding

The MPEG coding algorithm was developed primarily for storage of compressed video on digital storage media [1]. Provisions were therefore made in the algorithm to enable random access, fast forward/reverse searches and other features when decoding from any digital storage media. However, the coding standard is suitable for a much wider range of video applications. Recent applications of MPEG-like coding algorithms have appeared for a variety of video services from multimedia workstations to high definition television.

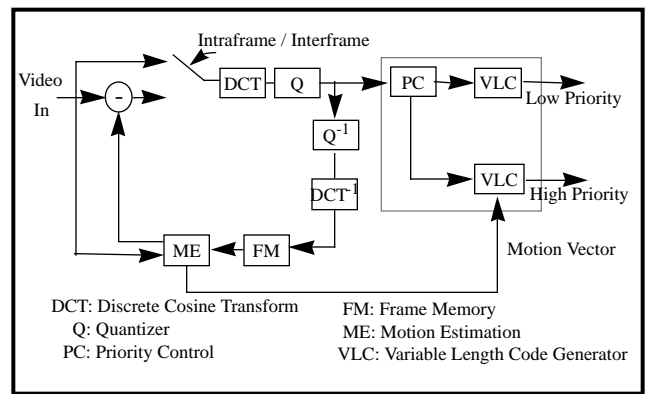
The MPEG coding scheme utilizes one of 3 coding modes for a frame in a video sequence; intraframe (I), predictive (P) or interpolative (B). Within a frame it is also possible to encode macroblocks (16x16 pixel blocks made up of 4 8x8 luminance pixel blocks and 2 8x8 chrominance pixel blocks) in one of several modes. A horizontal strip of 32 macroblocks which makes up a row in the frame is called a slice. In the MPEG coding algorithm the slice is an important since it is the smallest unit which can be reconstructed independently at the receiver.

The macroblock coding modes that can be used are dependent on the type of frame coding mode (I, B, or P) that is used for the current frame. In I frames, all macroblocks are coded in intraframe mode using 8x8 two dimensional Discrete Cosine Transform (DCT), and are then quantized and variable length coded. In P frames, macroblocks can be coded with or without motion compensation. If motion compensated mode is utilized, a motion vector is obtained by minimizing the absolute block difference between the current macroblock and a macroblock within the search window in the previous frame. The prediction error after utilizing motion compensation is coded using the DCT and quantized. Both the motion vector and the quantized DCT coefficients are then variable length coded. If the prediction error is too large, the macroblock is coded in the intraframe mode instead. For B frames, the coding procedure is similar to that for P frames. The major difference in B frames is that both forward and backward motion vectors are allowed unlike P frames which only utilize forward motion vectors. The use of backward motion vectors in B frames has some disadvantages including larger buffers at the source and receiver and longer processing times and therefore may not be suitable for all applications.

We have enhanced the operation of the coder by adding a layering scheme, which separates the bits generated by the encoder into HP or LP bit streams. A parameter β specifies the number of AC coefficients (frequency components) in the HP stream ($\beta=64$ puts all the coefficients into HP, i.e. the resultant bit stream is the same as the standard MPEG bit stream). More details on the layered coder can be obtained in [2]. In Figure 1, we show a block diagram of the MPEG coder which includes the layering mechanism.

Further details of the MPEG coding algorithm can be obtained in [1] and details on the application of the MPEG algorithm to variable bit rate video can be found in [3].

Figure 1. Diagram of MPEG encoder

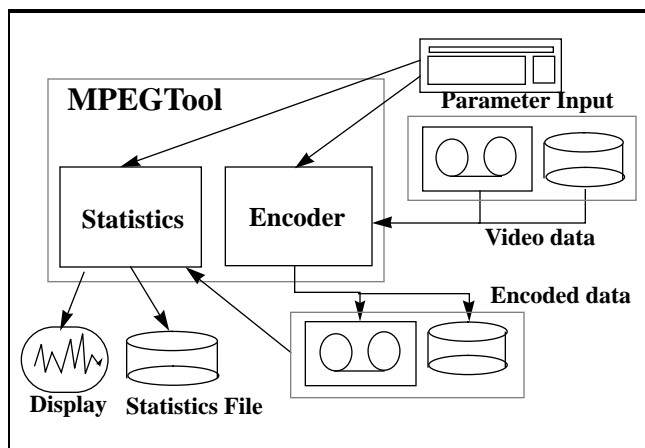


3.0 MPEGTool

Our tool consists of two components, an MPEG encoder and a statistics package. Since this tool has an X window based graphical user interface, it is user friendly and easy to operate.

Figure 2 shows the data flow diagram of our tool. The encoder reads raw digital video data from a tape device, performs the encoding and generates an encoded bit stream. Since the encoding process is performed in software and not hardware, it cannot be done in real time. For this reason, all statistics are processed and displayed once encoding is complete.

Figure 2. Flow Control diagram of MPEGTool



The tool is versatile since it allows the user to specify several parameters for the encoding process. The first set of parameters deals with the intraframe to interframe coding ratio. A sequence of I, P and B frames is defined by the two parameters M and N . N specifies the I frame interval whereas M specifies the I or P frame interval. N must be an integer multiple of M and when N is 1, the sequence contains only I frames. If M is 1 and N is greater than 1, the sequence contains I and P frames but no B frames. If N equals M and both N and M are greater than one, the sequence contains I and B frames but no P frames. Some examples of (N, M) combinations and the resultant frame coding sequences are shown below.

1. $N = 1, M = 1$: I I I I I I I I I I I I I I (I frames only)
2. $N = 4, M = 1$: I P P P I P P P I P P P I (I and P frames)
3. $N = 6, M = 2$: I B P B P B I B P B P B I (I, P and B frames)
4. $N = 6, M = 3$: I B B P B B I B B P B B I (I, P and B frames)
5. $N = 6, M = 6$: I B B B B B I B B B B B I (I and B frames)

In the combinations which utilize interpolative coding it should be pointed out that the order of transmission (or file storage) of frames is not the same as that presented above. This occurs because in order to reconstruct a B frame information from both a past and a future frame is required. The next P or I frame following the B frame must therefore be transmitted before the B frame can be decoded. The sequence of frame transmission for the cases depicted above is:

1. $N = 1, M = 1$: I I I I I I I I I I I I
2. $N = 4, M = 1$: I P P P I P P P I P P P I
3. $N = 6, M = 2$: I P B P B I B P B P B I B
4. $N = 6, M = 3$: I P B B I B B P B B I B B
5. $N = 6, M = 6$: I I B B B B B I B B B B B

The two other important parameters are: the quantizer scale, q , which is specified in the MPEG coding standard and the layering parameter for the prioritized coder, β , which specifies the number of AC coefficients to be placed in the HP and LP streams. Together, these four parameters characterize the MPEG coding scheme.

3.1 The Encoder

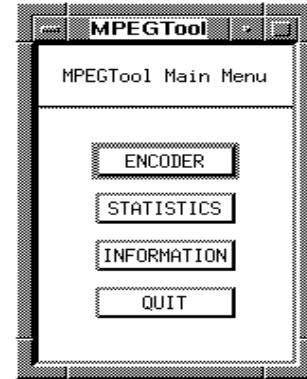
The input image sequences that are currently supported by MPEG-Tool are CCIR 601 format (720 x 480 pixels, YUV 4:2:2: format) and RGB format sequences. Since no standard file format definitions exist currently for RGB sequences, we have defined a simple file header structure similar to that of *pbm* format image files which contains the information that is required to encode the image sequence. In our format, the first line contains a string of the image format type (i.e. CCIR or RGB). For RGB format files, the next line contains the name of the sequence and the subsequent two lines contain the width followed by the height of the frames in the video sequence. Following the header, the image information is stored in separate red, green and blue component blocks. Each component block is composed of (*height x width*) 8 bit values and each frame in a sequence contains blocks arranged in R, G and B order. For CCIR format files, no extra header information is required since the physical properties of the image are defined in the standard. The video sequences can be read either from disk file or from a tape device.

The encoder reads the digital video data, encodes it using the N , M , β and q parameters and generates a bit stream for the number of bits produced per macroblock. Figure 3 shows the main menu window for the tool. The "INFORMATION" button will bring up a help message for the user which describes the tool. This includes the hardware requirements for the tool, easy to follow instructions on how to use the coder and an anonymous ftp location for the source code.

Clicking "ENCODER" at the main menu pops up the submenu shown in Figure 4 in which the MPEG parameters are set, source data file is chosen and the frames to be encoded are identified. After all the inputs have been specified, clicking "Start" checks all the inputs for errors. The following evaluation rules apply for the parameters:

1. $N > 0, M > 0$
2. N must be an integer multiple of M
3. $0 < \beta \leq 64$
4. $0 < q \leq 32$
5. Number of frames to be encoded must be an integer multiple of M plus 1.

Figure 3. MPEGTool main menu



If these rules are not satisfied, an error message dialog box appears and control is returned to the parameter setting menu. If the parameters are correct, MPEG encoding starts.

The encoding process generates two types of data files. A bit stream with the real MPEG coded video data is created and stored in a file. In addition, a file containing information about both the bits generated in each macroblock and the macroblock coding type such as I, P or B. After encoding is complete, control returns to the main menu.

Figure 4. MPEG encoder window

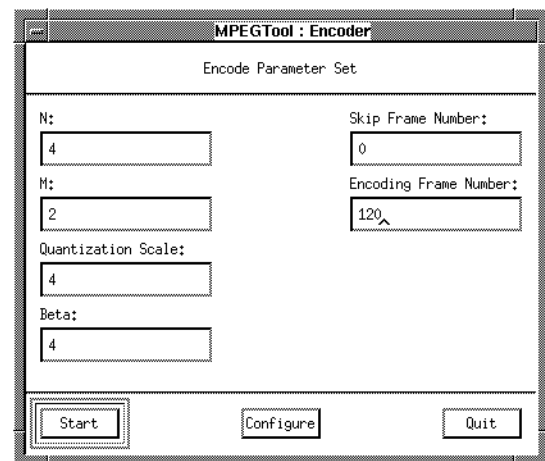
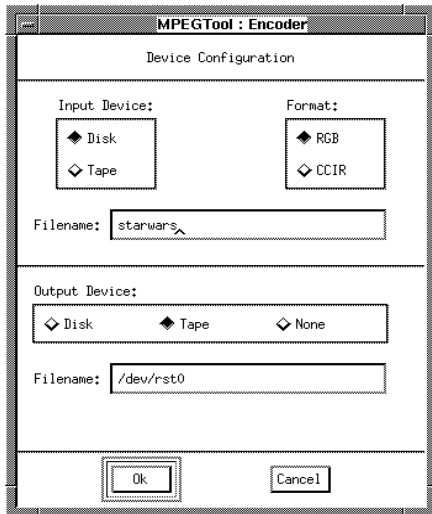


Figure 5. I/O devices configuration window

The encoded bit stream produced by the MPEGTool is fully compliant with the MPEG standard. The video sequence can be played back utilizing any MPEG decoder. The encoded data file has been tested with several public domain decoders and we have found no compatibility problems with any of them. It should be noted that for the layered MPEG coder only the HP bit stream is fully compliant with the MPEG syntax. Therefore only this part of the bit stream can be decoded by a standard MPEG decoder. However, it is possible to decode the HP and LP bit stream together if some simple changes are made in the decoders. The ability to view the decoded sequence is extremely important since visual quality is the primary measure of system performance of interest to a user.

3.2 Statistics

The statistical properties of video encoded bit streams are as yet not well understood. Knowledge of their properties is essential for determining the best strategies for allocating resources in BISDN



networks. It is essential that we understand the statistical properties of packet video streams if we are to design B-ISDN networks that handle heterogeneous traffic. Therefore an important component of our tool is the statistics package.

Once encoding is complete, the resulting video traffic can be studied with the statistics menu. Clicking "STATISTICS" at the main menu displays a file selection submenu (Figure 4) from which MPEG encoded video data files can be selected. When a file has been chosen, the statistics definition window (Figure 5) appears from which the statistical properties which need to be analyzed can be chosen. Each choice analyzes the encoded video data at either the bit or the ATM cell level and is calculated on a frame, slice or macroblock basis. The following four statistical properties can be analyzed with the MPEGTool;

Figure 6. File Selection Menu

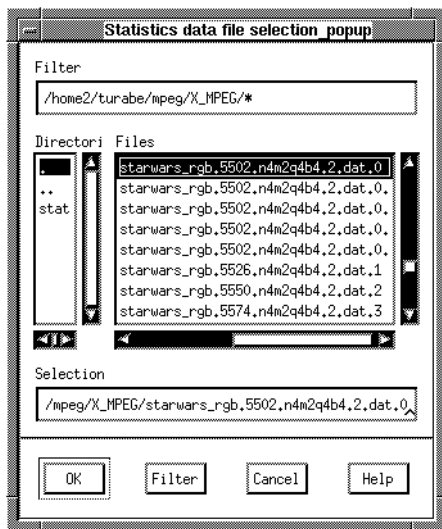
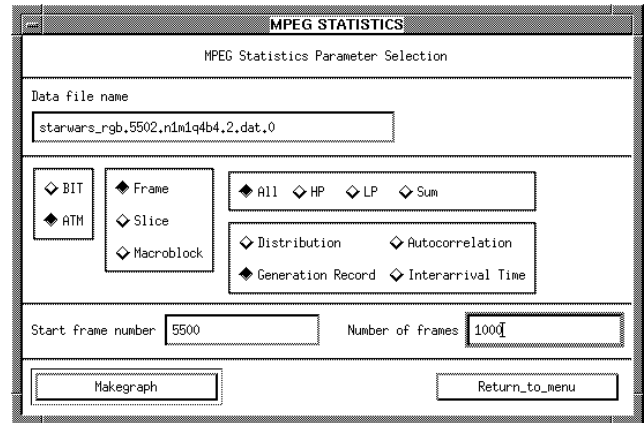


Figure 7. Statistics Definition Window



- **Distribution** - This option plots the distribution of bits or ATM cells per frame or slice. The distribution of cells per frame is an important statistic for video transmission in networks. The distribution function for a source can be used to determine the number of identical, independent sources that can be accommodated on a given network link for a given cell loss rate.
- **Generation Record** - This option plots the generation of the encoded data stream in time. The generation of either bits or ATM cells can be studied per frame, slice, or macroblock. The generation of bits per frame is simple and is calculated directly by summing the bits of the 960 macroblocks which constitute the frame. The bit generation per slice is calculated similarly by summing the bits of the 32 macroblocks that make up a slice. The total cell arrivals per frame is obtained by first determining the number of bits per frame and then converting this to the equivalent number of ATM cells with a payload of 44 bytes. Similarly, the total number of cell arrivals per slice is calculated by first determining the bits generated per slice and then converting this to the equivalent number of ATM cells. These records can be stored in files and can be used in trace driven simulations to determine performance of real video sources.
- **Autocorrelation** - This option plots the autocorrelation function of the number of bits or ATM cells generated for both frames and slices. The normalized sample autocorrelation function, $R(n)$, of a function x is calculated using,

$$R(n) = \frac{1}{R(0)} \left\{ \frac{1}{N} \sum_{k=0}^N (x_k - \mu_x)(x_{n+k} - \mu_x) \right\}$$
 where μ_x is mean of x . This function provides a measure of the correlation between any two points of the function x with a separation of n time units. This measure is useful in characterizing the video source for modelling purposes. The degree of correlation can also have a significant impact on the performance of video sources in networks.
- **Interarrival Time** - This option plots the time elapsed in between arrivals of ATM packets within a frame. The interarrival times are calculated from the bits generated per macroblock within a frame. This interarrival time is normalized to units of X seconds (where X will depend on the hardware implementation of the coder). We assume a simple packetization procedure for the source. Cells are formed when the sum

of bits generated by successive macroblocks is greater than or equal to the size of an ATM cell. It is assumed that bits generated for a frame will not be combined with the bits generated from the next frame.

For each statistic, the user has the option of specifying a specific range of data (start frame number and number of frames) over which the analysis is to be done. When no range specification is made, the tool automatically selects a range which displays all the encoded data. The displayed graph can be sent to a laser printer or be stored as postscript files.

4.0 Results

In this section we present some of the statistics generated by the MPEGTool. The video sequence used to test the coder was a 2 minute segment from the movie "Star Wars". The sequence was digitized into RGB images from a laserdisc which has a resolution close to NTSC broadcast quality. The spatial dimensions of the digitized frames is 512x480 pixels. The sequence was encoded utilizing a variety of N , M , q and β parameters and sample statistics were collected.

Figure 8 and Figure 9 show the generation of bits per frame for two different encoding sequences. The first sequence is IBPBIBPB... ($N=4$, $M=2$) and uses the interpolative encoding mode while the second sequence is IPPPIPPP... ($N=4$, $M=1$) and uses the predictive encoding mode. In each of these sequences we utilized a value of $q=4$ and $\beta=4$. The X axis and Y axis show respectively the frame number and the number of bits generated in each frame. The graph shows the number of bits generated in the HP and LP bit streams.

From Figure 8, we can observe that interpolative coded frames achieve the highest compression ratios. However, with $N=4$ we notice that the mean bit rate for the first and second sequences are not very different. Since the bit rate is predominantly composed of I frames for this case, there is little advantage in utilizing interpolative coding to increase the compression ratio.

Figure 10 shows the generation of ATM cells per slice for $N=4$, $M=2$, $q=4$ and $\beta=4$. This graph shows the sum of ATM cells in the HP and LP streams. The peaks in the cells per slice graph occur at intervals of approximately 30 slices. These regular peaks are a result of the high temporal correlation between slices in adjacent frames. These correlated bursts that occur at the slice level can cause losses at the cell level in network switching elements. This is the reason why it is important to perform some smoothing within a frame at the encoder before cells are transmitted to the network.

Figure 11 and Figure 12 show the distribution of ATM cells per slice for $\beta=4$ and $\beta=16$ respectively and with $N=4$, $M=2$ and $q=4$. Since a larger β value puts more bits into the HP stream, the number of ATM cells shifts to the right for the HP stream and shifts to the left for the LP stream.

In packet switched networks where congestion occurs in the network, β can be utilized to ensure that the source and the receiver do not lose synchronization. Congestion feedback information from the network can be utilized by the coder to ensure that HP cells are not lost. This layered approach also results in the best visual quality since the most important information, DC coefficients and motion vectors, will never be lost.

Figure 13 and Figure 14 show the interarrival time of ATM cells for $M=4$, $N=2$, $\beta=4$ and two different quantization scales $q=4$ and $q=16$. The generation interval of ATM cells becomes longer for larger values of q as the encoder generates fewer bits (lower image resolution). For coders which do not utilize any output buffering, the interarrival times can be used in modeling the arrival process to the network at the cell level.

The autocorrelation function for a 600 frame sequence of an $N=1$, $M=1$ coder is shown in Figure 15. The autocorrelation function can be used both qualitatively and quantitatively to describe the bit and cell generation process for a video coder. The autocorrelation function for this coder monotonically decreases except for a peak at 60 frames. In general, the correlation stays fairly high for frame lags as large as 100 frames. The autocorrelation function is also utilized to determine the coefficients for autoregressive models for video sources. These models are useful to characterize video sources for network simulation.

5.0 Summary

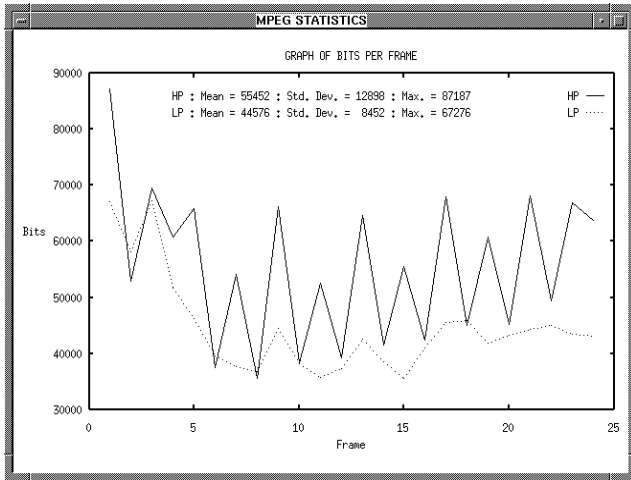
We have described an X windows based MPEG encoder and statistical analysis tool, with which the characteristics of VBR MPEG video data can be easily studied. The tool consists of two basic components: an MPEG encoder and a statistical package. The tool can read input video sequences in 2 different formats, CCIR and RGB, from either a file on disk or a tape device. Both the actual bit stream for the video sequence as well as statistics for the encoded video sequence can be obtained from the tool. The bit stream can be played back by any MPEG compliant decoder (many of which are available as public domain tools also). We have presented an assortment of results which can be obtained from encoding a video sequence and examining the associated statistics.

The tool is available for public use and information on the tool can be obtained by sending e-mail to mpegttool@ee.upenn.edu.

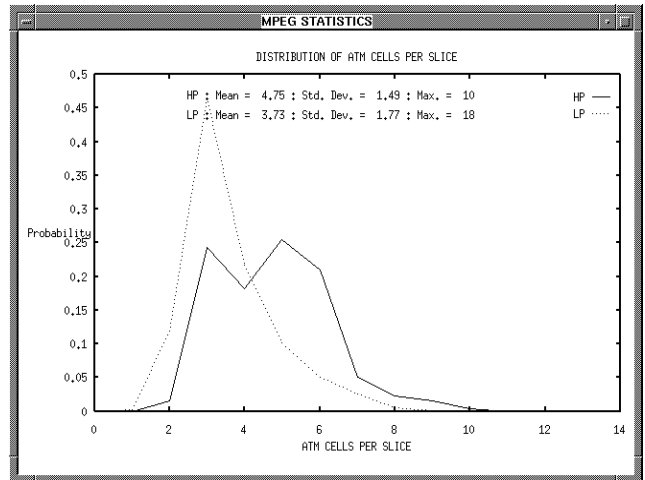
References

- [1] D. Le Gall. MPEG: A video compression standard for multimedia applications Communications of the ACM, 34(4):305-313, April 1991.
- [2] P. Pancha and M. El Zarki. Prioritized Transmission of Variable Bit Rate MPEG Video, Proceedings of GLOBECOM '92, December 1992.
- [3] P. Pancha and M. El Zarki. A look at the MPEG video coding standard for variable bit rate video transmission, Proceedings of INFOCOM '92, May 1992.
- [4] Dan Heller. Motif Programming Manual, O'Reilly & Associates, Inc., 1991
- [5] Y-Q Zhang, W. W. Wu, K.S. Kim, R.L. Pickholtz, and J. Ramasastry. Variable Bit Rate Video Transmission in the Broadband ISDN Environment. Proceedings of the IEEE, 79(2):214-221, February 1991.

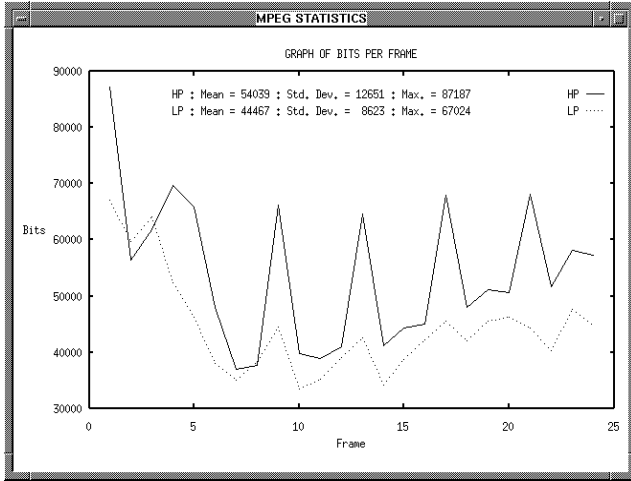
e 8. Bit generation per frame (N=4, M=2, q=4, β=4)



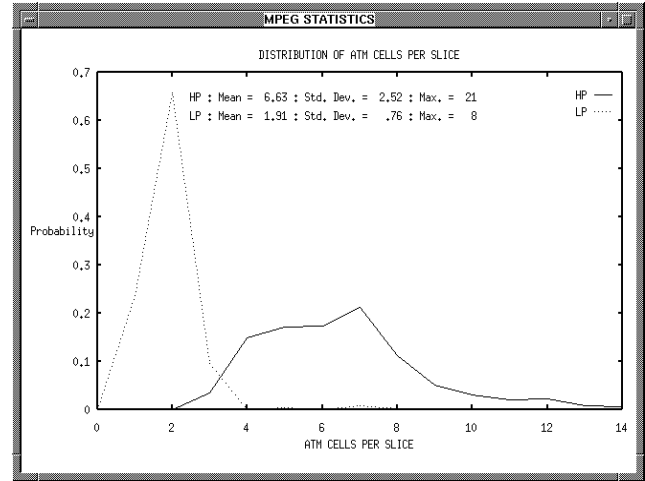
e 11. ATM cells/slice distribution (N=4, M=2, q=4, β=4)



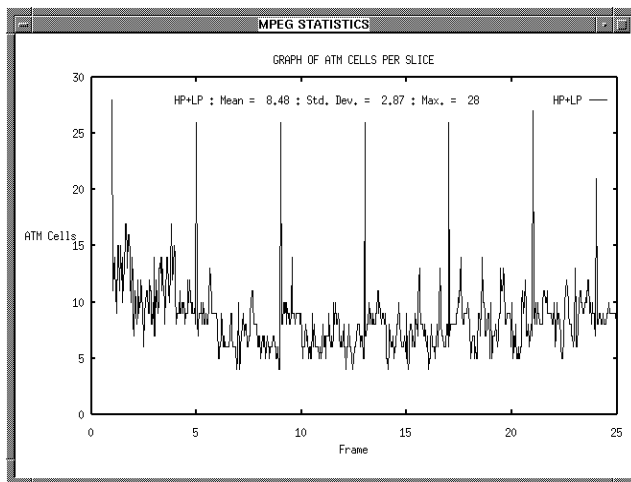
e 9. Bit generation per frame (N=4, M=1, q=4, β=4)



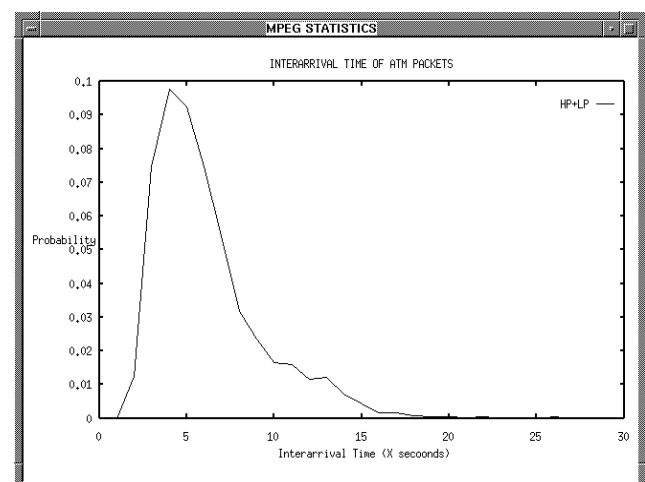
e 12. ATM cells/slice distribution (N=4, M=2, q=4, β=16)



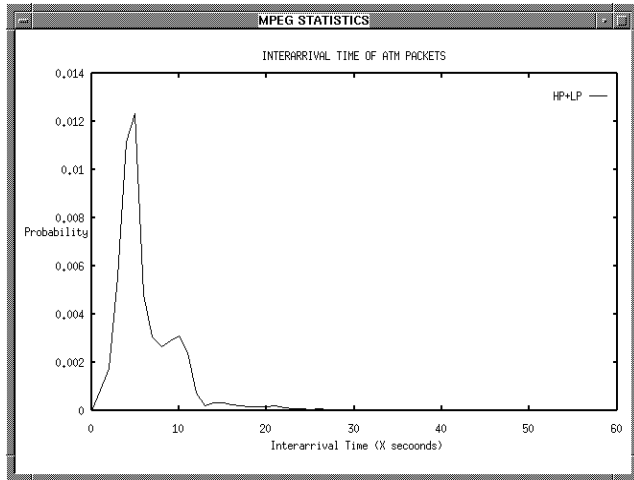
e 10. ATM cell generation per slice (N=4, M=2, q=4, β=4)



e 13. Interarrival time of ATM cells (N=4, M=2, q=4, β=4)



e 14. Interarrival time of ATM cells ($N=4, M=2, q=16, \beta=4$)



e 15. Autocorrelation function ($N=1, M=1, q=4, \beta=16$)

