Fuzzy Based Rate Control for Real-Time MPEG Video

Danny H.K. Tsang,*Member, IEEE*, Brahim Bensaou, *Member, IEEE*
and Shirley T.C. Lam ,*Member, IEEE*

Abstract— **In this paper, we propose a fuzzy logic based control scheme for real time MPEG video to avoid long delay or excessive loss at the user-network interface (UNI) in an ATM network. The system consists of a shaper whose role is to smooth the MPEG output traffic to reduce the burstiness of the video stream. The input and output rates of the shaper buffer are controlled by two fuzzy logic based controllers. To avoid a long delay at the shaper, the first controller aims to tune the output rate of the shaper based on the number of available transmission credits at the UNI and the occupancy of the shaper's buffer in the video frame time-scale. Based on the average occupancy of the shaper's buffer and its variance, the second controller tunes the input rate to the shaper over a much larger time-scale by applying a closed loop MPEG encoding scheme. With this approach, the traffic enters the network at an almost constant bit rate (with a very small variation) allowing simple network management functions such as admission control and bandwidth allocation, while guaranteeing a relatively constant video quality since the encoding rate is changed only in critical periods when the shaper buffer "threatens" to overflow. The performance of the proposed scheme is evaluated through numerical tests on real video sequences.**

Keywords— **Traffic shaping, Fuzzy control, Asynchronous transfer mode, MPEG real time video.**

I. Introduction

Video applications are expected to be the major source of traffic in future broadband networks. Such applications include interactive video-phone conversations and video-conferencing, retrieval of pre-recorded video sequences in multimedia databases, remote viewing of live events (sport, news) and simply watching a movie from a video on demand server. To support the large amount of traffic as well as the high transmission bit rates required by these applications, asynchronous transfer mode (ATM) technology has been adopted as the switching technique for future broadband networks. ATM provides a suitable and economical way for the transmission of information at a variable bit rate (VBR) in general and of VBR encoded video in particular. Encoding video signals at VBR is particularly appealing from the service provider's point of view because a constant video quality can be provided by the encoder. However, from the network provider's point of view, and paradoxically because of the VBR encoding, the output of the encoder becomes unpredictable which can cause congestion in the network leading to data loss and thus image quality degradation. To avoid this problem and to ensure that the quality of service (QoS) can be maintained to the customers of the network, it is necessary to implement appropriate traffic control and enforcement schemes at the edge of the network.

Dr Tsang is with the Department of Electrical and Electronic Engineering, Hong Kong University of Science and Technology eetsang@ee.ust.hk

Dr Bensaou is with the Centre for Wireless Communications, National University of Singapore. E-mail: cwcbb@leonis.nus.edu.sg

Ms Lam is with Motorola Asia/Pacific Ltd., Hong Kong

To guarantee the QoS in the broadband network, non-adaptive traffic control algorithms are preferred by many network providers because this type of algorithms can be easily maintained by the network providers and is economically implemented by switch vendors. Upon connection setup, a contract is established between the traffic source and the network. The traffic source must declare a set of traffic parameters that characterize the data it generates. These traffic parameters should be useful in determining the required amount of network resources to guarantee the QoS and must also be easy to control. The only traffic parameters currently adopted in ATM standards [1] are the peak cell rate, the sustainable cell rate and the burst tolerance. Based on the values of these parameters, the network determines whether the connection can be accepted without infringing either its QoS constraints or the QoS requirements of the connections in progress. To ensure that the user respects the contract and to avoid the QoS degradation caused by misbehaving sources, a usage parameter controller (e.g., leaky bucket) must control the user traffic at the user-network interface (UNI). A leaky bucket LB(r,b), where r is the leak rate (i.e., token generation rate) and b is the token pool size, can be considered as a counter which is increased at rate r cells/s up to a maximum of b cells and is decreased by one when a cell enters the network. Cells are delayed or dropped when the counter would otherwise decrease below zero. This cell drop/delay will degrade the overall QoS of the connection.

To be able to guarantee the QoS for variable bit rate (VBR) video connections, there are two options: either find a set of suitable traffic parameters to characterize the output traffic of the video coder so that a suitable and accurate model can be used to predict the QoS, or force the coder to make the traffic stream it generates conform to some pre-defined characteristics.

The characterization of video coder output for different applications (e.g., video-conferencing), coding algorithms and video sequences have been the subject of many performance studies (e.g., see [2] to [7]). These studies show that there are only a limited number of applications (e.g., video-phone, video-conferencing) whose traffic can be characterized in terms of a small number of parameters (e.g., the first two moments of the bit rate and the coefficient of autocorrelation [4]). For other types of video applications, to make the mathematical models analytically tractable, researchers often assume that the autocorrelation function of the coder's output rate is an exponentially decreasing function. With this latter assumption, classical Markovian methods are used to derive the performance of a network

model fed by the approximate traffic model (e.g., [8]). However, it has been shown recently [7], [2] that most long video sequences seem to systematically exhibit long-range dependence and thus show a non-exponentially decreasing autocorrelation function. Intuitively, a long-range dependence in a video sequence indicates that while the long-term correlations (large lags) remain individually small, their cumulative effect is significant. One direct consequence of this is, for instance, that buffering the traffic in the network for subsequent transmission does not help the congestion since the buffer occupancy probability for long-range dependent traffic seems to be heavily tailed. In other words, to prevent data loss, very large amounts of memory are needed which contradicts the video delivery delay constraints inherent to most video services. The scenarios produced are thus drastically different from those experienced with traditional short-range dependent models such as Markovian processes. These observations give rise to new and challenging problems in traffic engineering for high speed networks. Recent research results have shown that it is very difficult to analyze the performance of a network multiplexer with long-range dependent traffic streams. Besides, statistics on real data traces have shown that the Markovian assumption is obsolete, and that the parameter that characterizes the long-range correlations (the Hurst parameter) seems to be varying from one application to the other and even from one video sequence to the other [9]. This makes the simple characterization of the video coder output very difficult if not impossible.

The alternative approach of modifying the coder output so that it becomes more predictable has been considered by a few researchers. Reibman and Haskell [10] suggest constraining the bit rate to prevent the codec (COder-DECoder) buffer from overflowing in the case of a leaky bucket controlled channel. Heeke [11], and Coelho and Thome [12] aim to make the output of the video coder behave like a predefined Markov chain. Pickering and Arnold [13] propose a rate enforcement algorithm that produces a VBR traffic lying between given output rate upper and lower bounds. Hamdi and Roberts [14] also propose a rate control algorithm based on a closed loop MPEG encoding scheme to ensure that a VBR video traffic conforms to a given sustainable cell rate and burst tolerance, thereby eliminating cell dropping or delaying by the leaky bucket at the UNI.

⇒ [Fuzzy systems have been extensively used during the past few years to efficiently solve several traffic control problems in ATM networks. In [15], [16], [17] fuzzy logic has been used to design efficient traffic policing mechanisms in ATM networks. Problems such as flow control for ABR service category in ATM networks has been addressed in [18]. Routing problems both in ATM and non-ATM based networks have been addressed in many recent publications through neuro-fuzzy based approaches (e.g. [19]). Other problems such as cell loss probability estimation in ATM multiplexers as well as measurement based call admission control problems have been addressed in [20], [21]. For a more comprehensive list of references and/or general discussion of fuzzy logic control in ATM networks, the interested reader may refer to the excellent paper in [22]. The main advantage of fuzzy logic over Markovian probabilistic approaches resides in the simplicity and low computational complexity it requires to provide similar or even better approximate solutions to the real complex problems. In high speed networks, where the control decisions have to be made in extremely short times, fuzzy control have certainly a major role to play. On the other hand, due to its empirical nature, (i.e., depends extremely on human knowledge and experience), fuzzy control should not be systematically to solve traffic control problem. Fuzzy logic should only be applied when the traffic behavior is unpredictable, such as for real-time variable bitrate traffic, and/or the computational complexity required to solve the problem through classic teletraffic theory is very high (e.g. solving numerically a Markov chain of thousands of states). ⇒]

In this paper, we present a fuzzy-based algorithm that implements a self-policing function in a video coder to avoid cell dropping/delaying at the edge of the network by the leaky bucket. With this additional function in the coder, the video coder output traffic becomes easily characterizable. The basic idea of the fuzzy-based rate control scheme presented in this paper is to some extent similar to the rate control algorithm proposed by Hamdi and Roberts [14]. However, in their paper, Hamdi and Roberts assume implicitly that the output rate from the video coder is a slowly varying process. In other words, the rates in two successive groups of pictures are assumed to be roughly close. This assumption might be very different from the real behavior of an MPEG stream, especially at scene changes when the variation of the rate is very bursty. Cell loss at the UNI is, therefore, more likely to be unavoidable. In our proposed scheme we avoid cell loss by imposing a systematic shaping on the video output stream in a physical buffer, in order to make the video traffic always in conformance to the declared sustainable cell rate and burst tolerance. With this auxiliary self-policing function, no cells will be dropped at the edge of the network. On the other hand, to maintain a steady image quality, our algorithm tries to reduce the variance of the image quality while avoiding excessive access delay at the UNI. Like many schemes using the approach of modifying the coder output traffic, our scheme also eliminates the long-range dependence of the VBR video traffic.

The remainder of this paper is organized as follows. Section II describes the essential features of the MPEG standard and then introduces the motivations of our scheme. Section III presents the design of the fuzzy rate control scheme. The performance of our proposed scheme is studied in Section IV. We finally draw our conclusions in Section V.

II. MPEG Video Coding

In this section, we first discuss the essential features of the MPEG standard and the different MPEG encoding schemes. We also highlight the basic implications of these different standards on the real-time transmission of

broadcast-quality video using ATM networks. We then focus on the motivations of our proposed scheme.

A. MPEG (Motion Picture Expert Group) Picture Types and Sizes

MPEG is the ISO/IEC standard for digital video coding. It has been designed to satisfy a large variety of video applications. MPEG was first designed to overcome the problem of the storage of pre-recorded video on digital storage media, because of the compression ratios it achieves. However, the emergence of the ATM technique with the attractive features it provides has made the MPEG video compression standard appropriate also for the transmission of real-time digital video signals over high speed communication networks. The MPEG standard was developed to produce a bit stream which represents the raw video sequence by using fewer bits. A digital video sequence is a set of sequentially displayed frames. Each frame is made of a set of 8×8 arrays of pixels called blocks. The block is the smallest coding unit in MPEG. The blocks are grouped 4 by 4 to form macro-blocks, which represent the basic coding unit in MPEG. The blocks are transformed through the discrete cosine transform (DCT), and then digitized by quantization. The step size for the quantization of each DCT coefficient determines the compression ratio achieved by the MPEG algorithm. An MPEG compliant bit stream can be produced by applying both intra-frame and inter-frame coding techniques. Intra-frame coding is performed on a single frame of the video sequence to reduce the spatial redundancies, while the inter-frame coding exploits the temporal redundancies across successive frames.

In the MPEG standard, three coding modes at the frame level can be utilized; intra-frame (I), predictive (P), and bi-directionally interpolative (B). I-frames are coded with respect to the current frame (i.e. without reference to other frames) using a two-dimensional discrete cosine transform. I-frames have a relatively low compression ratio. P-frames are coded with reference to previous I or P frames using inter-frame coding; they achieve a better compression ratio than I-frames. B-frames are coded with reference to the next and previous I or P frame; they achieve the highest compression ratio of the three frame types. The actual compression ratios are specified by the quantization parameter Q at the coder [23], [24], [25]. For a typical MPEG sequence, an I-frame size is on average four times larger than a B-frame and two times larger than a P-frame because of the difference of the quantization parameters for each type of frame. However, a particular I-frame size may be an order of magnitude larger than a particular B-frame. The use of these three types of frames allows MPEG to be both robust (I frames permit error recovery) and efficient (B and P frames allow a good overall compression ratio).

In addition, the quantization parameter, the MPEG algorithm controls the characteristics of the output bit stream through the intra-frame to inter-frame ratio which defines the number of B and P frames between two successive I frames. The sequence of frames between two I frames (including the I-frame starting the sequence) is called a group of pictures (GoP). For instance, a typical GoP is a sequence of frames of the form IBBPBBPBBPBBP. The quantization parameter, the GoP length as well as the length of the periodic sequence of B and P frames in a GoP are user specified. Successive frames with large size differences produce a very "bursty" bit stream when they are transmitted on a frame-by-frame basis over a network.

B. MPEG coding schemes

The MPEG coding algorithm is complex and is based on dividing each picture into blocks, macro-blocks, and slices. In this paper, we assume that each macro-block is coded as an indivisible entity, notably with respect to the choice of the quantization parameter Q. This parameter Q can be used to determine the spatial resolution of each frame. The bit rate and the image quality decrease with increasing values of Q. There are two standard coding schemes for MPEG video: the first is the so-called open loop coding (VBR coding) that allows the generated bit rate to vary without any constraint while keeping the video quality constant (Q is constant); the second is the closed loop coding (CBR coding) scheme, introduced by the ITU-T in [26] to allow transmission of video signals over (telephone lines) narrowband integrated service digital networks (ISDN). This scheme is also suitable for video transmission by using the CBR service in ATM networks. The constant bit rate output stream is achieved in the closed loop encoding scheme by varying the image quality.

In the open loop coding scenario the quantization parameter Q is constant for all macro-blocks resulting naturally in a variable-bit-rate output. The output rate is highly dependent on the image complexity and activity.

The variability of the output VBR video traffic can occur over a range of time-scales:

- packet scale: the data in a given frame may be transmitted in different ways.

  i) As soon as a macro-block is generated, all the bits are transmitted over a macro-block time.

  ii) All the bits of a frame are transmitted at the end of the frame at some peak bit rate for a fraction of the frame duration.

  iii) All the bits of a frame are transmitted at some constant rate calculated to fill the entire frame duration.
- frame scale: the MPEG algorithm introduces inherent bit rate variation from frame to frame due to the succession of I, B and P frames.
- GoP scale: the bit rate averaged over a GoP varies in a correlated manner from GoP to GoP as the image content changes. These changes can be gradual, within a scene, or sharp, in the event of a change of scene.

Figure 1 shows the variation of bit rate averaged over one GoP time with a constant quantization parameter Q. The test was performed on a long video sequence of 1200 frames taken from the movie "The Silence of the Lambs".

The closed loop encoding scheme is designed to allow the fluctuation of the image quality such that the output bit stream can be transmitted at a constant bit rate over circuit-switched networks. For this purpose, the data are

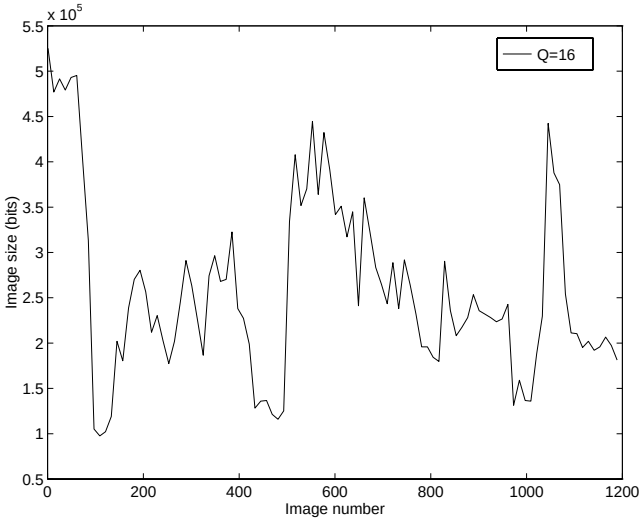


Fig. 1. Open loop coding, bit rate averaged over a GoP time trace

buffered at the encoder and are delivered at a constant rate R. Based on the buffer occupancy, the encoder modulates the quality of the pictures in order to avoid buffer overflow or buffer "starvation". The closed loop encoding scheme is shown in Figure 2. The quantization parameter Q, which determines the resolution and image size of the currently encoded macro-block, changes according to the feedback information on buffer occupancy. The details of the closed loop coding algorithm are given in [27].

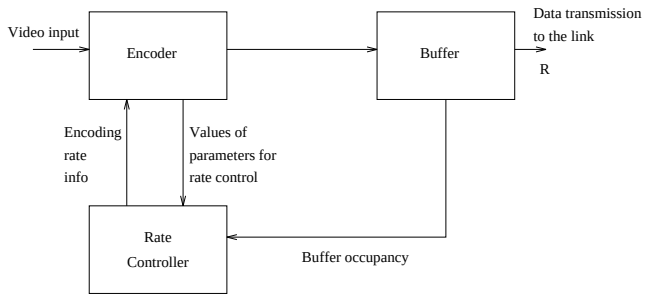


Fig. 2. Implementation of a CBR video encoder

C. Motivation of Our Proposed Scheme

In [14], it was shown empirically that there is a trade-off between the rate variations and the image quality variations as follows:

$$R(k)Q(k) = R_{open}(k)q, \tag{1}$$

where $R(k)$ is the output rate, averaged over the k^{th} GoP, from a closed-loop MPEG encoder, $Q(k)$ is the quantization parameter averaged over the k^{th} GoP and $R_{open}(k)$ is the output rate from an open loop MPEG encoder with the constant quantization parameter q (averaged over the k^{th} GoP). Basically, this relation shows that there must be a compromise between the output rate and the quality of the image. The larger the output rate the better the quality and vice versa. To transport a MPEG video over the ATM-based broadband-ISDN, the closed-loop encoding scheme normally results in a variable image quality unless a high rate is reserved on the network. It is especially true for those video sequences with very high motion (e.g. action movies, music shows, etc). On the other hand, because the video sequence behavior is unpredictable before the transmission starts, it is very difficult to characterize the video behavior and thus to decide beforehand what rate to reserve on the network such that the best quality is achieved. The main advantage of using the closed-loop encoding algorithm is, however, the simplification of the network management procedures, such as admission control, bandwidth allocation, and particularly the lossless transport of the video. From the end user's point of view, it seems to be much more interesting to use an open loop encoding scheme to achieve a constant image quality. However, the variation of the rates makes it impossible to achieve a lossless transport of the video over ATM-based B-ISDN. In addition, and paradoxically, the delay jitter due to the rate variation would finally result in a poor image quality delivery.

Our basic objective is to seek a compromise between these two schemes. In other words, we propose to use a closed-loop MPEG encoder but to allow a small output rate variation similar to the open-loop encoding scheme. Using this approach our aim is to reduce the image quality fluctuation at the encoder with respect to the closed loop scheme, and to reduce the image fluctuation due to the loss and delay jitter in the network. For this purpose we use a shaper to smooth the video output stream based on the provision of transmission credits while keeping a constant quantization parameter q. When either the shaper buffer "threatens" to overflow, which will lead to image quality degradation, or the transmission credits tend towards exhaustion, which would lead to long network access delays, the quantization parameter is changed to slow down the arrival rate to the shaper. However, when the shaper buffer occupancy starts to decrease, which would result in a loss of transmission credits, the quantization parameter is changed to make the image quality better and thus more traffic is sent by the video encoder.

Due to the unpredictability of most video traffic streams, we chose to base our control scheme on fuzzy logic, which is deemed to provide fair solutions for most complex control schemes in various engineering areas. Two fuzzy logic systems operating in two different time-scales are used. The first fuzzy system controls the intra-GoP traffic. It aims to ensure the compliance of the coded video output stream with a predefined sustainable cell rate and burst tolerance parameters to avoid cell dropping at the UNI by the leaky bucket. This is done by shaping the video traffic before it is handed to the network interface. The second fuzzy system operates in the inter-GoP time-scale. Based on the information on the congestion level of the shaper in the current GoP, the fuzzy logic controller changes the quantization parameter Q for the next GoP to either slow down or increase the coding rate. The rate fluctuation of the video output is thus reduced compared to the open loop coding

scheme, making the admission control as well as the bandwidth allocation processes in the network easier. On the other hand, by allowing the video traffic to suffer only a small additional delay in the video coder, our scheme also minimizes the variation of image quality when compared to the closed loop coding scheme.

III. The Fuzzy-Based Rate Control Scheme (FRC)

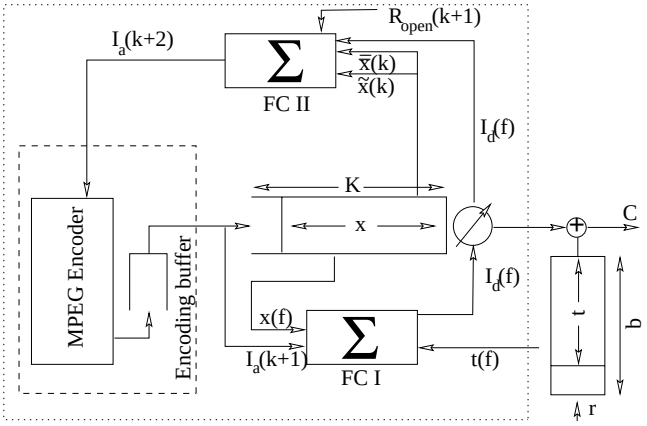


Fig. 3. Implementation of a FRC video encoder

Figure 3 illustrates the fuzzy-based rate control scheme. A fuzzy-based traffic shaper, which consists of a buffer of size K (cells) and a single output link with a capacity C cells per second (the video output peak rate) is implemented in the video coder. To ensure that the output stream conforms to the leaky bucket LB(r,b), the fuzzy controller I (FC I) adjusts the shaper output rate on a frame by frame basis. FC I takes as input the queue length $X(f)$ at the beginning of frame f, the average cell interarrival time $I_a(k+1)$ for the current GoP, and the number of available tokens $t(f)$ in the leaky bucket at the beginning of frame f, and adjusts the output rate of the buffer by either increasing or decreasing the cell interdepartures interval $I_d(f)$ for frame f. On another hand, to prevent the shaper buffer from either overflowing or starving, a second fuzzy controller (FC II) adjusts the average arrival rate to the buffer on a GoP per GoP basis. FC II takes as input an estimation of traffic rate for the next GoP obtained with an open loop algorithm $R_{\text{open}}(k+1)$; the interdeparture time $I_d(f)$ at the end of the current GOP (frame f being the last frame of the GoP); the average shaper queue length $\bar{X}(k)$, and the queue length variance $\tilde{X}(k)$. This statistical information on the current GoP as well as the estimated rate that would have been required for an open loop algorithm is used to calculate the average arrival rate (average interarrival time $I_a(k+2)$) for the next GoP. The MPEG encoder adjusts the quantization parameter as it is done in a closed loop encoding algorithm to accommodate this estimated output rate.

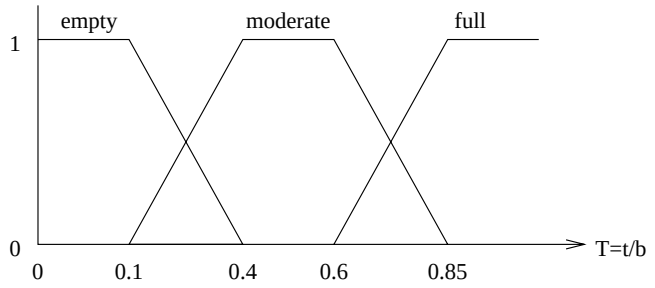


Fig. 4. Membership functions for the linguistic variable "available tokens"

A. The basic fuzzy logic engine

In this section, we briefly introduce the adaptive fuzzy system we use to implement our two fuzzy controllers. These controllers are based on the numerical information obtained from the real-time monitoring of the shaper buffer occupancy statistics, the number of available transmission credits in the leaky bucket, and the input and output rates to and from the shaper. In the following paragraphs, a fuzzy system constructed from numerical information is discussed briefly.

Suppose we have a system whose behavior is difficult to represent accurately by any analytical model. As an example of such system the output rate from an MPEG video encoder: as discussed in [9] can hardly be characterized quantitatively by few traffic parameters to allow accurate mathematical modelling. However, suppose we know qualitatively (from human experience) the approximate behavior of the video traffic. In [28], a general method is provided to design an optimal fuzzy system with universal approximation capabilities. There are a variety of choices in the fuzzy inference engine, the fuzzifier and the defuzzifier. Based on these choices, a number of different fuzzy systems can be constructed. In this paper, we choose the most commonly used fuzzy system with the product inference engine, singleton fuzzifier and center-average defuzzifier given in the following.

$$f(\mathbf{X}) = \frac{\sum_{j=1}^{M} y_c^j \prod_{i=1}^{n} \mu_i^j(x_i)}{\sum_{j=1}^{M} \prod_{i=1}^{n} \mu_i^j(x_i)}, \qquad (2)$$

where $X = (x_1, x_2, \ldots, x_n)^{\mathrm{T}} \in \mathbb{R}^n$ is the input to the fuzzy system, $f(X) \in \mathbb{R}$ is the output of the fuzzy system, y_c^j is the center value of the output fuzzy set in the jth rule and $\mu_i^j(x_i)$, $j = 1, ..., N$, $i = 1, ..., n$, are the membership values of x_i to the fuzzy sets defined in the current system. The derivation steps of the above fuzzy system are beyond the scope of this paper, which focuses mainly on the efficient application of this fuzzy system. The interested reader may refer to [28] for the details and the derivation of (2).

B. Fuzzy Controller I (FC I)

At the beginning of each video frame, two variables (t and x) representing the number of credits available in the

if T is empty and X is empty	then I_d is very large
if T is empty and X is moderate	then I_d is very large
if T is empty and X is moderately full	then I_d is very large
if T is empty and X is very full	then I_d is large
if T is moderate and X is empty	then I_d is very large
if T is moderate and X is moderate	then I_d is medium
if T is moderate and X is moderately full	then I_d is small
if T is moderate and X is very full	then I_d is very small
if T is full and X is empty	then I_d is very large
if T is full and X is moderate	then I_d is small
if T is full and X is moderately full	then I_d is very small
if T is full and X is very full	then I_d is very small

TABLE I
LINGUISTIC RULES OF FUZZY CONTROLLER I

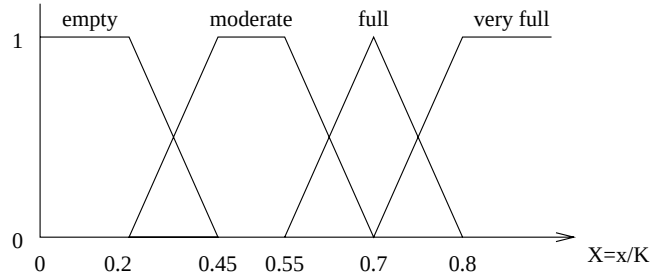


Fig. 5. Membership functions of the linguistic variable "shaper buffer occupancy"

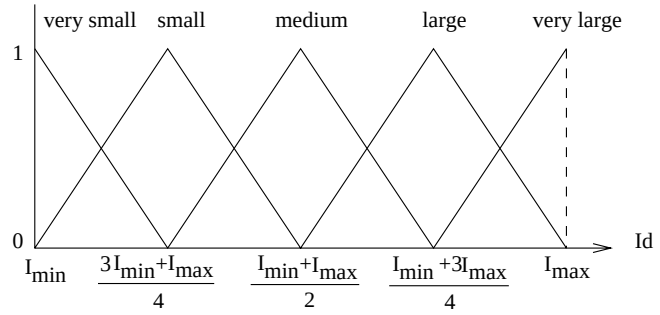


Fig. 6. Membership functions of the linguistic variable "cell inter-departure time from the shaper"

leaky bucket and the current occupancy of the shaper's buffer respectively are fed into the controller FC I. Note that t and x are normalized with respect to b and K respectively. Based on the normalized values T and X of t and x respectively, and the linguistic information stored in the rule base, FC I generates the new cell inter-departure time from the shaper. The control rules are predefined in the rule base shown in Table I. The rules are designed on the basis of human knowledge in order to avoid cell drop/delay at leaky bucket. The linguistic values of the variables are defined by their respective membership functions shown in ⇒ [Figures 4, 5 and 6. Roughly speaking, the membership functions in Figures 4 and 5 are, to some extent, dictated by common sense assumptions in teletraffic: a buffer that is 80% full can be considered as being much closer to congestion and thus loosing cells than the opposite. While a token pool that is 80% full is considered as much closer to wasting bandwidth than to causing starvation. As a common sense rule, more fuzzy sets can be added in the membership function membership functions to result in a smooth adjustment of the output rate, however at the expense of ⇒] more computational complexity.

The center values of the output membership functions are changed dynamically based on the value of the current cell inter-arrival time to the shaper. These output membership functions are designed such that the output rate from the shaper lies always between the leak rate (r) and the current input rate ($1/I_a$) to the shaper, $I_d \in [I_{\min}, I_{\max}]$ with $I_{\min} = \min(I_a(k), 1/r)$, $I_{\max} = \max(I_a(k), 1/r)$. Intuitively, with this approach, the shaper buffer is never empty, which would otherwise lead to a large delay jitter, and the sustainable cell rate is guaranteed because the output stream enters the network at least at rate r.

The fuzzy system presented in Equation (2) is chosen to implement FC I. Intuitively, the controller aims to adjust the output rate of the shaper to meet a compromise between the occupancy of the shaper's buffer and the occupancy of the leaky bucket token pool. In other words, it tries to avoid overflow in the shaper while saving tokens for the next frame so that it would not suffer any access delay. In some senses, FC I tries to keep the token pool half full by pulling t back to either the extreme b or 0 when t approaches either the extreme 0 or b, respectively.

C. Fuzzy Controller II (FC II)

Fuzzy Controller II can be described as follows. At the beginning of the system operation, an MPEG coder encodes the first GoP with a constant average quantization value q. The encoded data are stored in a coding buffer until the encoding of the first GoP is completed. The data in the coding buffer begin to be transmitted to the shaper at the average encoding rate. At the same time, the MPEG coder starts to encode the second GoP using the same average quantization value q. After the first GoP has been completely sent to the shaper, the encoding buffer contains the

second GoP. Again, the second GoP will enter the shaper at the average encoding rate. The average arrival rate to the shaper for the second GoP can be considered as the open-loop rate since the constant quantization value q was used in the encoding. This open-loop rate for the second GoP; the two variables $\bar{x}$ and $\tilde{x}$ (where $\bar{X} \equiv \bar{x}/K$ and $\widetilde{X} \equiv \tilde{x}/K$) for the first GoP denoting the average shaper buffer length and the standard deviation of the shaper buffer length during the transmission of the first GoP into the shaper; and the current output rate of the shaper are fed into FC II. Based on this information and the linguistic information stored in the rule base shown in Table II, FC II estimates the cell inter-arrival time for the third GoP to the shaper.

As the second GoP begins to enter the shaper buffer, the coding proceeds as follows. Let $k = 2$ (i.e. the second GoP has been encoded).

1. Based on $I_a(k+1)$, the value of the cell inter-arrival time to the shaper estimated from FC II, the MPEG coder encodes the $(k+1)^{th}$ GoP using a close-loop coding with the constant rate $1/I_a(k+1)$. At the same time, the data for the k^{th} GoP begin to be transmitted to the shaper buffer using their average encoding rate, $R(k)$.
2. After the complete transmission of the k^{th} GoP into the shaper buffer, the $(k+1)^{th}$ GoP begins to enter the shaper buffer with its average encoding rate, $R(k+1)$ which should be close to the value estimated from FC II (i.e., $1/I_a(k+1)$).
3. The average encoding rate and the average quantization value for the $(k+1)^{th}$ GoP (i.e., $R(k+1)$ and $Q(k+1)$) are used to determine the open-loop rate $R_{open}(k+1)$ for the same GoP using the average quantization q according to (1).
4. $R_{open}(k+1)$, $\bar{x}(k)$ and $\tilde{x}(k)$ denoting the average shaper buffer length and the standard deviation of the shaper buffer length for the k^{th} GoP, and the current shaper output rate $I_d(f)$ are fed into FC II.
5. Based on this information and the linguistic information stored in the rule base (Table II), FC II estimates the arrival rate to the shaper $(1/I_a(k+2))$ for the next GoP (i.e., the $(k+2)^{th}$ GoP).
6. The process continues until all the GoPs are encoded and transmitted.

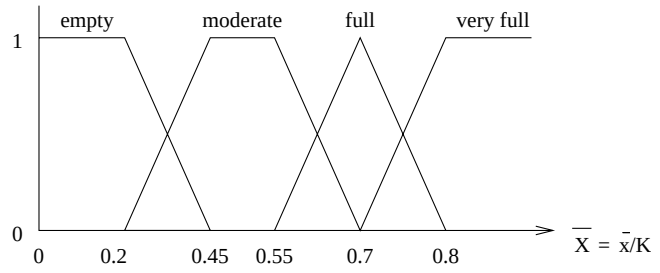


Fig. 7. Membership functions of the linguistic variable "average shaper buffer occupancy"

The control rules in the rule base are predefined on the basis of human knowledge of the qualitative behavior of video traffic so as to avoid cell dropping at the shaper buffer. The linguistic values of each variable are defined by the membership functions shown in Figures 7, 8 and 9. Note that for the membership function in Figure 7 the choice of values is justified by the same arguments as in the previous subsection. Regarding the membership function in Figure 8, theoretically, there is no standard rule to follow in designing this function as it represents the variance of the buffer and thus depends on the activity of the movie. We chose the values shown in Figure 8 by trial-and-error. Tests on different video sequences have shown that these values are quite reasonable, nevertheless, we do not claim that these values suit any video trace. We believe, for a practical implementation of this methodology, the designer should train offline his fuzzy system on a very large number of video sequences with very wide range of bandwidth requirements and burstiness to fine tune the membership functions for $\tilde{X}$. ⇒[⇒]

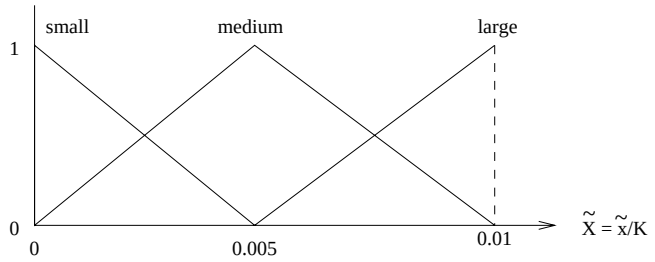


Fig. 8. Membership functions of the linguistic variable "standard deviation of shaper buffer occupancy"

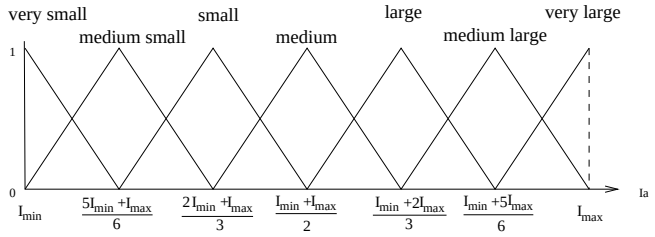


Fig. 9. Membership functions of the linguistic variable "cell inter-arrival time to the shaper"

The center values of the output membership functions are changed based on the values of $I_d(f)$ and $R_{open}(k+1)$ as follows,

$$
\begin{aligned}
&\text{if} \quad (I_{open}(k+1) > I_d(f)) \\
&\qquad I_{min} = 0.8/r \\
&\qquad I_{max} = 1.2/r \\
&\qquad \text{if} \quad (I_{max} > I_{open}(k+1) \text{ and } I_{min} < I_{open}(k+1)) \\
&\qquad\qquad I_{max} = I_{open}(k+1) \\
&\text{else} \\
&\qquad I_{min} = I_{open}(k+1) \\
&\qquad I_{max} = 1/r,
\end{aligned}
\tag{3}
$$

where $I_{open}(k+1)$ is the average output rate of the $(k+1)$th GoP with quantization parameter q, $I_d(f)$ is the current cell inter-departure time from the shaper. In other words, we first assume the value of $R_{open}(k+2)$ is close to $R_{open}(k+1)$ and use $R_{open}(k+1)$ (i.e. $1/I_{open}(k+1)$) as a reference. By defining the center values of the output membership functions as in (3), FC II will increase the

if $\bar{X}$ is empty and $\widetilde{X}$ is small	then I_a is medium small
if $\bar{X}$ is empty and $\widetilde{X}$ is medium	then I_a is small
if $\bar{X}$ is empty and $\widetilde{X}$ is large	then I_a is medium
if $\bar{X}$ is moderate and $\widetilde{X}$ is small	then I_a is medium
if $\bar{X}$ is moderate and $\widetilde{X}$ is medium	then I_a is large
if $\bar{X}$ is moderate and $\widetilde{X}$ is large	then I_a is medium large
if $\bar{X}$ is moderately full and $\widetilde{X}$ is small	then I_a is medium large
if $\bar{X}$ is moderately full and $\widetilde{X}$ is medium	then I_a is very large
if $\bar{X}$ is moderately full and $\widetilde{X}$ is large	then I_a is very large
if $\bar{X}$ is very full and $\widetilde{X}$ is small	then I_a is very large
if $\bar{X}$ is very full and $\widetilde{X}$ is medium	then I_a is very large
if $\bar{X}$ is very full and $\widetilde{X}$ is large	then I_a is very large

TABLE II
LINGUISTIC RULES OF FUZZY CONTROLLER II

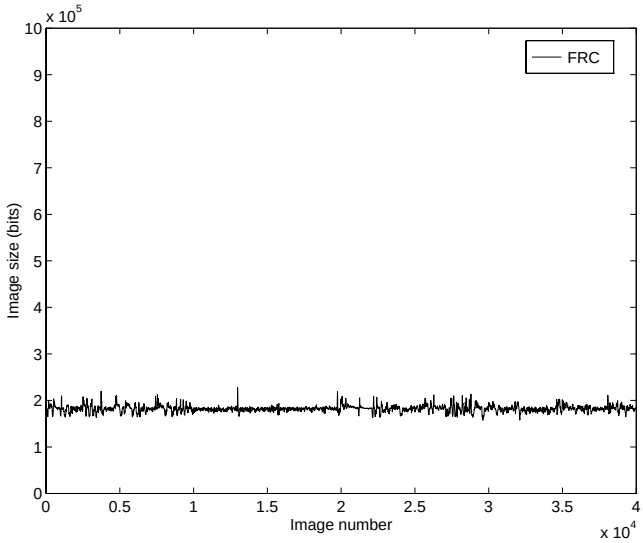
rate for the $(k+2)$th GoP in some small range $[R_{\min}, R_{\max}]$ (where $R_{\min} = 1/I_{\max}$ and $R_{\max} = 1/I_{\min}$) which contains the leak rate $1/r$ if $R_{open}(k+1)$ is smaller than $R_d(f)$ (i.e. $I_{open}(k+1) > I_d(f)$). This can increase the image quality while preventing the shaper's buffer from overflowing. In addition, if $R_{open}(k+1)$ is larger than $R_{\min}$ and at the same time smaller than $R_{\max}$ (i.e., $I_{\min} < I_{open}(k+1) < I_{\max}$), the value of $R_{\min}$ is chosen as $R_{open}(k+1)$ in order to further achieve a better image quality. On the other hand, FC II will reduce the rate for the $(k+2)^{th}$ GoP if $R_{open}(k+1)$ is larger than $R_d(f)$ (i.e. $I_{open}(k+1) < I_d(f)$) but the rate for the $(k+2)^{th}$ GoP will not be smaller than the leak rate so that the image quality can be bounded. Therefore, by using (3) the rate to the shaper will be appropriately bounded by $R_{\min}$ and $R_{\max}$ so that the variation of the image quality can also be bounded.

The fuzzy system (2) is also chosen to implement FC II. The controller aims to adjust the input rate to the shaper to avoid buffer overflow at the shaper. According to the new cell inter-arrival time I_a, a closed loop encoder will start to encode the next GoP to achieve the desired inter-arriva time by adjusting the parameter Q.

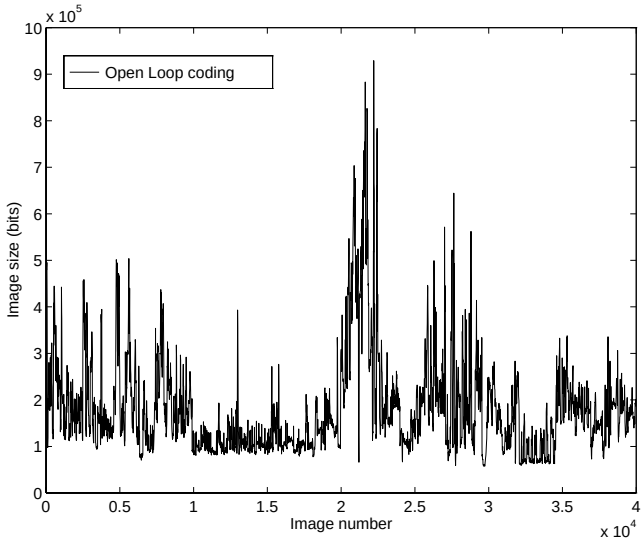
Note that the whole GoP is encoded before it is given t the shaper because, in the frame time-scale, the output rat from the closed loop encoder has a small variation aroun the desired cell rate. Once the whole GoP is encoded, it traffic is fed into the shaper buffer at the rate $R = 1/I_a$.

IV. PERFORMANCE OF FRC

⇒ [The implementation of the FRC algorithm prove straightforward and the required computation time is ver small, which makes it appealing for real-time traffic control To evaluate the performance of the FRC algorithm, a set o experiments on real digitized video sequences has been car ried out. Among these traces, the three sequences whos results we deemed representative of the experiments are th Star Wars movie sequence, a German news sequence, and the silence of the lambs sequence. These sequences have been chosen due to their very high traffic burstiness (see



a: FRC



b: Open loop

Fig. 10. Instantaneous bit rate

Sequence name	Sequence length (GoP)	r (cells/sec)	b (cells)	Burstiness (Peakrate/Meanrate)
Star Wars	10000	908.85	1819	11.88
News	3333	1000.00	2000	12.36
The Silence of the Lambs	3333	476.56	953	18.35

TABLE III

TEST SEQUENCES PARAMETERS

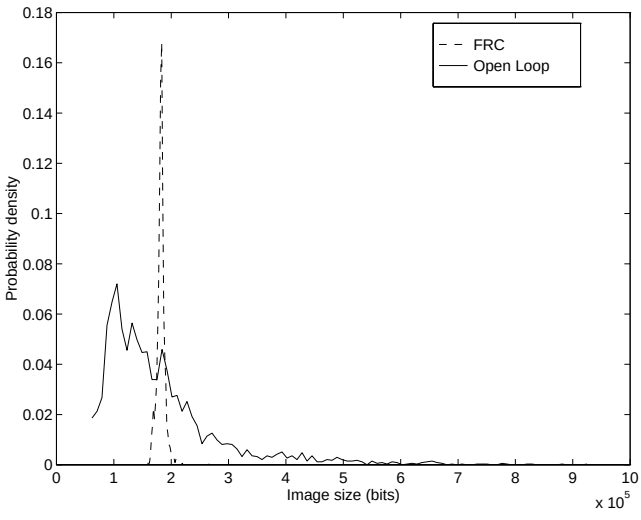


Fig. 11. Probability density of the output rate

Table III). Note that the higher the burstiness the more drastic the variations of the buffer and thus the highest the loss probability. These sequences were originally coded with the open loop coding algorithm with a quantization parameter $q = 16$. The leaky bucket parameters for each sequence have been chosen as r = mean rate of the sequence and $b = 4 * r *$GoP time. The values of r and b for the different sequences are given in Table III. To avoid long delays in the video coder, the shaper buffer size is chosen to be 500 cells. The frame rate for all three sequences is 24 ⇒] frames per second and the GoP size is 12 frames.

A. Rate Variation

In this section, we compare the output rate from the video coder resulting from the FRC algorithm to the one obtained from the open loop coding algorithm. For the sequence of the movie "The Silence of the Lambs ", we show the rate variations in Figure 10.a and Figure 10.b respectively, for the two algorithms. The stationary distributions of the frame bit rate observed from the open loop coding algorithm and the FRC algorithm are shown in Figure 11. The tests with the other video sequences showed similar results. For instance, the mean and variance of the output rate from the FRC algorithm are given in Table IV with respect to the mean and variance obtained from the open loop coding algorithm. We notice from the results that the FRC algorithm reduces the burstiness of the video output rate when compared with the output rate of the open loop algorithm. The slight increase in the mean rate is due to the improvement in the image quality (i.e., more data are produced) as shown in the next subsection.

In addition, by shaping the video traffic as in the FRC algorithm, the long-range dependence phenomenon can be avoided. This observation is shown through the comparison of the Hurst parameter for the bit rate obtained from the FRC algorithm with the one obtained from the open loop coding. In Table V, the Hurst parameter for all three sequences is close to 0.5 for FRC, which implies that the only relevant dependence is the short-range dependence. As shown in the same table, the original open loop coded sequences exhibit a significant long-range dependence.

B. Image Quality Variation

Table VI illustrates the effects of the FRC algorithm on the image quality. In this table, we compare the mean and the variance of the quantization parameter with those obtained from a closed loop coding algorithm. In the closed loop coding, the value of the data transmission rate for each sequence has been chosen as the value of the corresponding leak rate r (i.e. exactly CBR) in Table III; such that the data obtained from this coding can pass through the same leak bucket without loss/delay. The values of the quantization parameter for each GOP are calculated by using (1) with $q = 16$ and $R(k) = r$. With respect to the closed loop coding algorithm, we notice that the FRC algorithm reduces the average Q by a few per cent while its variance is reduced substantially. Note that the smaller the Q, the better the image quality; and the less Q varies, the more steady is the image quality. Compared with the closed loop algorithm, the FRC algorithm provides a relatively constant and good quality image.

C. Delay

An additional cell delay is experienced in the shaper. The delay statistics measured from the tests are shown in Table VII. From the results we observe that the mean delay is always approximately half the maximum delay. Moreover, the variance of the delay is very small which means that the mean delay can be used as a delay bound instead of the maximum delay since only one GoP at most may experience this maximum delay. In addition, for the three sequences, the maximum delay always seems to be smaller than one GoP time.

To illustrate the effect of the different system parameters on the cell delay, "The Silence of the Lambs" video sequence is used for the following tests. First, we investi-

Sequence	% of decrease in variance	% of increase in mean
Star Wars	98.8942	1.0739
News	99.149	2.2201
The Silence of the Lambs	99.4958	1.2448

TABLE IV

Comparison of video output rate statistics

Sequence name	Open loop coding	FRC
Star Wars	0.8042	0.5978
News	0.7101	0.5358
The Silence of the Lambs	0.9165	0.5908

TABLE V

Comparison of the Hurst parameter of the output streams

Sequence	% of decrease in variance	% of decrease in mean
Star Wars	17.6281	3.1131
News	17.1123	3.6294
The Silence of the Lambs	14.2084	4.0835

TABLE VI

Comparison of quantization parameter

Sequence name	variance of delay	mean delay	max delay
Star Wars	0.0058	0.2359	0.4373
News	0.00504	0.2065	0.3915
The Silence of the Lambs	0.02144	0.4587	0.8824

TABLE VII

Statistics of delay experienced in the shaper (in seconds)

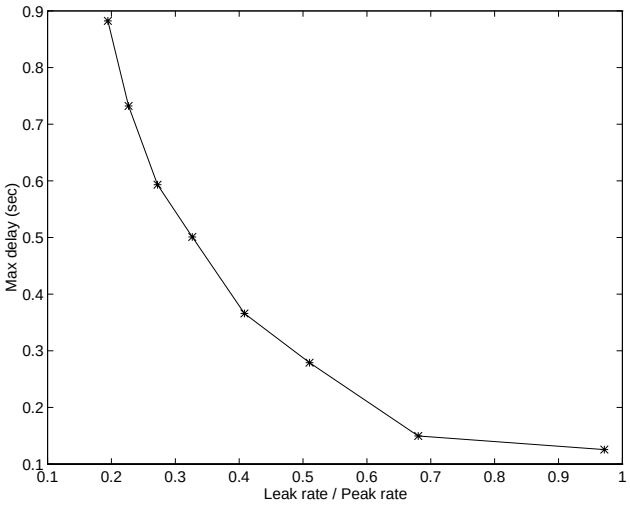


Fig. 12. Max delay vs. leak rate/peak rate

gate the effect of the leak rate on the cell delay by varying the leak rate while the token pool size is kept constant. Figure 12 shows that the maximum delay decreases exponentially with increasing leak rate. Second, we investigate the effect of the token pool size on the delay by varying the token pool size while keeping the leak rate constant. As shown in Figure 13, we observe that the delay decreases slightly by increasing the token pool size. Finally, we observe in Figure 14 that the delay increases linearly if the shaper buffer size increases. This implies that the value of leak rate has the largest relief effect on the delay in the video coder. By choosing the appropriate value for the leak rate, or by increasing it slightly, the delay experienced in the shaper's buffer can be minimized or decreased.

V. Conclusion

In this paper we have considered the problem of using a preventive traffic control mechanism to simplify the network management while improving the image quality for real-time VBR video applications. In this spirit, we

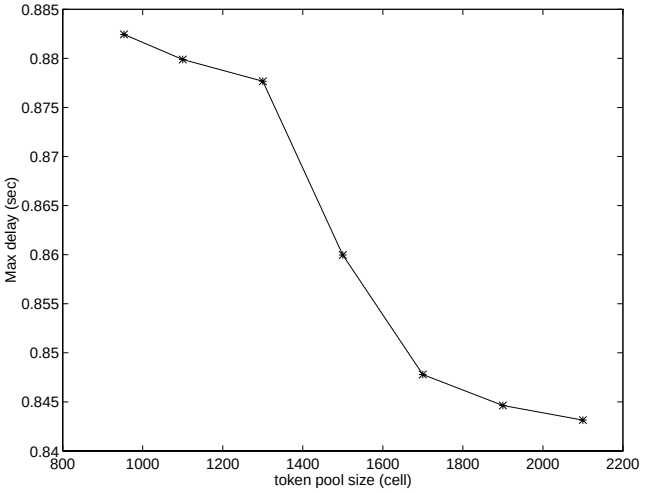


Fig. 13. Max delay vs. token pool size

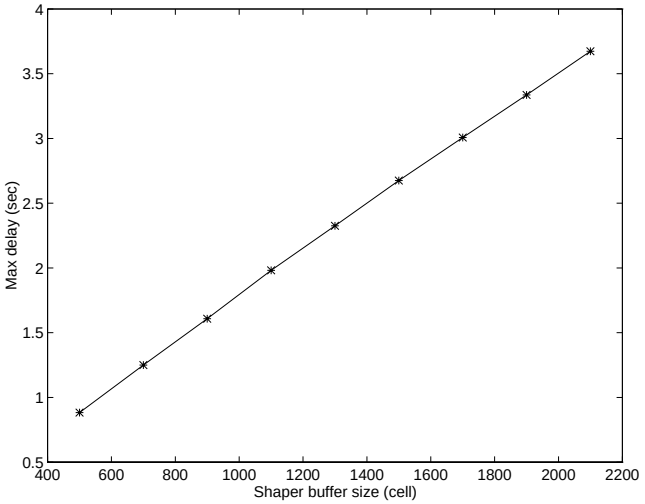


Fig. 14. Max delay vs. shaper buffer size

proposed a fuzzy logic based control scheme for real time MPEG video to avoid long delays or excessive losses at the user-network interface (UNI) in an ATM network. The scheme consists of associating a shaper with the MPEG video encoder. The role of this shaper is to smooth the MPEG output traffic in order to reduce the burstiness of the video stream. The input rate and output rate of the shaper buffer are controlled by two fuzzy logic based controllers. To avoid long delays at the shaper, the first controller aims to tune the output rate of the shaper based on the number of transmission credits available at the UNI and the occupancy of the shaper's buffer in the video frame time-scale. Based on the average occupancy of the shaper's buffer and its variance, the second controller role is to tune the input rate to the shaper by applying a closed loop MPEG encoding scheme in the group of picture time-scale. With this approach, the traffic enters the network at an almost constant bit rate (with a very small variation) allowing simple network management functions such as admission control and bandwidth allocation, while guaranteeing a steady video quality because the encoding rate is changed only in critical periods where the shaper buffer "threaten" to overflow. A performance study of the proposed scheme was carried out through numerical tests and simulations on real video sequences. As expected, our tests have shown that, unlike the classic method of allowing the video directly into the network, our proposed scheme reduces substantially the burstiness of the traffic with respect to the raw open-loop encoding scheme, while improving the steadiness of the image quality with respect to the raw closed-loop encoding scheme. In addition, the empirical experiments have shown that our scheme reduces the the self-similar nature of the video stream, thus allowing the use of the classic Markovian modelling when feeding the output traffic of our scheme to an ATM multiplexer.

References

[1] ATM Forum, *"Traffic Management Specifications"*. ATM Forum, March 1996.

[2] M. W. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic," *Proceedings of ACM SIGCOMM '94*, pp. 269–280, September 1994.

[3] H. Heeke, "Statistical multiplexing gain for variable bit rate video codecs in ATM networks," *International Journal of Digital and Analog Communication Systems.*, vol. 4, pp. 261–268, 1991.

[4] D.P.Heyman, A.Tabatabai, and T. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in ATM networks," *IEEE Transactions on Circuits and Systems for Video Technology.*, pp. 49–59, March 1991.

[5] B. Maglaris, D. Anastassious, P. Sen, G. Karlsson, , and J. D. Robbins, "Performance models of statistical multiplexing in packet video communications," *IEEE Transactions on Communications*, vol. 36, pp. 834–844, July 1995.

[6] M. Nomuar, T. Fujii, and N. Ohta, "Basic characteristics of variable rate video coding in ATM environment," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 752–760, June 1989.

[7] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger, "Long-range dependence in variable-bit-rate video traffic," *IEEE Transactions on Communications*, vol. 43, no. 2/3/4, pp. 1566–1579, 1995.

[8] A. Elwalid, D. Heyman, T. Lakshman, A. Weiss, and D. Mitra, "Fundamental bounds and approximations for ATM multiplexers with application to video teleconferencing."

[9] D.P.Heyman and T. Lakshman, "Source models for VBR broadcast-video traffic," *IEEE/ACM Transactions on Networking*, vol. 4, pp. 40–48, February 1996.

[10] A. R. Reibman and B. Haskell, "Constraints on variable bit rate video for ATM networks," *IEEE Transactions On Circuits and Systems for Video Technology*, vol. 2, pp. 361–372, December 1992.

[11] H. Heeke, "A traffic control algorithm for ATM networks," *IEEE Transactions on Circuits and Systems for Video Technology.*, vol. 3, pp. 182–189, June 1993.

[12] R. Coelho and S. Tohme, "Video coding mechanism to predict video traffic in ATM network," in *IEEE GLOBECOM'93*, pp. 447–451, 1993.

[13] M. R. Pickering and J. F. Arnold, "A perceptually efficient VBR rate control algorithm," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 527–532, 1994.

[14] M. Hamdi and J. W. Robert, "QoS guarantees for shaped bit rate video connections in broadband networks," *Proceeding of MmNet95.*, September 1995.

[15] K. F. Cheung, D. H. K. Tsang, C. C. Cheng, and C. W. Liu, "Fuzzy logic based ATM policing," in *IEEE ICCS'94*, pp. 535–539, 1994.

[16] C. Douligeris and G. Develekos, "A fuzzy logic approach to congestion control in atm networks," in *IEEE International Conference on Communications*, pp. 1969–1973, 1995.

[17] V. Catanaia, G. Ficili, S. Palazzo, and D. Panno, "A comparative analysis of fuzzy versus conventional policing mechanisms for ATM networks," *IEEE/ACM Transactions on Networking*, vol. 4, pp. 449–459, June 1996.

[18] A. Pitsillides, A. Şekercioğlu, and G. Ramamurthy, ""Fuzzy backward congestion notification (FCBN) congestion control in asynchronous transfer mode (ATM) networks"," in *IEEE Globecom'95*, pp. 280–285, 1995.

[19] P. Chemouil, J. Khalfet, and M. Lebourges, "A fuzzy control approach for adaptive traffic routing," *IEEE Communications Magazine*, vol. 33, July 1995.

[20] S. Lam, B. Bensaou, and D. Tsang, "Efficient estimation of cell loss probability in atm multiplexers with a fuzzy logic system," in *International IFIP-IEEE Conference on Broadband Communications* (L. Mason and A. Casaca, eds.), pp. 306–317, 1996.

[21] B. Bensaou, S. T. C. Lam, H.-W. Chu, and D. H. K. Tsang, "Estimation of the cell loss ratio in atm networks with a fuzzy system and application to measurement-based call admission control," *IEEE/ACM Transactions on Networking*, vol. 5, pp. 572–584, August 1997.

[22] C. Douligeris and G. Develekos, "Neuro-fuzzy control in atm networks," *IEEE Communications Magazine*, vol. 35, pp. 154–162, May 1997.

[23] P. Pancha and M. El Zarki, "MPEG coding for variable bit rate video transmission," *IEEE Communications magazine*, vol. 32, pp. 54–66, May 1994.

[24] D. Le Gall, "MPEG: A video compression standard for multimedia applications," *CACM*, vol. 34, no. 4, pp. 46–58, 1991.

[25] ISO-IEC, "Coding of moving frames and associated audio." ISO-IEC/JTC1 SC29 draft, November 1991.

[26] ITU-T, "Line transmission of non-telephone signals: Video codec for audiovisual services at $p \times 64$ kbits." ITU-T Recommendation H.261, November 1991.

[27] ISO-IEC, "Coded representation of picture and audio information MPEG test model 2." ISO-IEC/JTC1/SC29/WG11, July 1992.

[28] L. X. Wang, *Adaptive fuzzy system and control: design and stability analysis*. Prentice Hall, Inc., 1994.