An MPEG2 to ATM Converter to Optimize Performance of VBR Video Broadcast over ATM Networks

By

Paul Wong

B. Eng. Sc., University of Waterloo, 1995

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS OF THE DEGREE OF

MASTER OF APPLIED SCIENCE

In

THE FACULTY OF GRADUATE STUDIES

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

October 1999

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical & Computer Engineering

The University of British Columbia
Vancouver, Canada

Date Oct. 14. 99

DE-6 (2/88)

Abstract

Whereas variable bit-rate (VBR) MPEG2 (Moving Pictures Expert Group) video is desirable for broadcasting applications due to its conservation of bandwidth, the resulting traffic characteristics are content-dependent and unpredictable. Before broadcasting video over an ATM (Asynchronous Transfer Mode) network, a service contract is established specifying the source traffic descriptors and the required quality of service (QoS). Usage parameter control (UPC) is performed to ensure conformity of the source traffic to its descriptors. This protects the network from misuse of resources by misbehaving sources, which can affect the QoS of other connections. For real-time variable bit-rate (rt-VBR) traffic, the key traffic descriptors for UPC are PCR (peak cell rate), MBS (maximum burst size) and SCR (sustainable cell rate). ATM cells conforming to the UPC parameters are guaranteed delivery with the specified QoS, while nonconforming cells are subject to loss when congestion occurs. As cell loss results in the degradation of the received video quality, source rate control is usually required to minimize their effects on video quality relative to human visual perception.

This thesis presents a novel MPEG2-to-ATM converter for VBR video broadcast over ATM networks. It functions as an external post-coding source rate controller between any MPEG2 source (video archives, life programs, etc.) and an ATM network. Dynamic break points are employed to shape and partition the video data into high priority (HP) ATM cells which conform to the prevailing usage parameter control contract with the network, and nonconforming low priority (LP) cells which minimize the effects of cell loss due to network congestion on the subjective quality of the received video. Both high and low priority cells are transmitted over a common ATM virtual connection. Performance evaluations of actual VBR MPEG2 streams transmitted over a simulated converter and ATM network are presented to demonstrate the effectiveness of the proposed method.

Table of Contents

List of Tables

List of Figures

Acknowledgment

I would like to thank Dr. Victor C.M. Leung and Dr. Panos Nasiopoulos for their supervision and useful suggestions in this thesis. This research was supported by a grant from Canadian Institute for Telecommunications Research under the NCE program of the Government of Canada.

Chapter 1 Introduction

This introductory chapter gives the motivations for the development of an MPEG2-to-ATM Converter. Previous works on this research area are highlighted. The objectives of this thesis are stated. The topics that will be covered in subsequent chapters are outlined at the end of this chapter.

1.1 Motivations

A large number studies have been performed on the statistical characteristics of video sequences (e.g. [10], [21], [30], [37], [42]). From these studies, it is clear that variable bit rate (VBR) MPEG2 (Moving Pictures Expert Group) traffic is very content-dependent and thus almost unpredictable. Furthermore, VBR MPEG2 traffic patterns are far more complicated compared to other VBR traffic such as video conferencing and on-off source [29]. Although uncontrolled VBR coded video achieves consistent picture quality, the traffic stream is very bursty. Hence, bandwidth estimation for such traffic is a very complex task. The considerable amount of computation time would not be acceptable for real time applications.

With advantages such as guaranteed quality of service (QoS) and statistical bandwidth sharing ([5], [14], [22], [37], [38]), Asynchronous Transfer Mode (ATM) networks seem to be the most promising network architecture that may satisfy a wide variety of traffic types include real time variable bit rate (rt-VBR) sources [34]. However, buffer capacities at the switches are limited. ATM based networks are subject to congestion that could introduce random delays and cells losses due to buffer overflow. A very challenging task is to maintain the advantage of the consistent video quality offered by VBR source coding when there are delays and cell losses caused by limited bandwidth.

Therefore, enabling high quality rt-VBR video transmission in ATM networks is the key motivation of this research.

1.2 Previous Work

In order to avoid network congestion, traffic control (or traffic policing) is implemented in ATM networks [35, 3] to force the user to follow the established ATM traffic contract between the ATM networks and the users. When using an ATM network, a service contract is established specifying the source traffic descriptors and the required QoS. Usage parameter control (UPC) is performed to ensure the conformity of the source traffic to its descriptors. Policing service contract conformity protects the network from misuse of resources by misbehaving sources, which can affect the QoS of other connections. For rt-VBR traffic, the key traffic descriptors for UPC are PCR (peak cell rate), MBS (maximum burst size) and SCR (sustainable cell rate) [3]. ATM cells conforming to the UPC parameters are guaranteed delivery with the specified QoS, while nonconforming cells are subject to losses when congestion occurs. As cell losses result in the degradation of the received video quality, some sort of source traffic control is usually required to minimize the effects of cell losses on video quality degradation according to human visual perceptions [25].

Studies [30] and [37] have shown that a well designed dynamic resource renegotiations scheme can match the bit rate variation, and hence maintain a high level of video quality and good bandwidth utilization. However, renegotiations are not a deterministic process; renegotiation delays may occur or the request may be rejected. Between two successful renegotiations, the bandwidth requirement might change again resulting in the updated contract not reflecting the source bandwidth requirement and entering the renegotiations process again.

Another approach uses buffer delay. It simply performs traffic smoothing by buffering a certain number of frames at a time, then sending those frames at the average rate. This approach can be suitable for non-real time VBR transportation, but not real time VBR traffic [29]. Furthermore, for real time interactive video services, the smoothing period needs to be as small as the maximum tolerable end-to-end delay that can be at most 10 to 15 frames [23]. A previous study [31] has shown that for uncontrolled VBR video traffic, this approach does not yield any substantial gain when the smoothing period is limited to be less than 15 frames.

One of the MPEG-2 video's major extensions over MPEG-1 is the addition of scalability, which can support applications with multiple-layer video. One such application involves the provision of several different qualities of service to accommodate various classes of users. A two-layered source coding scheme generates a base layer which contains the basic video information. This base layer can be solely decoded to provide a basic quality of service. The residual information is coded as an enhancement layer so that when received and added to the base layer, an enhanced video quality can be obtained. [15], [25], and [26] have used a priority scheme similar to the idea of "Data Partitioning" as described in the MPEG-2 video compression standard [16] to have multiple layer outputs. Furthermore, authors in [7] proposed the use of a hierarchical encoding scheme and a dynamic data-partitioning strategy to output multiple layers of MPEG-2 video streams. However, multiple transmission channels are required. In other words, multiple ATM contracts, multiple bandwidth estimation, and other considerations are required. Synchronization becomes an issue at the receiver end of ATM networks.

An add-on approach is to generate a controlled VBR bit-stream that conforms to specified bit rate boundaries and buffer constraints, and produce picture quality degradation at acceptable levels. This approach is known as source rate control [15] [26]. Data partitioning in [9], [32], and [38] was based on splitting the DCT (Discrete Cosine Transform) coefficients at some predetermined point resulting in two partitions of the bit-streams. Recently, adaptive data partitioning algorithms have been proposed. Luo and Zarki [26] designed an adaptive DP scheme within the MPEG framework which has the capability of detecting long bursts and reacting accordingly to stay within the ATM contract requirement. In the document [11], M. Hamdi, J. Roberts and P. Rolin presented a rate control algorithm that adjusts the coder quantization parameter on a GOP (Group of Pictures) -by-GOP basis in order to ensure that the output satisfies the burstiness constraint imposed by the leaky-bucket traffic control. Reibman and Haskell [36] present bit rate constraints that prevent codec buffer overflow in the case of a leaky-bucket-controlled channel. Heeke [13] and Coelho [6] tried to make the output behave like a predefined Markov chain. Pickering and Arnold [33] proposed a rate control algorithm that produces a VBR traffic lying between predefined upper and lower bounds. Pancha and Zarki [31] also used rate controls with an algorithm similar to that of [36] where a bucket, that is few frames long, is used to control traffic variability. However, most of these source rate control schemes are implemented inside the MPEG2 encoder.

1.3 Objective

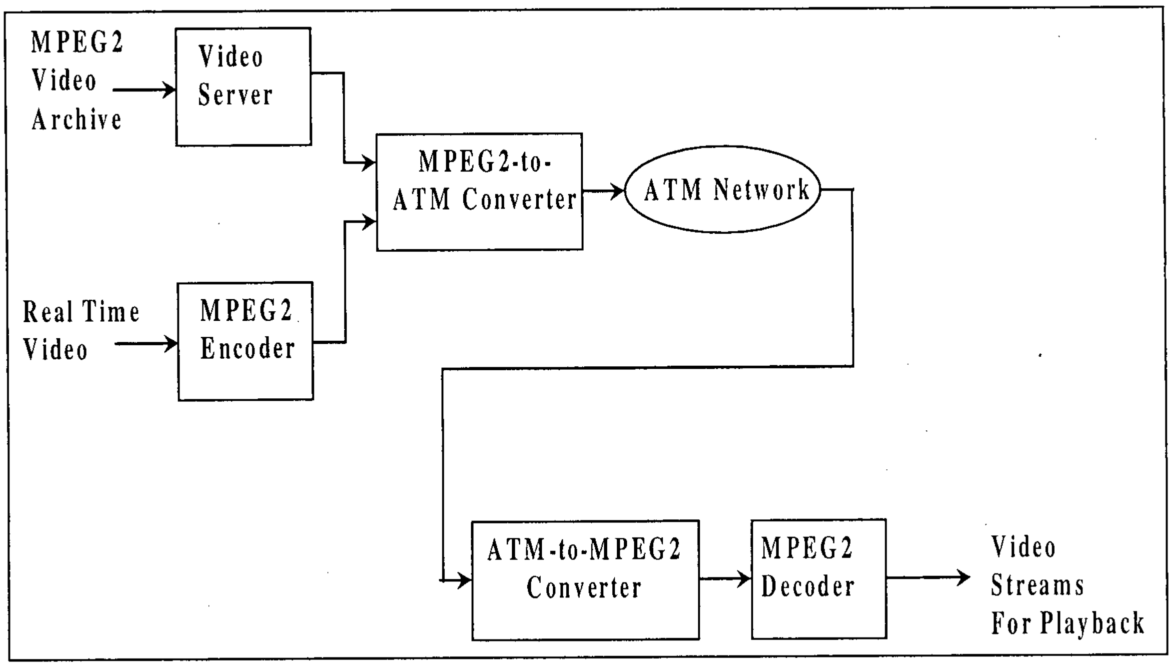

Figure 1-1: Application of MPEG2-to-ATM Converter

As source rate control at the encoder is not always practical or desirable, e.g., in the cases of pre-encoded video archives, or video traffic originating from another network that has no notion of UPC. We propose an MPEG2-to-ATM converter, which functions as an external post-coding source rate controller between any MPEG2 source (archives, life programs, output of a MPEG2 encoder, etc.) and ATM network (please refer to Figure 1-1). Dynamic break points (BPs) are employed to shape and partition the video data into high priority (HP) ATM cells which conform to the UPC contract, and low priority (LP) ATM cells which are subject to network congestion induced losses. The partitioning aims minimizing the effects of cell loss on the subjective quality of the received video. At the same time, the proposed MPEG2-to-ATM converter shall be easy to implement and adapt to existing MPEG2 encoders.

1.4 Thesis Outline

The outline of this thesis is as follows:

Chapter 2 presents background knowledge of the MPEG2 video standard. Topics include a brief introduction, its differences from MPEG1, bandwidth requirement, data structure and format. Then the concept of MPEG2 data partition is given in detail. At last, ATM traffic control, ATM VBR traffic requirements, and ATM Adaptation Layer 5 (AAL5) are briefly described.

Chapter 3 first gives an overview of the proposed MPEG2-to-ATM converter. Then the structure diagram and operation flow are presented. Two key components: the data structure and break point assignment algorithm, are then described. In this chapter, the ATM-to-MPEG2 converter is also described.

Chapter 4 presents the simulation model. It includes five modules: "Traffic Manager", "Simulated Leaky Bucket", "MPEG2 video-AAL5 PDUs-ATM Cells Converter", "ATM Channel", "ATM cell-AAL5 PDUs-MPEG2 Video Converter" and "Observation Center". In addition to these modules, video streams, assumptions in the simulation, and evaluation cases are discussed.

Chapter 5 describes the evaluation test cases first. Then the simulation results are discussed in four different areas. They include overhead introduced by LP data structure, preset vs. effective ATM traffic parameters, average PSNR comparison, channel efficiency and frame by frame PSNR evaluation.

Chapter 6 gives the conclusions.

Chapter 2 Background

In this chapter, the MPEG2 standard is introduced first. Topics include MPEG2's brief history, its differences from MPEG1, bandwidth requirement, data structure and format. Then the concept of MPEG2 data partition is given in detail, because data partition is the basic idea for the break point algorithms in this thesis. At last, ATM traffic control, ATM VBR traffic requirements, and ATM Adaptation Layer 5 (AAL5) are briefly described.

2.1 MPEG2 Video

2.1.1 Introduction

MPEG2 Video is a generic method for compressed representation of video sequences using a common coding syntax defined in the document [16] by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC).

MPEG2 Video is the successor to MPEG1 Video Standard (ISO/IEC IS 11172-2), which also supports interlaced video formats and a number of other advanced features, including features to support HDTV (High Definition Television) [28]. MPEG2 is intended to be generic, supporting a diverse range of applications. Different algorithmic elements or 'tools', developed for many applications, have been integrated into a single bit-stream syntax. To implement the full syntax in all decoders is unnecessarily complex, so a small number of subsets of the full syntax, or 'profiles' have been defined. Also, within a given profile, a 'level' is defined which identifies a set of constraints imposed on parameters within the syntax subset. A decoder which supports a particular profile and level is only required to support the corresponding subset of the full syntax and set of parameter constraints [39].

The profiles are separated into the following categories [28]:

- **Simple:** Same as Main, only without B-pictures. Intended for software applications, perhaps CATV.

- **Main:** Most decoder chips, CATV, satellite. 95% of users.
- **Main+:** Main with Spatial and SNR scalability.
- **Next:** Main+ with 4:2:2 macroblocks.

Table 2-1 shows the different levels, the corresponding sampling limits, and bit rates [28].

Level	Max. sampling	Pixels/sec	Max. bit rate
Low	352 x 240 x 30	3.05M	4 Mb/s
Main	720 x 480 x 30	10.40 M	15 Mb/s
High 1440	1440 x 1152 x 30	47.00 M	60 Mb/s
High	1920 x 1080 x 30	62.70M	80 Mb/s

Table 2-1: Different levels, the corresponding sampling limits, and bit rates.

At the Sydney MPEG meeting, the MPEG2 Main Profile was defined to support digital video transmission in the range of about 2 to 15 Mb/sec over cable, satellite, and other broadcast channels, as well as for Digital Storage Media (DSM) and other communications applications. In fact, the Main Profile meets the needs of 95% of the users [28]. In this thesis, MPEG2 Simple Profile (same as Main, only without B-pictures) bit-streams are used for evaluation.

2.1.2 MPEG2 vs. MPEG1

MPEG2 Video is a syntactic superset of MPEG1 Video [12]. As in MPEG1 Video, MPEG2 Video is based on motion compensated DCT (Discrete Cosine Transform) coding, Block-pictures, and a Group-of-Pictures structure. Furthermore, in MPEG2 video standard, variable quantization is extended, DC precision can be customized, and parameter ranges such as picture width, bit rate, and buffer sizes are extended. Copyright information can be recorded. Display characteristics can be transmitted including an offset for viewing a subset of the picture in a window [27]. In short, MPEG2 supports more applications, more aspect ratios, higher resolutions, and much larger frame sizes than MPEG1. Therefore, MPEG2 provides much better video quality, higher bit rate, better compression and more flexibility. For example, the Main Profile and Main Level (abbreviated as MP@ML) supports resolutions as high as 704x480 (with maximum 15 Mb/s data rate), four times as high as MPEG1 (with about 1.5 Mb/s constant data rate), which is generally used in VCD (Video Compact Disc).

2.1.3 Bandwidth Requirement of MPEG2 Video

With variable quantization, variable bit rate and high bit rate (up to 15 Mb/sec for MP@ML), MPEG2 gives much better video quality over MPEG1. However, as stated in Chapter 1, exact bandwidth estimation for MPEG2 traffic is a very complicated and time-consuming task. For example, a normal news clip may include silent moments such as still-screen play and actions such as sports within a very short period. In order to maintain the same high picture, sports action scenes require much more bandwidth than still-image scenes. Channel over-booking causes bandwidth under-utilization, but under-booking causes data loss, which leads, consequently, to video degradation. To address this issue, MPEG2 supports data partitioning. MPEG2 video data partition is described in detailed in section 2.2.

2.1.4 Data Structure and Format of MPEG2 Video

The MPEG2 standard defines a hierarchy of data structures in the video stream as shown in Figure 2-1. There are six layers: the sequence layer, the group of pictures (GOP) layer, the picture layer, the slice layer, the macroblock layer, and the block layer.

A video sequence begins with a sequence header (which may contain additional sequence headers), followed by one or more GOPs, and ends with an end-of-sequence code. Each GOP includes a header and a sequence of pictures. The picture is the primary coding unit of a video sequence. Generally, there are three types of pictures in MPEG2 video, namely, the intra-frames (I-frames), the predicted frames (P-frames) and the bi-directional frames (B-frames). An I-frame is the start of a certain coding period and it is obtained by simply encoding an individual picture without utilizing any related information in the past frames. The I-frame is always the starting frame of a GOP and is intended to allow random access. A P-frame contains the information that is predicted from the immediate previous I- or P-frames. A B-frame is generated by considering the contents in both the past and future I- or P-frames immediately adjacent to it. The coded video sequence is in a format like I B B P B B P B B P I B B P. In this specific example, the GOLP size is 10 frames (number of frames between two I frames plus one).. The slice is one or more "Contiguous" macroblocks. Slices are the basic units for error handling. If the bit-stream contains an error, the decoder can skip to the start of the next slice. A macroblock is a 16-pixel by 16-line section of luminance components and the corresponding 8-pixel by 8-line section of the two chrominance components. A block is an 8-pixel by 8-line set of values of a luminance or a chrominance component.

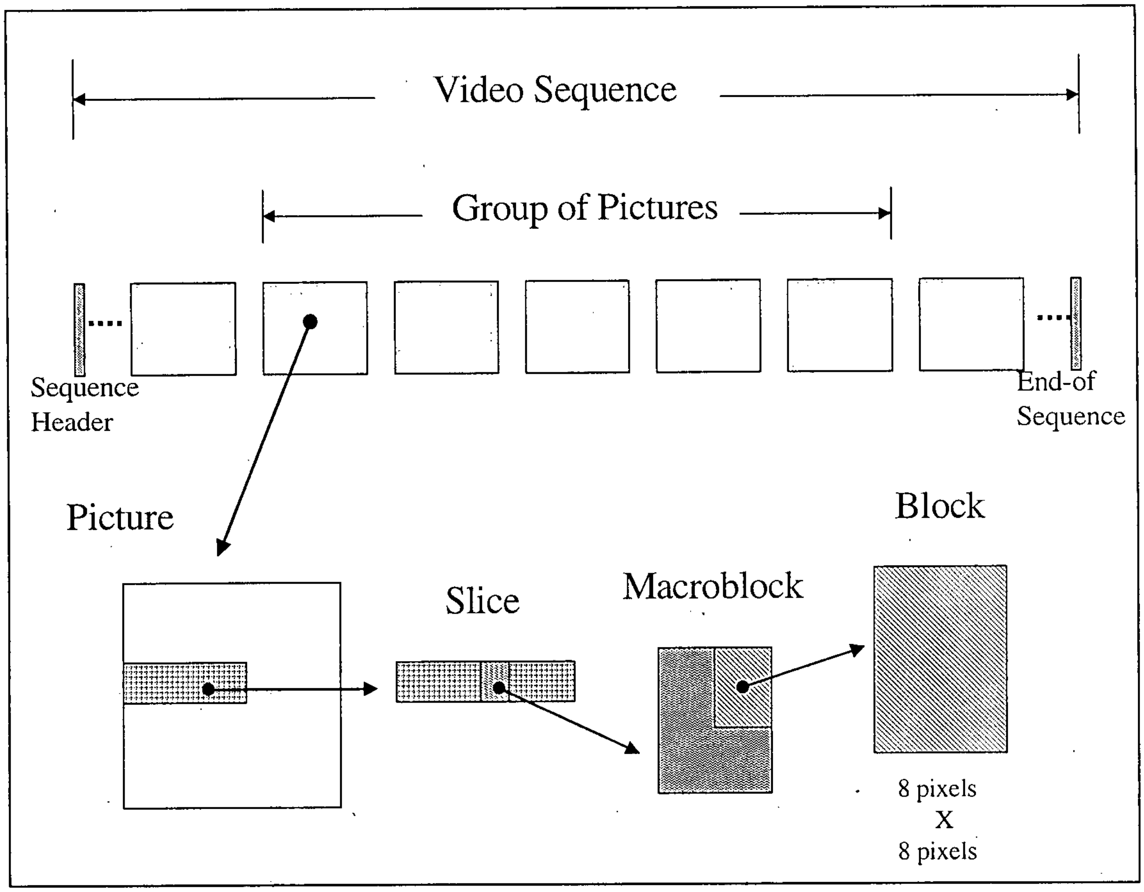

Figure 2-1: MPEG2 Video Data Structure

2.2 MPEG2 Data Partition

In order to reduce high spatial redundancy in both image blocks and prediction-error blocks, the MPEG2 algorithm transforms 8x8 blocks of pixels or 8x8 blocks of error terms from the spatial domain to the frequency domain with the Discrete Cosine Transform (DCT).

Then, the encoder chooses a quantization matrix that determines how each frequency coefficient in the 8 X 8 block is quantized. Quantization is the process of assigning each frequency coefficient to one of a limited number of values. Human perception of quantization error is lower for high spatial frequencies, so high frequencies are typically quantized more coarsely (i.e., with fewer allowed values) than low frequencies.

The combination of DCT and quantization results in many of the frequency coefficients being zeros, especially the coefficients for high spatial frequencies. To take maximum advantage of this, the coefficients are organized in a zigzag order to produce long runs of zeros. The coefficients are then converted to a series of run-amplitude pairs, each pair indicating a number of zero coefficients and the amplitude of a non-zero coefficient. These run-amplitude pairs are then coded with a variable-length code, which uses shorter codes for commonly occurring pairs and longer codes for less common pairs.

The MPEG2 is similar to the JPEG's (Joint Photographic Experts Group) frequency progressive mode where only the slice layer indicates the maximum number of block transform coefficients contained in the particular bit-stream (known as the priority break point or PBP). In other words, PBP information is stored in slice headers. Data partitioning is a frequency domain method that breaks the block of 64 quantized transform coefficients into two bit-streams. The first, a higher priority bit-stream contains the more critical lower frequency coefficients and side information (such as DC values, motion vectors). The second, a lower priority bit-stream carries higher frequency AC data.

2.3 ATM Traffic Control and VBR Requirements

ATM is a connection-oriented method that transfers service information by establishing a virtual channel. Whenever a virtual channel is set up, a connection identifier is assigned, and when the connection is removed, the identifier is removed as well.

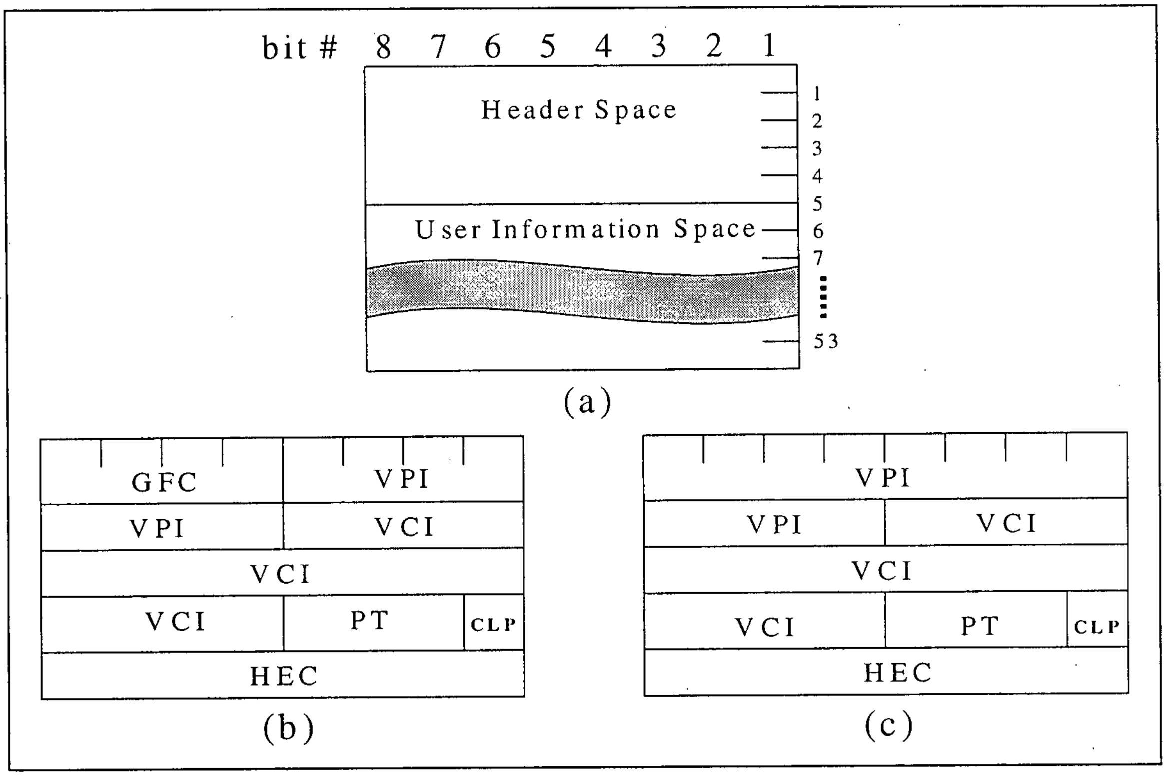

Figure 2-2: The ATM cell structure: (a) cell structure; (b) header structure at UNI; (c) header structure at NNI.

An ATM cell consists of 53 bytes, which are divided into 5 bytes of header and 48 bytes of payload space (Figure 2-2(a)). The main function of the cell header is to identify the travel path in the network. This function is designated as virtual path identification (VPI) and virtual channel identification (VCI) in Figure 2-2(b, c). A virtual path implies a bundle of virtual channels that share a common path. Other cell header functions include identifying payload types (PT), indicating cell loss priority (CLP), and providing header error control (HEC). At the UNI, the cell header additionally provides the generic flow control (GFC) function [24].

The GFC field is used to alleviate short-term overload conditions that may occur at the UNI (User Network Interface) by controlling the flow of traffic submitted to the network by users [24]. Since we have real time VBR traffic, GFC function is not applicable in this case.

CLP is a bit used for indicating whether the corresponding cell may be discarded during the time of network congestion. HEC is a CRC byte for the cell-header field and is used for sensing and correcting cell errors and delineating the cell header [24].

ATM networks [34] are able to support VBR sources. However, because of limited buffer capacities at the switches, when multiple connections transmit at peak rate simultaneously, they may cause congestion and buffer overflow. In order to avoid such congestion, traffic control is implemented in ATM networks [19] [34]. Traffic control in ATM networks mainly incorporates two functions: connection admission control (CAC) and usage parameter control (UPC).

CAC is implemented during the call setup procedure to ensure that the admission of a call will not jeopardize the QoS of the existing connections. At the time of call setup, the user requesting the call setup must use a signaling message to present the characteristics of the user traffic and the QoS required. CAC uses this information to decide whether to grant or to refuse the connection, determines the traffic characteristics for UPC, and allocates network resources. If the connection is admitted, a service contract is established specifying the amount of resources reserved according to the source traffic descriptor and the required QoS. The service contract also specifies the traffic behavior to which the input bit stream should conform in order to achieve the desired QoS. The traffic and QoS characteristics can be renegotiated upon the user's request during the course of the call connection. The processes of the user establishing the call, the user negotiating with the network to specify the desired traffic and QoS characteristics, and the user renegotiating a service contract are not in the scope of this thesis.

UPC is performed during a connection's lifetime to monitor the input traffic. Its main purpose is to protect network resources from malicious as well as unintentional misbehavior that can affect the QoS of other connections by detecting violations of negotiated parameter values. Consequently, the UPC algorithm must be equipped with the capability to monitor illegal traffic conditions, discriminate between whether or not the actual traffic parameters exceed the specified range limits, and cope quickly with parameter usage violations. The usage parameter can be a part or all the traffic characteristic parameters used for CAC. In our thesis, the usage parameters of UPC are the same as the ones in CAC. For rt-VBR traffic, the key traffic descriptors for UPC are PCR (peak cell rate), MBS (maximum burst size) and SCR (sustainable cell rate) [3].

If UPC identifies a parameter usage violation, several methods can be applied. The simplest is to discard the cells in question and this is the one used in this thesis. Other methods that can be considered include indicating the violator cells and removing the connection that contains the violator cells.

ATM cells conforming to the UPC parameters are guaranteed delivery with the specified QoS, while nonconforming cells are subject to losses when congestion occurs. In this thesis, we assume that all the cells conforming to the UPC parameters are guaranteed delivery, i.e., guaranteed service is used. Nonconforming cells are tagged (setting the CLP bit to 1), and can be sent as best effort traffic, but the network provides no guarantees for their delivery as no network resources are reserved for these cells. They are discarded if the network is congested. Therefore, our analysis in this paper assumes the worst case scenario - all tagged cells are discarded during the transmission and therefore not available at the receiver.

2.4 Implementation of Usage Parameter Control and Cell Loss Priority Functions

The Usage Parameter Control (UPC) function is a mechanism for monitoring the traffic from a source to make sure that its parameters are within the limits set in the traffic contract.

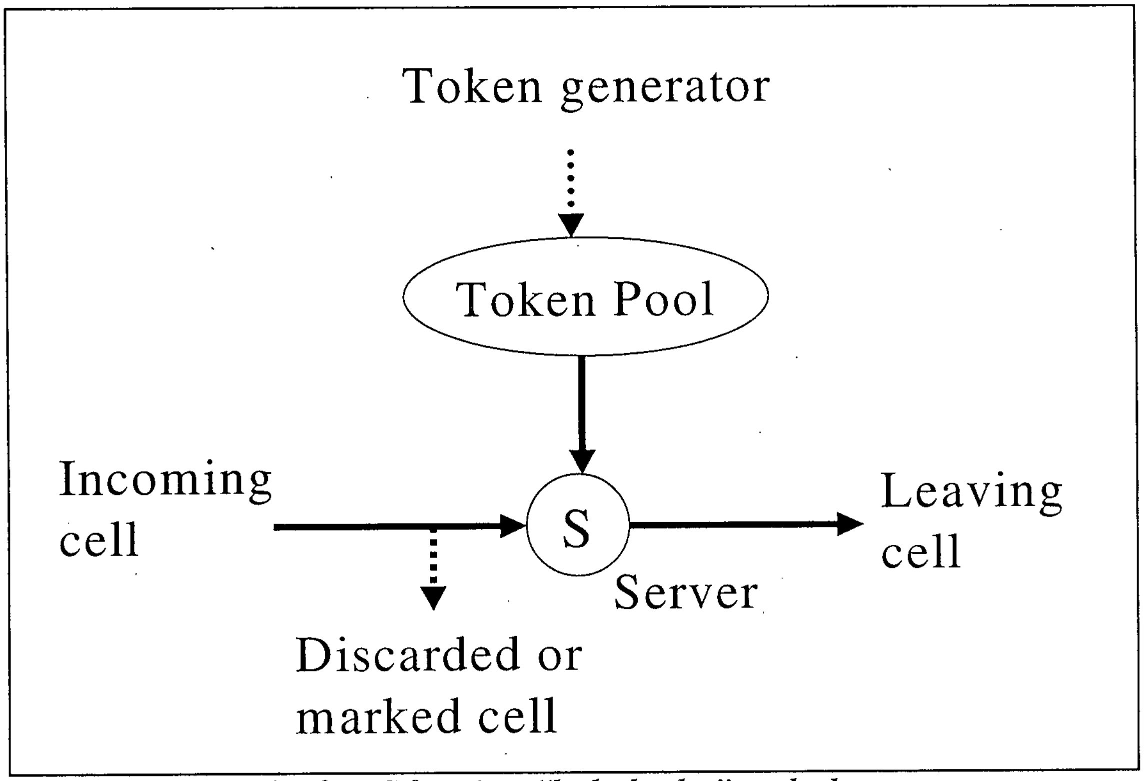

Figure 2-3: Example of UPC function: "leaky bucket" method

In our thesis, a simple and common UPC algorithm, the "leaky bucket" method, is used. The most basic form of it uses a token generator, which generates tokens periodically. The tokens go into a token pool, and whenever a cell is transmitted a token is used from the pool. By controlling the size of the pool and the token generation rate, various traffic parameters may be controlled (refer to Figure 2-3). In the thesis, the basic form of the "leaky bucket" method provides an effective UPC function for SCR and MBS. In other words, the maximum token pool level is equal to the maximum number of ATM cells can pass at once, which is equivalent to maximum burst size (MBS) . In the long run, the token generate rate is equal to the average number of ATM cells passed per time unit, which is equivalent to sustainable cell rate (SCR). In additional to pool size and token generation rate, an additional buffer is used to control the third parameter: PCR. Please refer to Figure 3-1 and the next chapter for details.

Since VBR services can widely vary in bit rates, at the moment when the various VBR services all manifest their maximum possible bit rates, the network can be severely congested. As a means of resolving such traffic congestion, the CLP function is used. That is, priority level to be used for cell loss (or cell discard) is recorded in the CLP field of each ATM cell employed for VBR services, and when congestion occurs, the cells with lower priority are discarded first. If the CLP bit indicates 1, then it represents a cell with a lower priority that can be abandoned.

2.5 ATM Adaptation Layer

The MPEG2 standard was developed by the Moving Pictures Expert Group (ISO/IEC JTC1/SC29/WG11), while ATM is being standardized by the ITU SG-XVIII and the ATM Forum. The ATM Adaptation Layer (AAL) is a special layer that provides an interface between the user and control data units and the service provided by the ATM layer, and is common to both the control and the user planes [20]. The ATM adaptation layer type 5 (AAL5) is an adaptation layer protocol for the ATM layer of broadband integrated services digital networks (BISDN's), which targets connection-oriented services. AAL5 was initially introduced as an efficient AAL for carrying data traffic. The main service it provides is detection of corrupted Convergence Sub-Layer PDUs (by means of CRC-32) [4], [40]. The ISO/IEC Standard Committee has adopted a packet format known as the Transport Stream (TS) for sending MPEG2 encoded data over communication networks [1]. In the ATM adaptation layer, AAL5 is selected by the ATM Forum to produce PDUs (protocol data units) from MPEG2 transport packets [40]. The reasons are

- The wide acceptance of AAL5 from both the computer and telecommunication industries [1],
- No extra hardware required [1],
- Effective error handling [8].

Therefore, AAL5 is used in this project for all segmentation and reassemble of data-grams and detection of transmission errors and dropped cells. AAL5 PDU format is shown as follows [41]:

< 64 K	0-47 bytes	16 bits	16 bits	32 bits
Data	Pad	Reserved	Length	CRC-32

where:

- pad so trailer always falls at end of ATM cell
- Length: size of PDU (data only)
- CRC-32 (detects missing or disordered cells)
- End-of-PDU bit in Type field of ATM header

Chapter 3 Proposed MPEG2-to-ATM Converter

This chapter first gives an overview of the proposed MPEG2-to-ATM converter. Then the structure diagram and operation flow are presented. Two key components: the data structure and break point assignment algorithm, are then described. The proposed data structure includes both high priority and low priority data structure. The design concepts behind them are explained. The two key components, standard_ratio and break point look up table, of the break point assignment algorithm are addressed. The description of the ATM-to-MPEG2 converter is also given.

3.1 Overview

The role of the MPEG2-to-ATM converter is to serve as a video data processor and protocol formatter between the MPEG2 video source and the user network interface (UNI) that connects the source to an ATM network. At the source, the MPEG2-to-ATM converter functions as an external post-coding source rate controller positioned between any MPEG2 source (archives, live programs, etc., which may be remotely located) and the UNI (see Figure 1-1).

The incoming MPEG2 data is partitioned into two bit-streams by splitting the discrete consine transform (DCT) coefficients into high priority (HP) and low priority (LP) partitions, containing low and high frequency coefficients, respectively, using a concept similar to the priority break point (PBP). The output high priority (HP) cells always conform to the UPC contract (the service contract established by the CAC process) policed by the UNI, and low priority (LP) cells may be non-conforming and are marked in the cell loss priority (CLP) bit in the cell header as candidates for discarding by the ATM network. ATM cell transmissions employ one virtual connection for both priority levels, for which a service contract is assumed to have been pre-established in our subsequent performance evaluation. The service contract mainly includes three parameters: MBS, SCR and PCR for rt-VBR traffic.

An ATM-to-MPEG2 converter at the receiver performs the reverse operation of assembling an MPEG2 stream from the received HP and LP ATM cells for playback. Although the proposed converter is originally intended for real-time broadcasting applications, it can also be applied to non-real time applications. Consequently, it is assumed that an up to 2 seconds delay of the video stream in the converter can be tolerated.

Here are some remarks about this MPEG2-to-ATM converter:

- The MPEG2-to-ATM converter is designed in such a way that the receiving end does not need to have the functionality of combining corresponding HP and LP data, in order to minimize the impact of the actual implementation. However, received video quality may be improved if this function is included at the receiver. Unlike many proposed methods, current MPEG2 decoders can decode the video streams from this MPEG2-to-ATM converter without the need for extra hardware or major software upgrade. The ability of detecting priority indicator and dropping the LP data is however required at the AAL layer at the receiving end (please refer to section 3.3 for details). The functionality of translating ATM cells back to AAL5 PDUs and then finally to an MPEG2 stream is assumed provided at the receiver end. In this case, this MPEG2-to-ATM converter is applied only at the source to shape the video traffic for the ATM channel. If the functionality of merging HP and LP data in the ATM-to-MPEG2 is not presented at the receiver end, the result will be equivalent to losing all the low priority data during the transmission.
- Only one ATM transmission channel is required, thus avoiding the channel synchronization issue at the receiver end.
- The algorithm is designed to be as general as possible. The idea can also be applied to any similar compression standards. There are many user-defined parameters introduced in the model. Different applications such as TV news, sports, and games, can use different set of parameters to achieve better picture performance.
- In the simulation model, several key components are also separated. Therefore, each component can be modified by using different algorithms for further research.

- In terms of received video quality comparison, a new method has been proposed to complement conventional average PSNR values methods. For each frame, the new method compares the frame's PSNR value against two reference values, which are determined based on subjective qualitative evaluation.

3.2 Structure and Operation Flows

Figure 3-1 shows the structure of the converter. The incoming bit-stream (up to 60 frames or 2 seconds) is stored in the elementary stream (ES) buffer, and examined by the Traffic Manager (TM) to determine the traffic load of current and future frames. Combining this information with the available resources (tokens) from a simulated leaky bucket, the TM then assigns a proper distribution between HP and LP ATM cells by determining an appropriate break point (BP). The DCT coefficients in the incoming VBR MPEG2 bit-stream are separated into HP and LP data according to the BP, converted into AAL5 PDUs and subsequently into HP and LP ATM cells for transmissions over the ATM network.

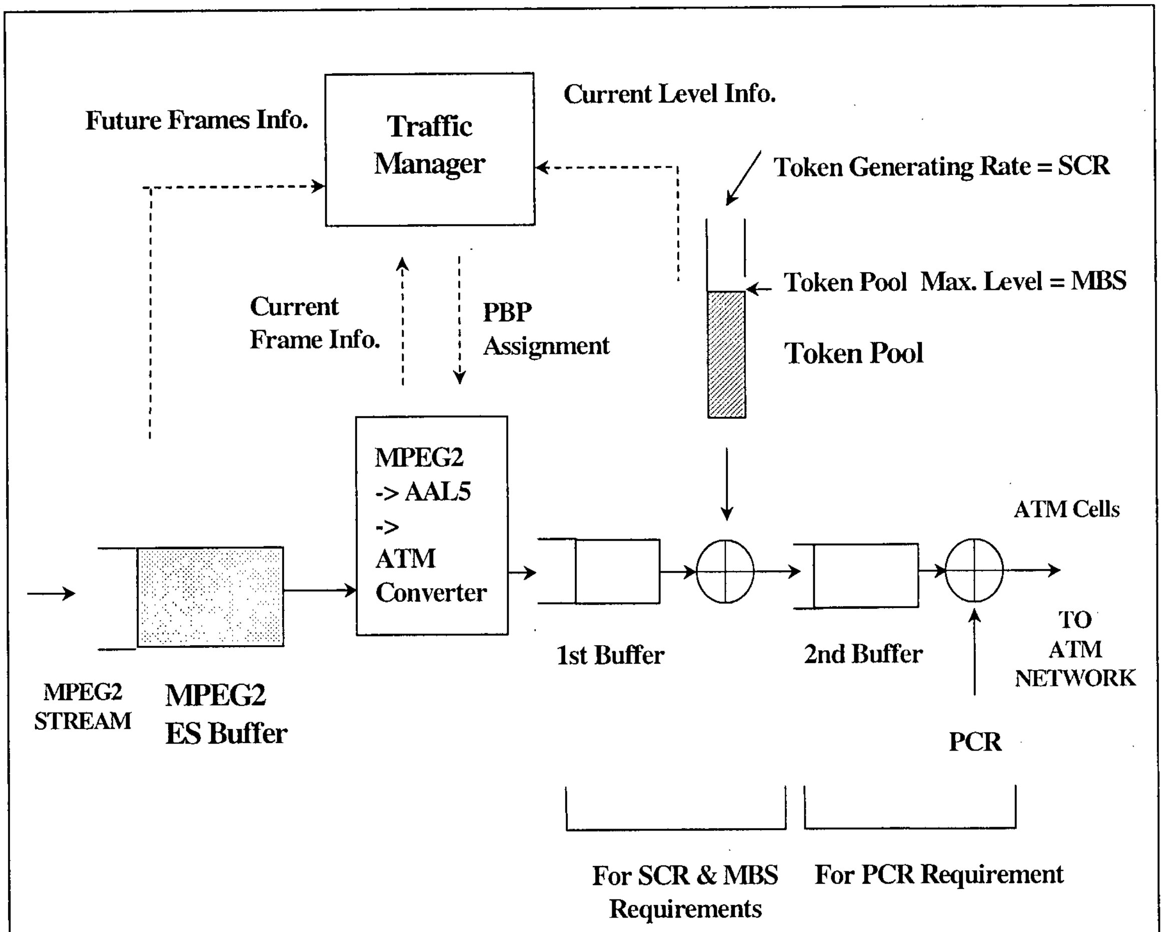

Figure 3-1: Structure of MPEG2-to-ATM Converter

The service contract for rt-VBR traffic includes three parameters for UPC: MBS, SCR and PCR. Using a concept similar to the PBP, data partition with dynamic BP settings is applied to the DCT coefficients in the incoming stream of VBR MPEG2 video data to yield a HP and an LP bit-stream, containing low and high frequency coefficients, respectively. The outgoing HP ATM cells satisfy the established ATM service contract.

In the simulated leaky bucket, the SCR and MBS requirements are met by setting the token generating rate to be SCR and the maximum token level to be MBS. The true SCR in the long run will be less than the SCR, if there is traffic shaping involved or token pool overflow. The idea behind the data structure design is to make sure all the high priority data can go trough the ATM networks without any discarding by the network police, UPC. Since we have no control of the real world implementation of UPC in the ATM networks, it is wise to leave some spare room. The PCR requirement is met by introducing another buffer (the 2nd buffer in the Figure 3-1) and the PCR token generator. According to the MPEG2 video standard, PBPs can be defined only on a slice-by-slice basis. However, the MPEG2-to-ATM Converter controls the source rate by changing the BP continuously.

In the ATM adaptation layer, AAL5 is selected by the ATM Forum to produce PDUs from MPEG2 transport packets [4], [40]. In this thesis, the AAL5 PDU format is used as the bridge between MPEG2 video data and ATM cells in the data structure. Since there are two different data priorities, two data structures are developed, one for high priority (HP) MPEG2 data and one for low priority (LP) data. Since MPEG2 video data is the focus in this thesis, only MPEG2 elementary streams (ES, video only) are used for testing purposes. From an error detection viewpoint, CRC32 code in each AAL5 PDU provides strong error detection capability [40].

Figure 3-2 presents the flow chart of the overall operation of the MPEG2-to-ATM converter. The key operations such as calculation of standard ratio (or standard_ratio), BP (break point) Look Up Table (or BP_look_up_table), assignment of HP and LP data, break point (BP) re-evaluation process, and others are addressed in the following sections of this chapter. The proposed data structure is described in the next section.

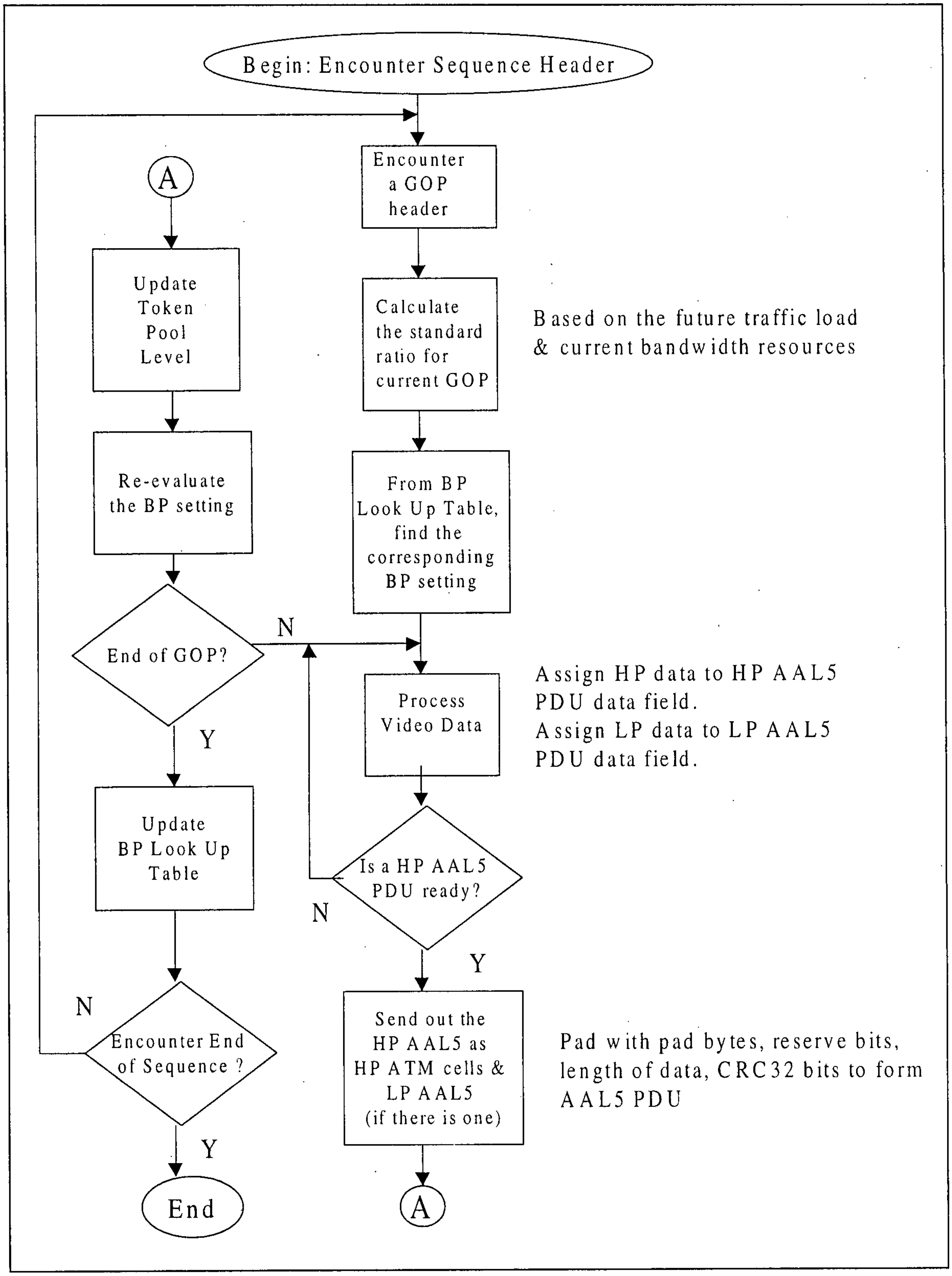

Figure 3-2: Overall Operations of the MPEG2-to-ATM Converter

3.3 Data Structure

3.3.1 High Priority Data Structure

The design of the data structure is intended to minimize the overall overhead. In other words, the target is to have the high priority data structure with the smallest overhead.

An end of block (EOB) code word is used to separate each piece of information. This piece of information is considered as the basic unit in the data structure. In the later section of this thesis, the term "piece" refers to this basic unit. Pieces may contain either header information or coefficient data information. For example, the first piece of the sequence contains the sequence header, a GOP header, a slice header, and coefficients, followed by the first EOB (end of block). The next piece generally contains only coefficient data. The BP (break point) in this scheme defines the number of non-zero encoded coefficients to be included in HP AAL5 PDU data. All headers, motion vectors and the first DCT coefficient (i.e., the DC coefficient) are always placed in the HP PDU without exception followed by the encoded DCT coefficients until the break point (BP) setting. All the high priority data continue to be padded together until the length is over 376 bytes. This number is chosen because an MPEG2 Transport Stream (TS) has a fixed size of 188 bytes, and the default length for the MPEG2 AAL5 PDU is 2 TS packets [4] [40]. This accumulated data then becomes the data payload in a HP AAL5 PDU, and is padded with extra bytes if needed to ensure that the PDU length is a multiple of 48 bytes, the payload size of an ATM cell. The remaining encoded coefficients, if any, are then placed in the LP AAL5 PDU data until the end-of-block (EOB).

The received MPEG2 streams size	29871360 bits
Required ATM cells	86541 ATM cells = 36693384 bits
Required AAL5 PDUs	13999 PDUs
No. of HP Flag (1 per PDU)	13999 bits
Size of overhead by the data structure	36693384 – 29871360 = 6822024 bits
% overhead introduced by the data structure	6822024/ 29871360 = 22.8%
% overhead introduced by the AAL5 and ATM cells	(6822024-13999) / 6822024 = 98.8 %

Table 3-1: Example of overhead calculation from a test stream

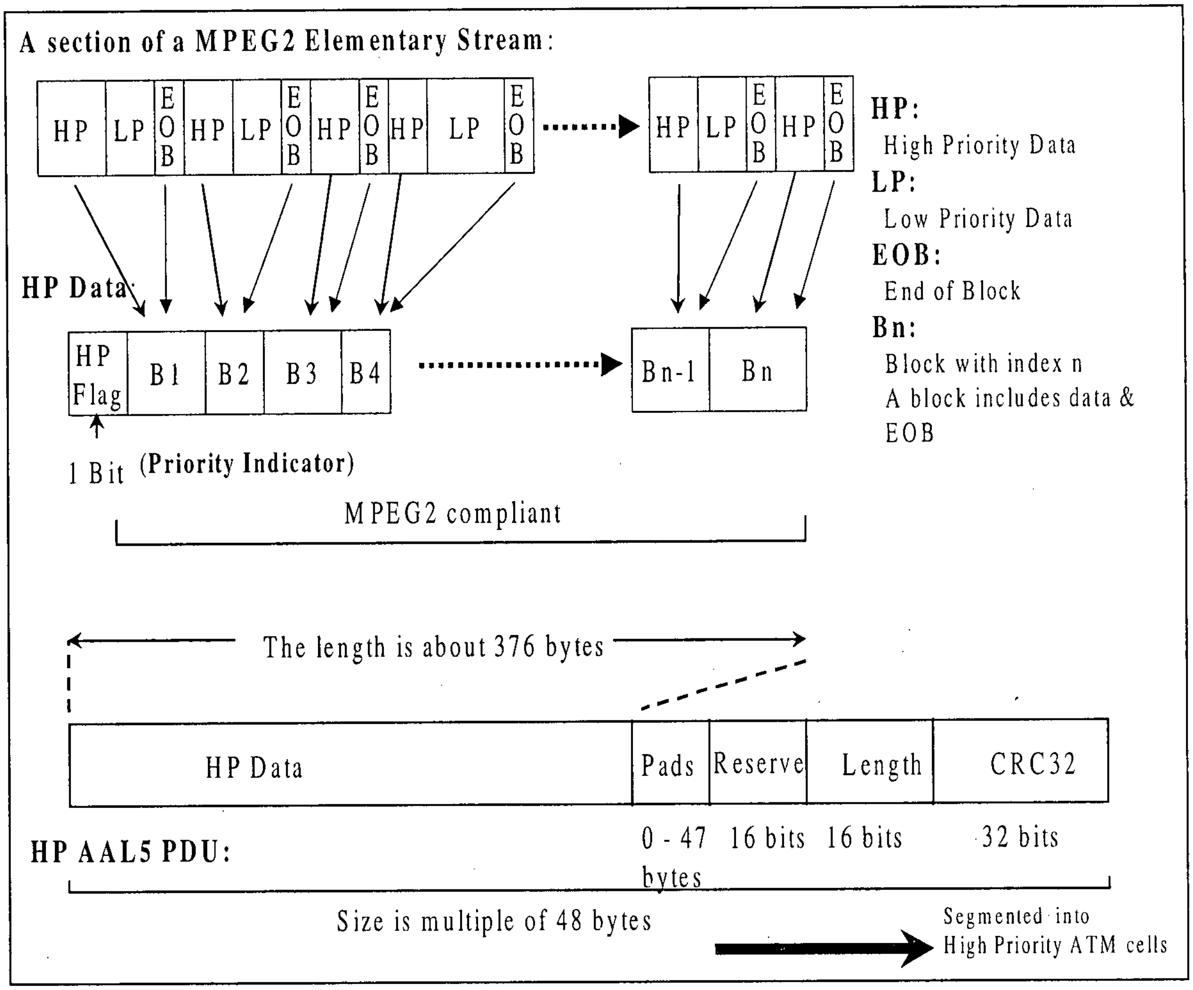

Figure 3-3: Data Structure for High Priority Data

One of the simulation results (see Table 3-1) shows that 22.8% overhead is introduced by the HP data structure; however, about 98.8% of this is contributed by the overhead of the AAL5 PDUs and ATM cells. This illustrates that the increase in overhead, caused by the HP data structure, is minimal (1.2% in the above example).

3.3.2 Low Priority Data Structure

LP data pieces containing frequency coefficients beyond the BP are packed into LP AAL5 PDUs, which are not considered as bandwidth consuming, because the CLP bits in the resulting ATM cells are marked for possible discarding by the network. The LP AAL5 PDU must provide enough information for the receiver to assemble the received LP data with the HP data into an MPEG2 compatible stream. Therefore, the LP data structure has more overhead information than that of HP data.

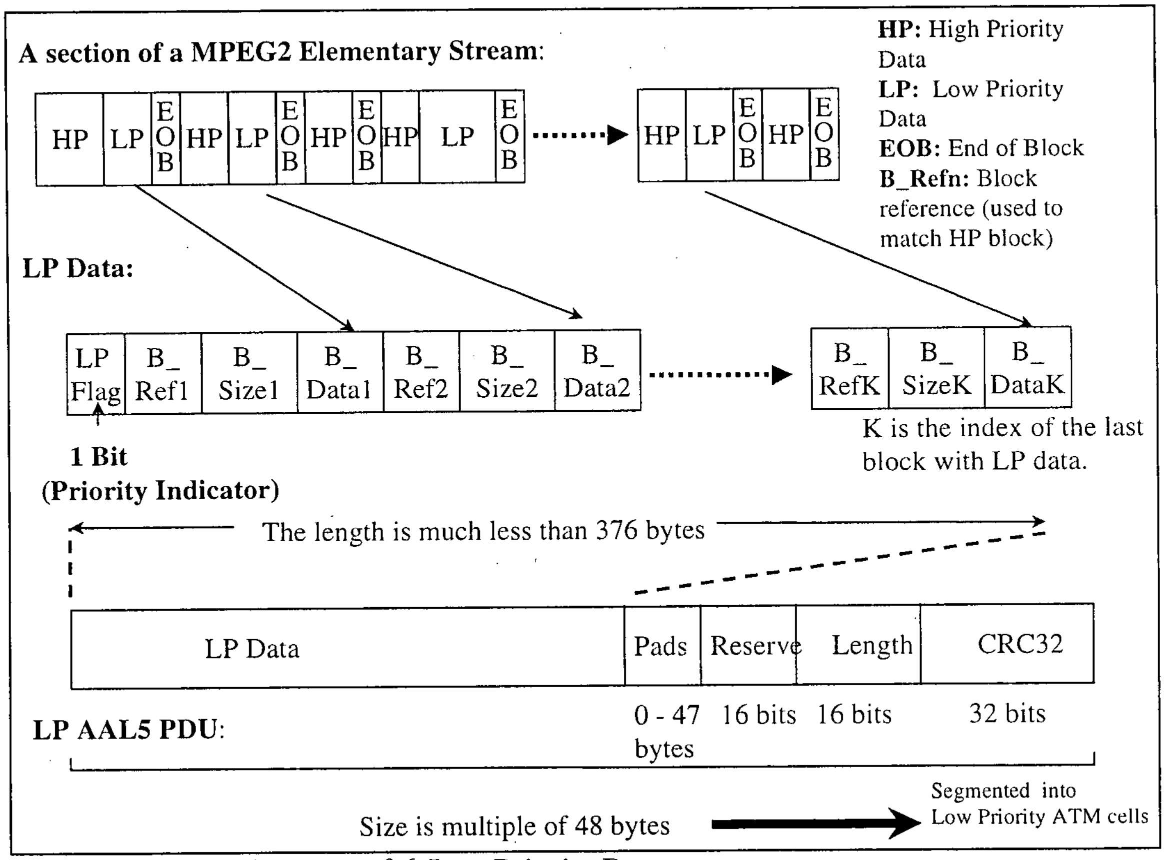

Figure 3-4: Data Structure for Low Priority Data

The LP data structure is shown in Figure 3-4. The first bit indicates that the current AAL5 PDU is LP. The block reference (B_ref) indicates which HP block the following LP data block is associated with. The next two fields are block size (B_size) and block data (B_data). The length of B_size is 1 byte. In other words, the maximum number of bits for an LP B_Data is 254 bits. If LP B_Data has more than 254 bits, only those coefficients within the first 254 bits are sent. These 254 bits are also the more important ones.

In the case where the service contract parameters, PCR, SCR, and MBS are far below the actual traffic requirements, the LP data structure has almost zero overhead because of this 254 bits limit. For example, the service contract parameters settings in test no. 1-10, 2-10 and 3-10 result in almost zero overhead in LP data structure (please refer to chapter 5 for details). However, in the case where the service contract parameters are only slightly below the actual traffic requirements, the LP data structure results in up to 119% of overhead in our simulation results. In this particular example, 387970 HP ATM cells are output from the MPEG2-to-ATM converter. At the same time, 77382 LP ATM cells are introduced by the LP data structure. If there is enough channel bandwidth for transmitting 100% of the video data, only 35386 additional HP ATM cells are required to transmit the rest of the information. Therefore (77382-35386) / 35386 = 119% overhead is calculated. For the same video stream, only 4% of the overhead is introduced by the LP data structure, when the service contract parameters are much more restricted. Please refer to chapter 5 for further discussion.

From simulation results, it is found that with Ibp, Pbp, and Bbp less than 16, LP B_Data starts to exceed 254 bits. Thus low break points may cause the 254 bits to be insufficient. If a service contract is reasonably set up, this case should be very rare. As in HP PDUs, LP data are likewise padded so that the PDU length is a multiple of 48 bytes.

3.4 ATM-to-MPEG2 Converter

At the transmitter end, both HP and LP AAL5 PDUs are segmented into ATM cells for transmission. The HP AAL5 PDU is sent out first, followed by the LP AAL5 PDU, if any. At the receiver, an ATM-to-MPEG2 converter is responsible for converting these ATM cells back to the original MPEG2 stream by properly merging the HP and LP data, using the information given by the EOB, B_ref, and B_size. The flow chart of the ATM-to-MPEG2 converter is presented in Figure 3-5.

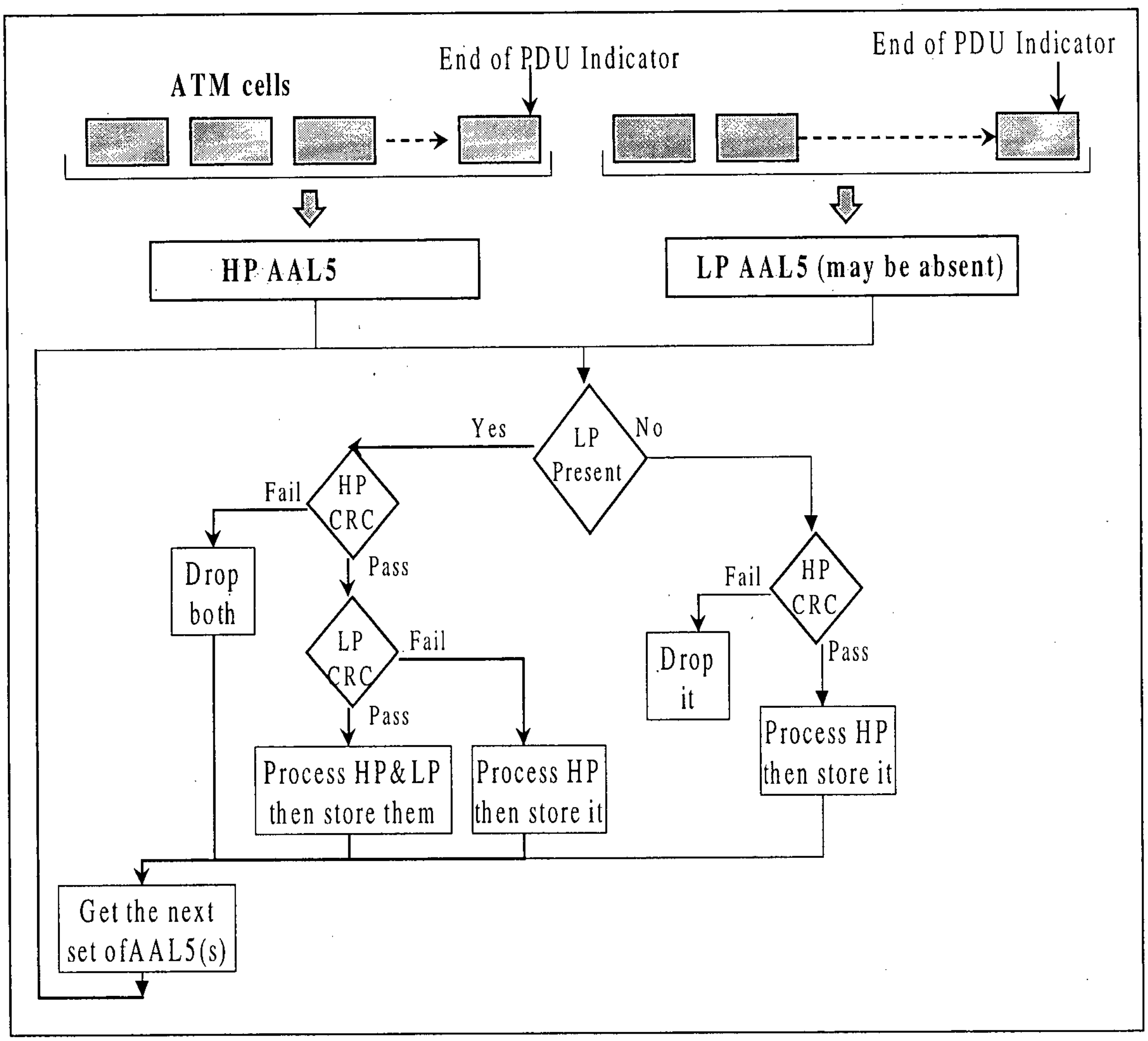

Figure 3-5: Flow Chart of ATM-to-MPEG2 Converter

3.5 User Defined Parameters and Global Variables

This section presents some key parameters and variable used in the MPEG2-to-ATM converter.

- **Token_pool_level**: The token pool level in the simulated Leaky Bucket at any given time.
- **Ibp**: Current break point setting of I-frame (intra-frame). I-frames contain only intra-coded blocks. In fact, Ibp is used to split intra-coded blocks. If there is rapid change in the video scene, intra-coded blocks will also be used in P- and B- frames. In general, I-frames contain more important information than both B- and P- frames while P-frames are more important than B-frame.
- **Pbp**: Current break point setting of non-intra-coded blocks in P-frames (predicted frames).
- **Bbp**: Current break point setting of non-intra-coded blocks in B-frames (bi-directional frames).
- **Bbp_ratio**: The ratio to calculate Bbp. Bbp = Ibp * Bbp_ratio. In general, Bbp_ratio is less or equal to 1. The default value is 1, but it can also be user defined.
- **Pbp_ratio**: The ratio to calculate Pbp. Pbp = Ibp * Pbp_ratio. In general, Pbp_ratio is less than 1 and Pbp_ratio is greater or equal to Bpb_ratio. The default value is 0.5, but it can also be user defined.
- **BP_look_up_table**: A break point look up table which stores 20 Ibp values for Standard_ratios from 0 to 1. The default table is shown in Table 3-3, but the table entries can also be user defined.
- **SCR**: Sustainable cell rate or token generated rate.
- **MBS**: Maximum Burst Size or the maximum level of token pool in the simulated Leaky Bucket.
- **PCR**: Peak cell rate.

- **Min_Ibp**: The minimum break point of intra-coded blocks the MPEG2-to-ATM converter will use. This is used to maintain a certain minimum level of video quality. In fact, this also limits the capability of the MPEG2-to-ATM converter. If the outgoing HP data with this minimum break point setting cannot satisfy the ATM service contract, then the MPEG2-to-ATM conversion would fail because the output cannot conform to the UPC requirements. The default value is 16, but it can also be user defined.
- **Min_token_pool_level**: If the token pool level reaches this Min_token_pool_level, Ibp will be set to Min_Ibp, and then Bbp and Pbp will consequently be updated. This is used to prevent the token pool from running out of tokens. The default value is 25, but it can also be user defined.
- **GOP1_CR**: the average ATM cell rate for the first GOP.
- **GOPn_CR**: the average ATM cell rate for the nth GOP.
- **Standard_ratio**: In our break point assignment algorithm, standard_ratio is the parameter used to prevent the token pool from running out of tokens and thus provide smooth traffic shaping. It is defined in the next section.

The use of these different parameters will be stated, clarified, and demonstrated in the next section.

3.6 Standard Ratio

3.6.1 Basic Concepts

In our BP (break point) assignment algorithm, which will be described in the next section, a key variable called the ***standard_ratio*** is used to prevent the token pool from running out of tokens and thus provide smooth traffic shaping.

The basic form of standard_ratio is defined as:

$$\text{Standard\_ratio} = \frac{\text{(tokens available)}}{\text{(tokens required for perfect transmission)}}$$

$$= \frac{\text{(current Token Pool level + tokens generated up to the worst case)}}{\text{(additional tokens required up to the worst case + tokens generated up to the worst case)}} \qquad \textbf{(3-1)}$$

3.6.2 An Example

Here is an example to demonstrate how the parameter Standard_ratio is used to prevent the token pool from running out of tokens and also provide smooth traffic shaping. Figure 3-6 shows the virtual token pool level over serveral GOP periods for this example. The virtual token pool is used only for evaluation purposes and is allowed to overflow and underflow.

Assume that the virtual token pool at point A has X tokens. The MPEG2-to-ATM converter monitors the next 2 seconds of frames, which includes the next 6 GOPs. Because the MPEG2 video stream has piece-wise variable bit rate (constant bit rate within a GOP), the virtual token pool level will look like Figure 3.6. Point B turns out to have the worst token pool level given by Y, a negative number. The period between A and B is called Time_A_B. Over this period, (Time_A_B*SCR) tokens are generated, but another (-Y) tokens would have been needed if the token pool level was to stay above zero. Therefore, the standard_ratio is equal to (X + Time_A_B * SCR) / [(X-Y) + Time_A_B * SCR].

In order to get the smoothest picture quality and maximum throughput, the distribution of high priority and low priority data needs to ensure that the token pool level is slightly above zero when reaching the worst point. In other words, at the end of GOP n+3 (point B), the token pool level should be slightly above zero line instead of the level Y, which is below zero. This is achieved by selecting a proper BP, as a function of the calculated standard_ratio. The standard_ratio is re-calculated every GOP and fixed within the GOP. The standard_ratio can also be seen as the number of HP ATM cells divided by total ATM cells originally required without using priority BPs. For example, suppose the calculated standard_ratio is 0.6, and there is a high priority PDU containing 12 ATM cells and a low priority PDU containing 17 ATM cells (due to the overhead introduced by the low priority PDU data structure). If there is no data partitioning involved, there should be 12/0.6 = 20 ATM cells in this time period to send all the video data. Therefore, BPs must selected so that only 12 HP ATM cells are generated from the 20 ATM cells originally required.

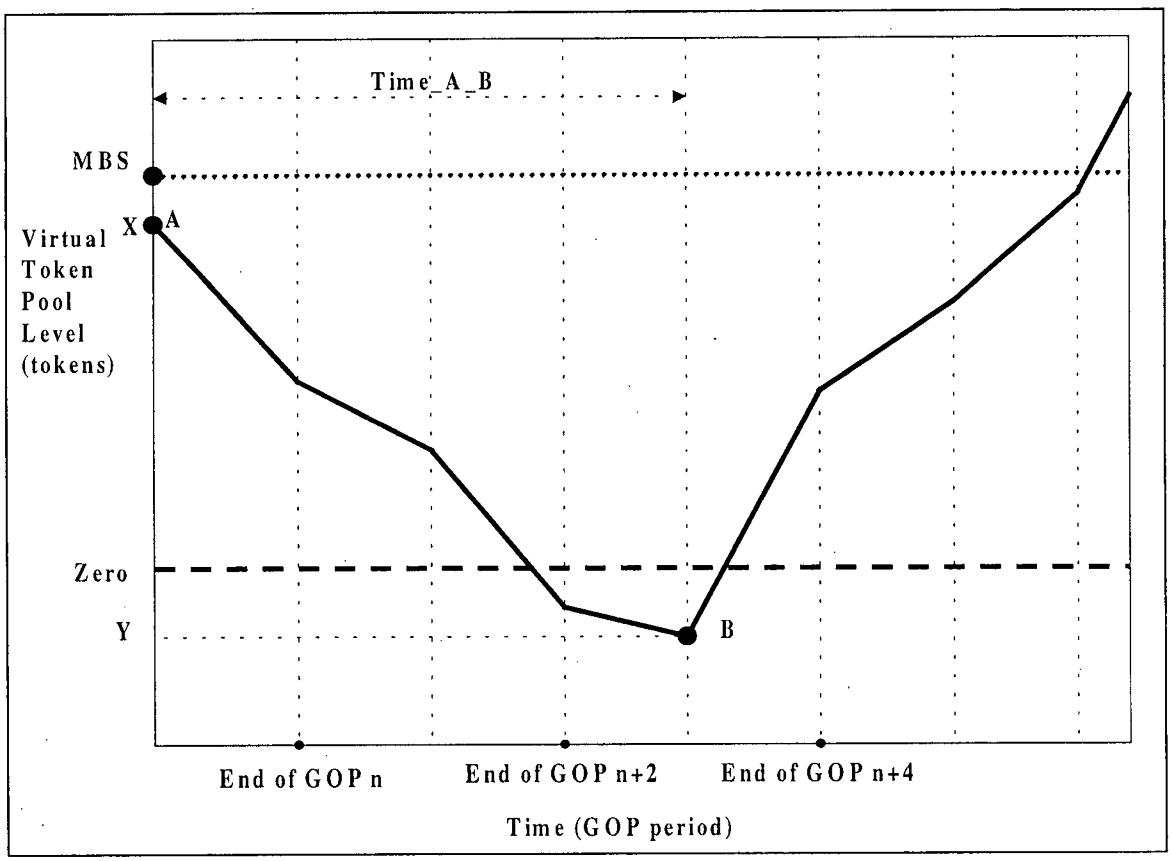

Figure 3-6: An example to demonstrate the use of Standard_ratio

3.6.3 Calculation of Standard_ratio

In the MEPG2-to-ATM converter, the calculation of standard_ratio is more complicated than the basic form given by equation (3-1). There are other issues to be considered. These considerations are discussed in the following. Then the complete set of steps for the standard_ratio calculation is given.

First, if the standard_ratio is larger than 1, the resource is enough for all the video data. Then the standard_ratio is set to 1. This is equal to the all 64 coefficients-passed case where break points of I-frames, B-frames and P-frames are set to 64.

Another case where break points of these three types of pictures are set to 64 (or standard_ratio is set to 1) is when the virtual token pool's level at the end of any GOP before the minimum token pool level occurs is larger than the MBS. In other words, the average cell rate for the GOP is less than the token generating rate.

If neither of these two cases apply, the initial standard_ratio may still need some adjustment. If by following calculated standard_ratio, the resulting token pool level at the end of the first GOP is over MBS, then it is wasting resources (because of token pool overflow). Therefore, the standard_ratio will be adjusted to be (Tokens generated in this GOP – MBS + Current Token Pool level) / (Tokens required for perfect transmission in this GOP).

Next, peak cell rate restriction must be met as well. If the average cell rate for the first GOP (GOP1_CR) is larger than PCR, the standard_ratio needs an adjustment. If the the standard_ratio is greater than PCR/GOP1_CR, then it will be set to PCR/GOP1_CR in order to meet the PCR parameter in the service contract.

In summary, the standard_ratio is calculated using the following steps:

1. /* analysis the future frames */

Calculate the lowest virtual token pool level case (referred as the worst case) in the next 2 seconds. Collect the following values:

- Total GOP number in the next 2 seconds.
- Token pool level at the end of the first GOP.
- The average cell rate for the first GOP: GOP1_CR.

2. /* token pool overflow case */

If the token pool level at the end of any GOP >= MBS before the worst case GOP then

stanard_ratio =1;

jump to step 6;

3. /* initial standard_ratio calculation */

Standard_ratio =

$$\frac{\text{(current Token Pool level + tokens generated up to the worst case)}}{\text{(additional tokens required up to the worst case + tokens generated up to the worst case)}}$$

= (X + Time_A_B * SCR) / [(X-Y) + Time_A_B * SCR] (please refer to Figure 3-6).

4. /* the maximum value for standard_ratio is 1 */

If standard_ratio > 1 then

standard_ratio = 1;

jump to step 6;

5. /* check for token pool overflow when using the stanard_ratio */

Following the calculated standard_ratio, estimate the token pool level at the end of first GOP.

If the estimated token pool level > MBS then

Standard_ratio = (tokens generated in the GOP – MBS + current token pool level) / (tokens required for perfect transmission in this GOP)

6. /* check for the peak cell rate parameter */

Following the calculated standard_ratio, estimate the average cell rate for the first GOP.

If the estimated average cell rate > PCR then

Standard_ratio = PCR / GOP1_CR;

Several methods (for example, by using the ratio of average cell rate of different GOPs, by using linear or non-linear math equations for standard_ratio re-calculation process) have been evaluated. The above method gives the best simulation results combining with the break points assignment algorithm so far.

3.7 Break Point Look Up Table

The break point look up table is another important component in our BP (break point) assignment algorithm. In the algorithm, the break points should be assigned so that the output number of HP ATM cells follows the calculated standard_ratio. The break point values for different standard_ratio ranges are stored in this table.

The default break point look up table (BP_Look_Up_Table) is based on simulation results of a main sequence containing four MPEG2 clips. By using different break point settings, the number of high priority ATM cells required are calculated. The results are shown as follows:

The ratio between Ibp, Ppb, and Bpb is 1 : 1 : 0.5.

Ibp	64	58	52	48	40	34	28	22	16	10
Standard Ratio	0.90	0.87	0.82	0.77	0.72	0.66	0.60	0.52	0.44	0.36

Table 3-2: I-frame break points setting vs. Standard_ratio

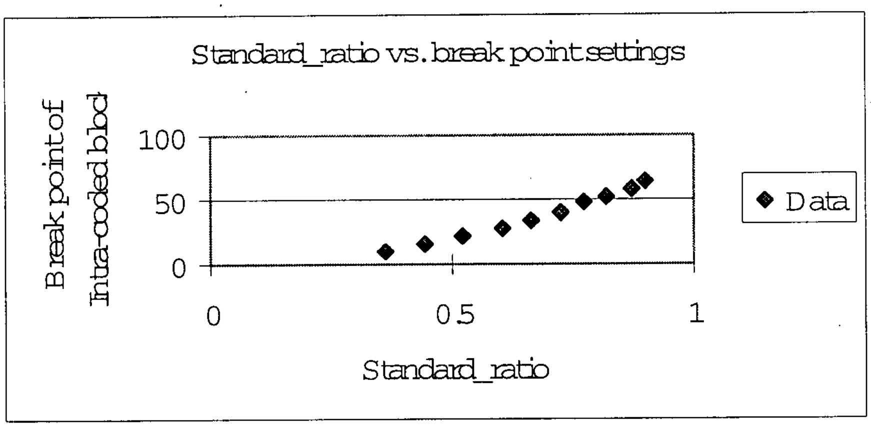

Figure 3-7: Standard_ratio vs. Break point settings

From Figure 3.7, it is found that the relationship between the Ibp setting and the standard_ratio is quite linear. Therefore, the break point look up table is set as follows:

Standard_ratio range	Ibp Setting (use B_ratio and P_ratio to calculate the Bpb and Ppb settings)
0.95 < standard_ratio <= 1.00	60
0.90 < standard_ratio <= 0.95	58
0.85 < standard_ratio <= 0.90	54
0.80 < standard_ratio <= 0.85	50
0.75 < standard_ratio <= 0.80	46
0.70 < standard_ratio <= 0.75	38
0.65 < standard_ratio <= 0.70	32
0.60 < standard_ratio <= 0.65	28
0.55 < standard_ratio <= 0.60	24
0.50 < standard_ratio <= 0.55	20
0.45 < standard_ratio <= 0.50	16
0.40 < standard_ratio <= 0.45	16
0.35 < standard_ratio <= 0.40	16
0.30 < standard_ratio <= 0.35	16
0.25 < standard_ratio <= 0.30	16
0.20 < standard_ratio <= 0.25	16
0.15 < standard_ratio <= 0.20	16
0.10 < standard_ratio <= 0.15	16
0.05 < standard_ratio <= 0.10	16
0.00 < standard_ratio <= 0.05	16

Table 3-3: Ibp setting in the BP_look_up_table of simulation

Please note that the initial table setting is not completely linear. There is a minimum value for an I-frame break point. The reason is that from evaluation results, video streams with an I-frame break point equal to 16 result in barely acceptable quality. The ratio between I-frame break point setting, B-frame break point setting, and P-frame break point setting is based on several evaluations for this particular test stream. This is one possible method to define the break points. The methods of defining break points dynamically can also be another research area. Several methods (for example, using different equations) have been tested. It is found that a lookup table is the most suitable one so far. The current table is separated into 20 ranges linearly: different I-frame break points are assigned for each range. The look up table is updated dynamically to follow the traffic trend in order to achieve the performance. Within the period of a GOP, standard_ratio is fixed, but BP setting can be changed for each AAL5 PDU. At the end of the GOP, the entry of the look up table for that standard_ratio is updated to (sum of I-frame break point values in the GOP) / (number of AAL5 PDUs in the GOP). The updating mechanism can also be another research topic.

3.8 Break Points Assignment Algorithm

After introducing the stanard_ratio and the break point look up table, the next is the break points assignment algorithm itself. In the MPEG2 video standard, PBP can only be changed on a slice basis.

The break points should be assigned such that the output number of HP ATM cells follows the calculated standard_ratio. First, the value is assigned from the BP_Look_Up_Table. There is a default break point look up table in the converter, but it can be user defined. This break point look up table is updated at the end of each GOP as the streams passes through. Since a fixed break point setting does not guarantee the number of HP ATM cells follows the standard_ratio, break point settings are changed continuously during the GOP period. Once a high priority PDU is ready to be sent out, the BPs setting is evaluated again.

Figure 3-8 presents the flow chart of the operation. The flow chart gives an overall picture about the operations. However, some detail operations are not included in the flow chart. For this reason, the pseudo code is also given. In addition to the parameters and variables defined in section 3.4, there are some internal variables that need to be defined.

- **Reduction_step**: This is the reduction step size during the break point reduction process. This is a variable.
- **Increment_step**: This is the increment step size during the break point increment process. This is a variable.
- **Max_reduction**: The maximum reduction step. The default value is 3, but it can also be user defined.
- **Max_increament**: The minimum increment step. The default value is 3, but it can also be user defined.
- **AAL5_PDU_Period**: The time period of the AAL5 PDU. It is not a fixed value.
- **Orig_cell_no**: Original ATM cells required during the AAL5 PDU period.
- **HP_atm_no**: The HP ATM cells sent out during the AAL5 PDU period.

- **Ibp_total**: Sum of Ibp values so far in the current GOP.
- **Ibp_count**: The number of times that Ibp has been examined in the current GOP.
- **Real_ratio**: It is the actual ratio of (No. of HP ATM cells sent out in the AAL5 PDU period / Original ATM cells required in the AAL5 PDU).

Basically, if the real ratio is within a limit of the calculated standard_ratio, no break point changes are made. However, if the difference is over the limit, break points reduction or increment is performed. The reduction or increment step is increased until its maximum value (Max_reduction or Max_increment). If the token pool's level is over MBS, it will be set back to MBS. In this case, which is referred as token pool overflow, excess tokens are lost. Please refer to the pseudo code starting on page 49 for a step by step description of this operation.

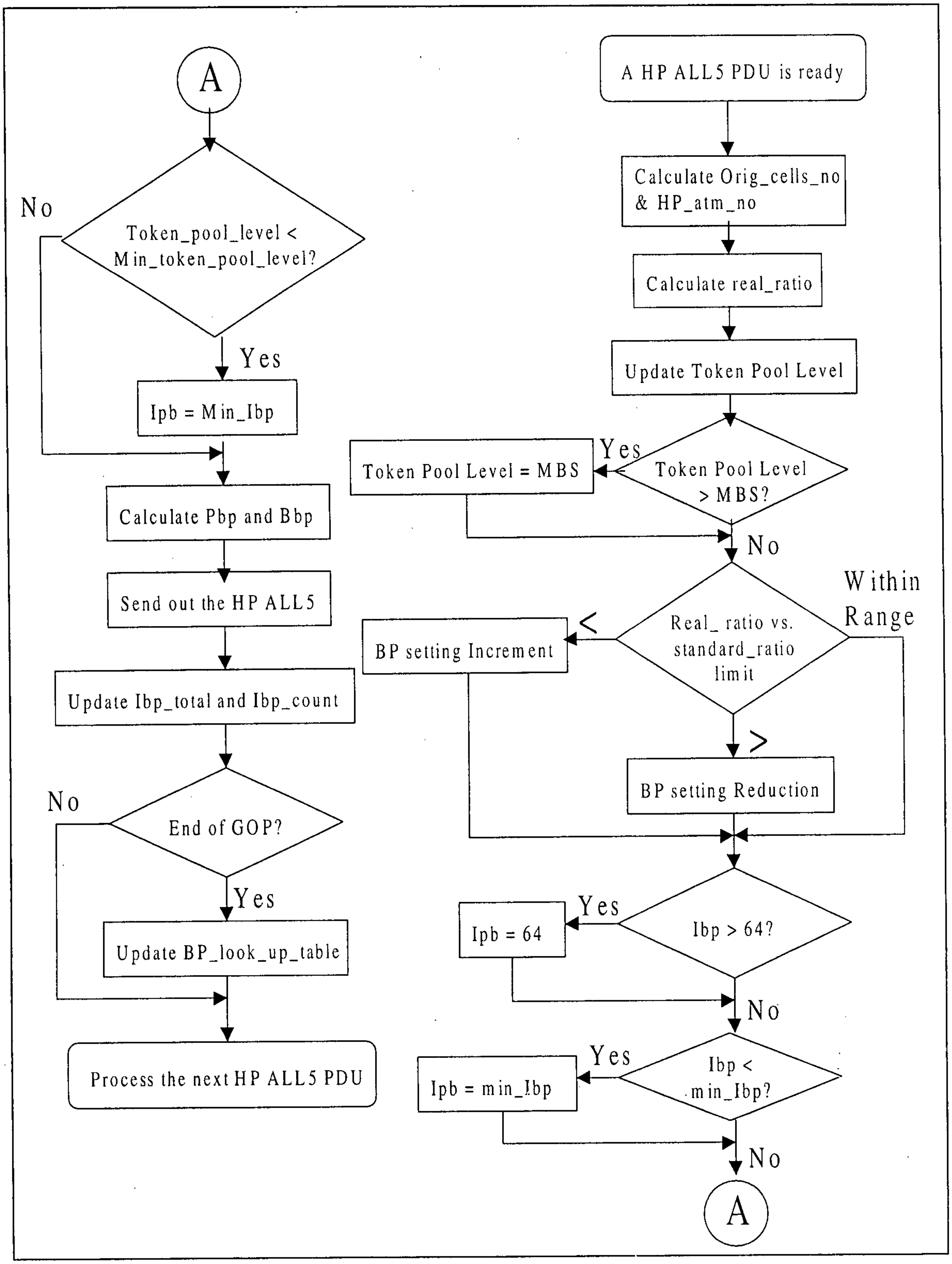

Figure 3-8: Break Point Assignment Algorithm Flow Chart

The following is the detailed pseudo code for the operations.

```
When a HP AAL5 PDU is ready
      Calculate the Orig_cells_no;
      Calculate the HP_atm_no;
      /* Calcuate the Real_ratio */
      real_ratio =HP_atm_no / Orig_cell_no;
      /* Update the Token Pool Level (Token_pool_level) */
      Token_Pool_Level = Token_Pool_Level – HP_atm_no + SCR * AAL5_PDU_period;

      /* examine the token pool level for overflood case */
      if (token_pool_level > MBS){
             token_pool_level = MBS;
            Ibp = Pbp = Bbp = 64;
      }

      /* check for real_ratio vs. stanard_ratio limit, in this case 5% is used */
      if ( real_ratio / stanard_ratio > 1.05)){
            /* BPs  reduction */
            Ibp = Ibp-Reduction_step;
            Reduction_step ++;
            if (Reduction_step > Max_Reduction)
                  Reduction_step = Max_Reduction;
             Increment_step = 1;
      }
      if ( real_ratio / stanard_ratio < 0.95){
             /* BPs Increment */
            Ibp = Ibp + Increment_step;
            Increment_step ++;
            if (Increment_step > Max_Increment)
                  Increment_step = Max_Increment;
            Reduction_step = 1;
      }
      /* Ibp refinement */
      if (Ibp > 64){
            Ibp = 64;
      }
      if (Ibp < Min_Ibp){
            /* the minimum Ibp */
            Ibp = Min_Ibp;
      }
      if (token_pool_level < Min_token_pool_level){
            /* almost hit the token pool bottom */
            Ibp = Min_Ibp;
      }
```

```
/* Use Ibp setting to calculate Pbp and Bpb settings */
Pbp = Ibp * P_ratio;
Bbp = Ibp * B_ratio;

Send the HP ALL5 PDU out;

/* update the value for Ipb_total and Ibp_count */
Ibp_total = Ibp_total + Ibp;
Ibp_count ++;

If the end of GOP is encountered{
      /* update the BP_look_up_table */
      The corresponding Ibp setting of current standard_ratio in the
      BP_look_up_table is set to Ibp_total / Ibp_count;
      /* reset the Ibp_total ad Ibp_count */
      Ibp_total = 0;
      Ibp_count = 0;
}
Continue to process the next HP ALL5 PDU.
```

Chapter 4 Simulation Model

The simulation model is used to simulate the actual implementation. The performance results of simulation are used to evaluate the proposed MPEG2-to-ATM converter. Figure 4-1 shows the block diagram for the simulation model. It includes five modules: "Traffic Manager (TM)", "Simulated Leaky Bucket", "MPEG2 Video-AAL5 PDUs-ATM Cells Converter (MAAC)", "ATM Channel", "ATM cell-AAL5 PDUs-MPEG2 Video Converter (AAMC)" and "Observation Center". Since detailed design descriptions have been given in the previous chapter, the five modules are only discussed briefly in this chapter. The block-based design of the simulation model provides scalability and portability for future enhancement and research. In addition to the module descriptions, video streams, assumptions in the simulation, and evaluation cases are also discussed.

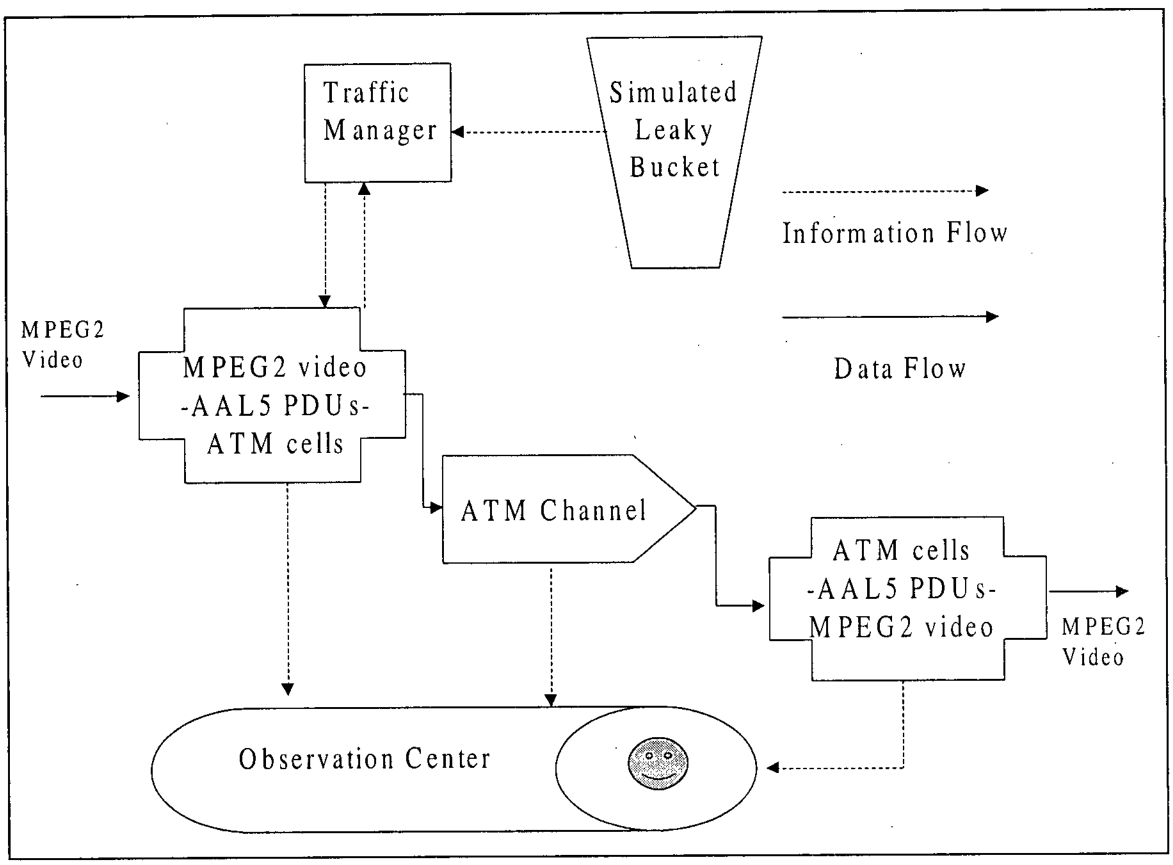

Figure 4-1: Block Diagram of the Simulation Model

4.1 MPEG2 Video Streams

The main traffic sequences used throughout this paper are composed of 4 pieces of short samples with various scene contents, including both scenes with quick background changes (football and cheerleader) and scenes without background change (garden and train & calendar). They are in I P P P P P P P P P I format. The total length of the 4 sequences is around 300 frames and duration is about 10 seconds. With GOP size set to 10, there are 30 GOPs in the combined sequence.

In order to have a sufficient sequence length, these 300 frames or 30 GOPs have been separated and then composed into three long video streams. The first two are randomly mixed with the GOPs in the original streams, and the third one is merely the original video stream repeated four times. The data rate diagrams (Figures 4-2, 4-3 and 4-4) show that these streams have piece-wise variable bit rates from 75K to 250K bits per second (or 300 to 1100 ATM cells per second.) The data rate distribution histograms of the sequences are shown in Figures 4-5 4-6, and 4-7. We also examined several other high quality video streams (with data rate up to 12M bits per second), which yield similar results. For this reason, only these three combined 75K - 250K bit rate streams are used in this research.

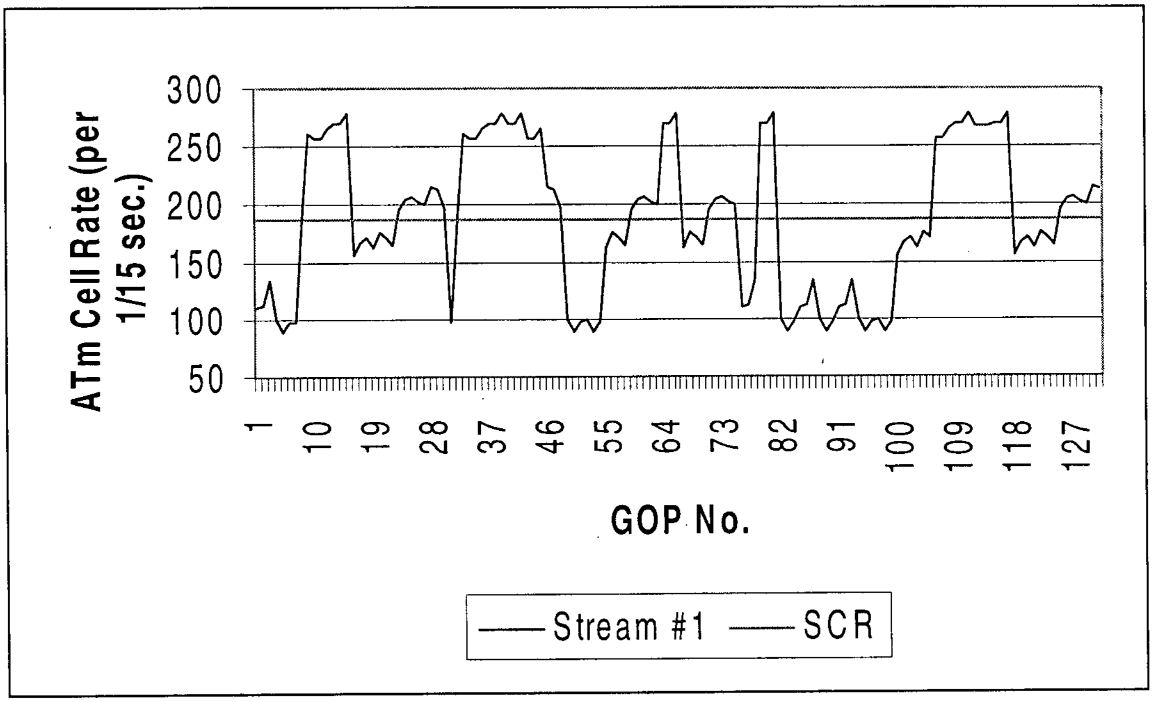

Figure 4-2: Data Rate Profile of Stream #1

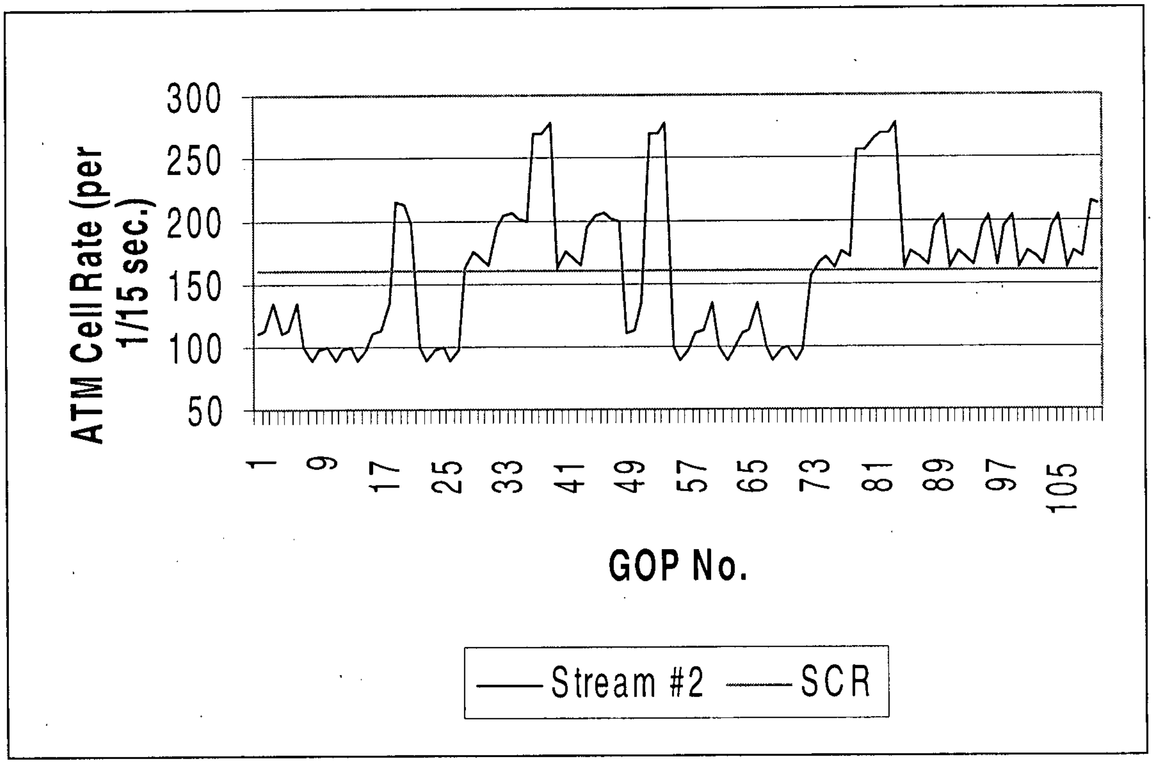

Figure 4-3: Data Rate Profile of Stream #2

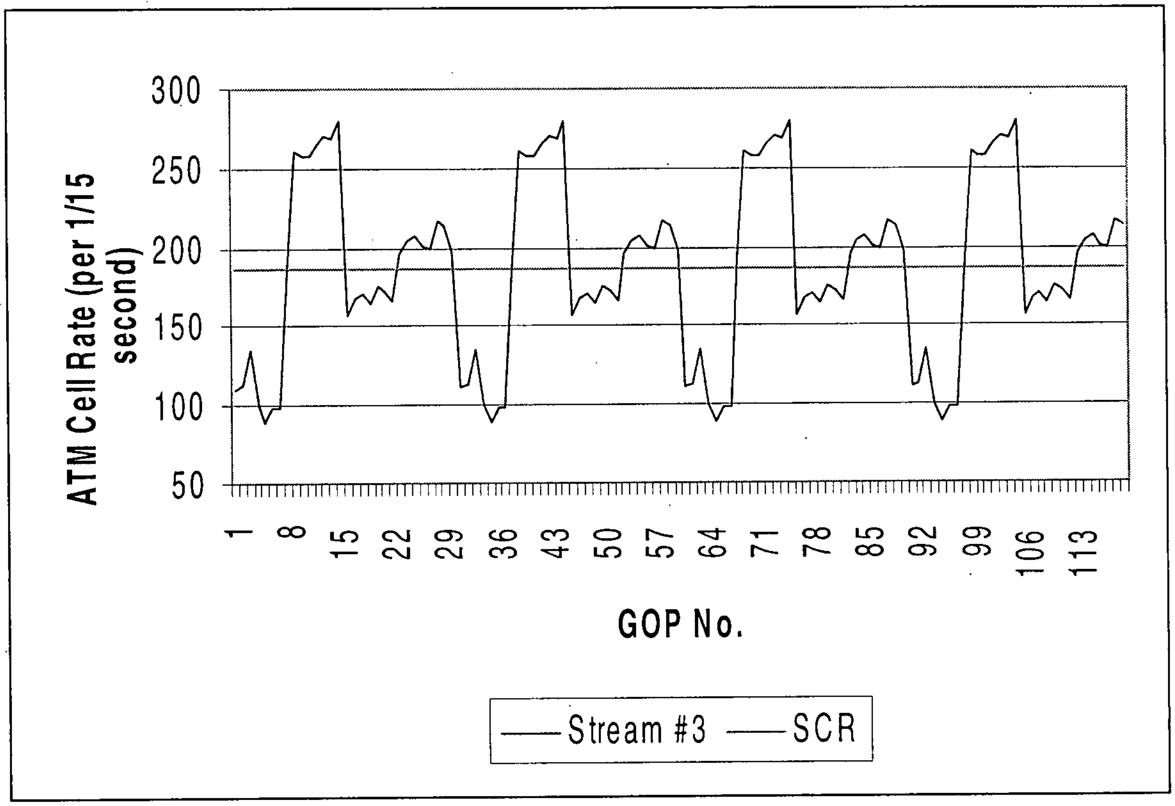

Figure 4-4: Data Rate Profile of Stream #3

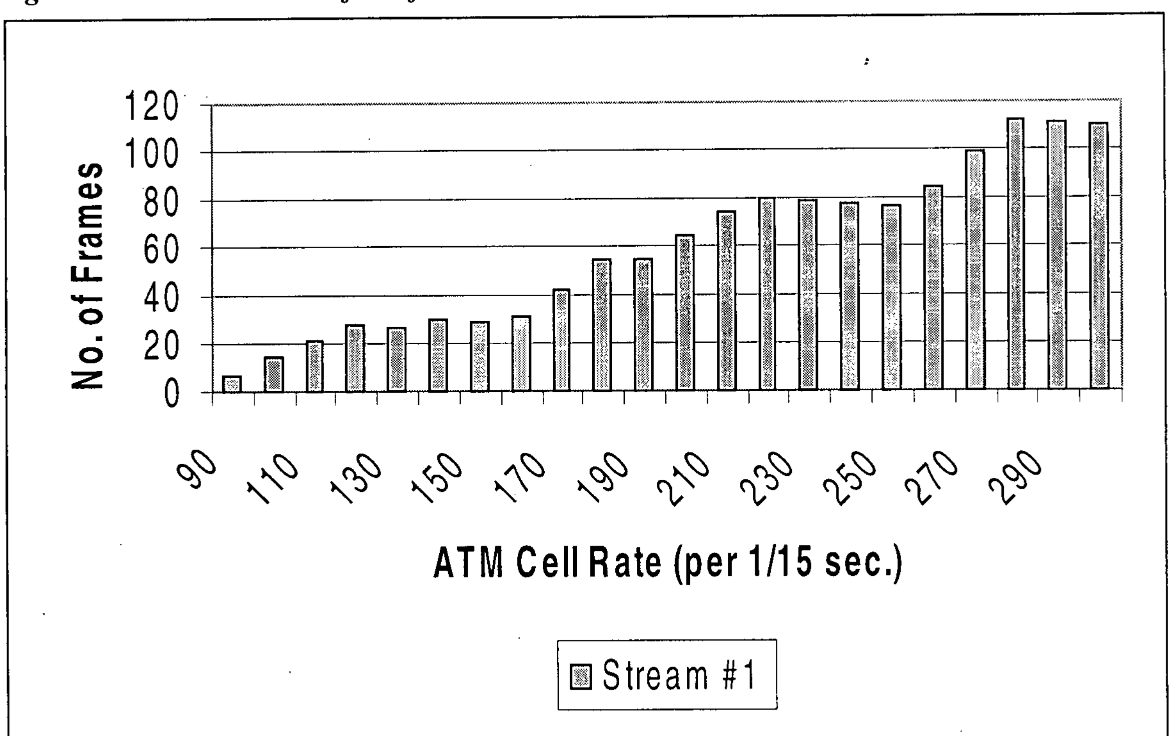

Figure 4-5: Date Rate Distribution Histogram of Stream #1

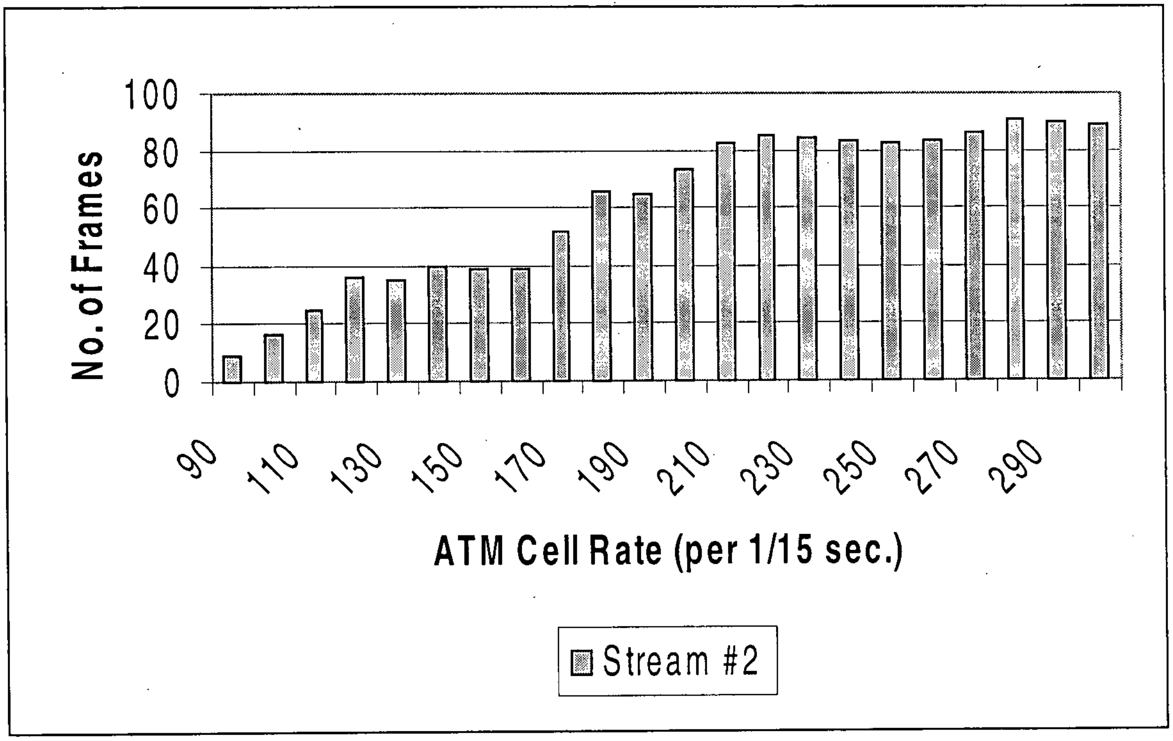

Figure 4-6: Data Rate Distribution Histogram of Stream #2

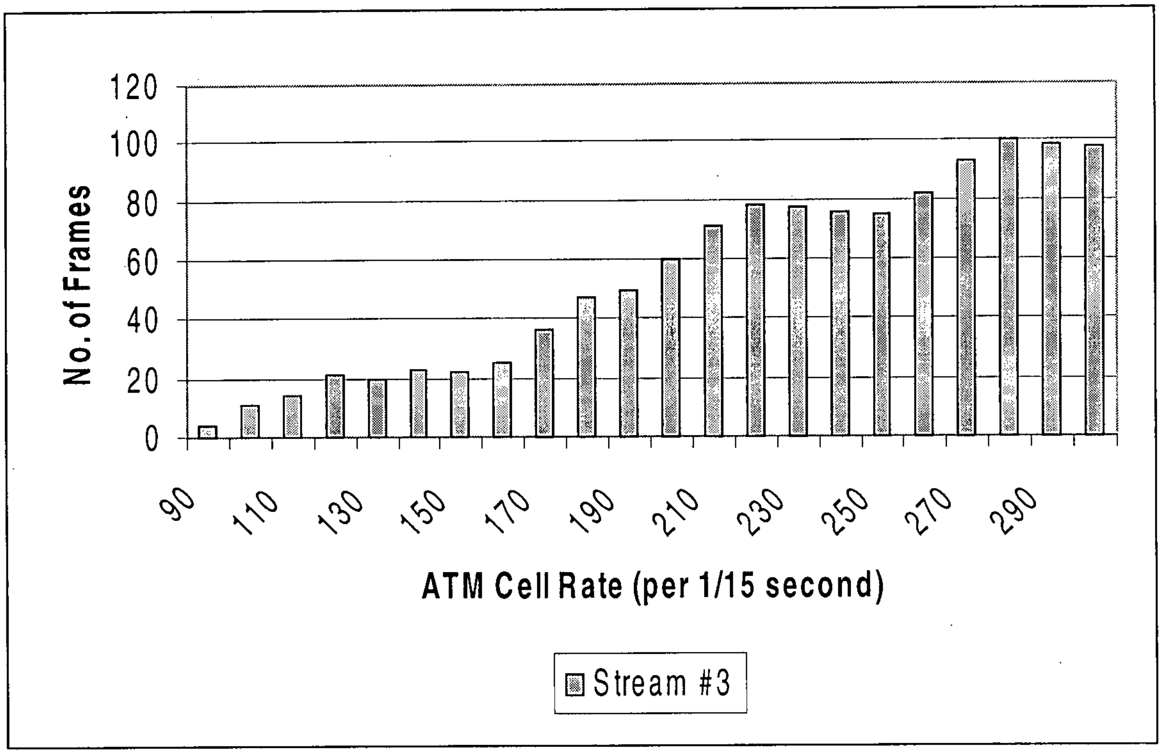

Figure 4-7: Data Rate Distribution Histogram of Stream #3

4.2 Traffic Manager Module

The job of the TM is to assign proper break points to the MAAC module. It can be seen as the brain of the simulator. The decision of break points are based on the current break points settings (in the MAAC module) and near future traffic load information (from the Simulated Leaky Bucket module). The break points are to shape the current incoming MPEG2 video and output high priority ATM cells which satisfy the established ATM service contract. The break points assignment algorithm has been detailed in chapter 3. Different algorithm can be adapted in this TM module for further research.

4.3 Simulated Leaky Bucket Module

The task of Simulated Leaky Bucket module is to simulate the current ATM network traffic load (or the ATM traffic parameters in UPC) and have the information ready for the Traffic Manager module. It functions like an ATM network traffic load reporter. In this thesis, a simple leaky bucket model (Figure 3-1) is used, as described in section 3.2.

4.4 MPEG2 Video - AAL5 PDUs - ATM Cells Converter Module

The MAAC module accepts incoming VBR MPEG2 bit-stream, separates the bit-stream into HP and LP data based on the break point settings provided by the TM, converts them into AAL5 PDUs and subsequently into HP and LP ATM cells for transmissions over the ATM networks. Please refer to section 3.2 and 3.3 for detail data structure definition and operation flow.

4.5 ATM Cells - AAL5 PDUs - MPEG2 Video Converter Module

The AAMC module is responsible for converting received ATM cells back to an MPEG2 video stream by merging the HP and corresponding LP data. The flow chart is presented in Figure 3.5. Please also refer to section 3.3 Data Structure as reference. The simulations assume that all the LP ATM cells have been dropped during transmissions over the ATM network. As a result, no merging is performed. Since the HP data are MPEG2 conformant, the data requires no further processing for decoding by a standard MPEG decoder.

4.6 ATM Channel Module

In our ATM Channel module, only high priority ATM cells consume bandwidth resources. The UPC (usage parameters control) implementation at the channel input is similar to the simulated leaky bucket model except without the second buffer and second token generator with token generation rate equal to PCR (please refer to Figure 3-1). We assume that the service contract of the ATM virtual channel has been established with reasonable values of SCR, PCR and MBS. SCR and MBS requirements are fulfilled by setting the token generating rate to SCR and maximum token pool level to be MBS. A token is consumed when a HP cell is transmitted. If the token pool level is below zero, the current HP cell will be turned into a LP cell. The PCR requirement is fulfilled by calculating the accumulated ATM cells used in the current GOP period. If the number of tokens used is more than the product of PCR and current GOP period (in other words, maximum number of tokens allowed in the current GOP period), the rest of the ATM cells in the GOP will be discarded.

The performance of the MPEG2-to-ATM converter is evaluated by simulating the transmission of ATM cells through an ATM network such that low priority cells are delivered to the destination according to a specified probability known as "LP_ATM_Pass_prob". The high priority ATM cell has a cell loss probability known as "HP_ATM_Loss_prob". For simplicity, in our simulations, both LP_ATM_Pass_prob and HP_ATM_loss_prob are equal to zero. This is equivalent to all LP ATM cells being dropped and no losses for HP ATM cells. A reconstruction buffer at the receiver converts the cells back to MPEG2 video for both quantitative and qualitative video quality evaluations. Playing back the received video through a hardware MPEG2 decoder enables the qualitative evaluation.

To facilitate the quantitative evaluations, the following assumptions have been made:

1. To enable calculation of the average PSNR (peak signal to noise ratio) and each individual frame's PSNR, GOP headers and Picture headers must be available at the receiver. Therefore, regardless of whether the cell carrying these header information are received or not, the information is made available to the receiver. This ensures that frame-by-frame PSNR comparison between the receiver and the video source is possible. If an error occurs in the transmission, although the video coefficients in the affected cell are lost, we are still able to identify the correspond frame at the receiver.
2. The effects of errors on the video play-back are as the follows:
   - If an AAL5 PDU containing a GOP header encounters an error, the whole GOP will be dropped if the GOP has different parameters from the previous GOP. Otherwise, only the first picture is dropped.
   - If an AAL5 PDU containing a picture header encounters an error, the picture is dropped.
   - If an AAL5 PDU containing a slice header encounters an error, only the slice is dropped.
   - If an AAL5 PDU containing only coefficient data encounters an error, all the coefficient data are dropped.
3. The play-back buffer at the receiver is large enough to handle the delay variations caused by the ATM network. Therefore, the delay variation issue is not considered in this thesis. A constant delay is assumed in the channel model.
4. The ATM UPC does not employ smart cell-dropping [2] which uses the end-of-packet (EOP) indicator in the ATM cell header to enable it to drop the remaining cells belonging to an incomplete packet [2].

4.7 Observation Center module

The Observation Center module acts as a data collector. The data includes the number of HP ATM cells transmitted, the number of HP AAL5 PDU produced, the number of LP ATM cells produced, the number of LP AAL PDU produced, the levels of Token Pool, the values of standard_ratio, the values of Break Point settings and many other parameters. The collected data is used for further analysis and performance evaluation.

4.8 Alternative MPEG2-to-ATM Conversion Methods for Comparisons

The performance of a number of VBR MPEG2 video streams is evaluated. The three key UPC parameters: SCR, PCR, and MBS are set lower than those calculated for a given video stream. Based on these values, the performances of three methods of VBR MPEG2 cell transmissions over an ATM network are compared. The first two methods do not employ the MPEG2 to ATM converter as described above.

Method one does not employ any control for partitioning the video stream based on the content. The incoming VBR MPEG2 bit-stream, with the BPs (break points) of the I, P, and B frames all equal to 64, is simply converted into ATM cells which are passed through UPC policing for transmission over the ATM network. There is no processing to separate video coefficients into different priority levels. All ATM cells violating the service contract are dropped. This is referred as the "uncontrolled" case. Video headers can be discarded as well, and the loss of these headers renders the subsequent video coefficients useless and could result in significant received signal quality degradation. However, for frame by frame comparison purpose, in these cases the video headers are assumed to be available at the receiver, but the corresponding coefficient data will be dropped in order to simulate the effect of losing the video headers.

The second method employs "fixed control" to partition the DCT coefficients at some predetermined PBP to eliminate the LP coefficients. An example of this case is 484848, where the PBPs for I, P and B frames are 48, 48 and 48 respectively. All the headers and HP coefficients are sent in ATM cells through UPC policing to the ATM network. Non-conforming cells are dropped by the ATM network. This is based on the argument that only the more important coefficients should be sent to save bandwidth and achieve smooth video quality degradation. However, the fixed PBP does not guarantee UPC conformance for the resulting HP coefficients, and therefore suffers from the same problem of degraded received video quality due to video header losses, as in the uncontrolled case. In addition to the degrading video quality, bandwidth usage also cannot be optimized. For example, some of the assigned bandwidth will be unused and wasted, when the bandwidth for a particular period allows all the coefficients (i.e., 646464) to be transmitted.

The third method uses our MPEG2-to-ATM converter to partition the video data using dynamically assigned BPs. LP cells are dropped by the network. It automatically ensures that cells carrying headers and HP coefficients conform to the UPC contract and pass through the network intact.

In each case, after converting the received ATM cells into an MPEG2 stream, the average peak signal-to-noise ratio (PSNR) for the entire video sequence and each individual frame are calculated and compared.

The PSNR is defined as:

$$\text{PSNR} = 10 * \log (255/\text{MSE}) \qquad \textbf{(4-1)}$$

where

$$\text{MSE} = \sum_{i=1}^{W} \sum_{j=1}^{H} [(\text{original } (i,j) - \text{reconstructed } (i,j)]^2 \qquad \textbf{(4-2)}$$

In (4-2), H and W are the height and width of the frame. Original (i,j) is the original pixel value of the frame at location (i, j). Reconstructed (i,j) is the new pixel value of the frame at the corresponding location.

Chapter 5 Performance Evaluation

We evaluate the performance of the proposed MPEG2toATM converter using the three VBR MPEG2 video streams described in section 4.9. In this chapter, the evaluation test cases are described first. Then the simulation results are presented in terms of the LP data overhead, preset vs. effective ATM traffic parameters, average PSNR comparisons, channel efficiency, and frame by frame PSNR evaluations.

5.1 Evaluation Test Cases

Each stream is first analyzed by a computer routine to calculate the ATM traffic parameters required to pass 100% of the video data by setting I-frame Break Point setting (Ibp), P-frame Break Point setting (Pbp) and B-frame Break Point setting (Bbp) to 64 (this is referred as the 646464 case). According to the proposed HP data structure the routine outputs the average cell rate for each GOP in the video stream. Thus, the date rate profile (Figures 4-1 to 4-3) for each stream is plotted. The overall average ATM cell rate, maximum burst size and maximum cell rate are also calculated. Then the 484848 case (by setting Ibp = Pbp = Bbp = 48) is used as a starting point for performance evaluation. The video stream with only the coefficients with index number less or equal to 48 is passed into the same computer routine to calculate average ATM cell rate, maximum burst size and maximum cell rate. The results are listed in Table 5-1 and used as the references of determining ATM service contract parameters: SCR, PCR and MBS. For definition of ATM traffic parameters, please refer to chapter 3.

Stream No.	646464			484848		
	SCR	PCR	MBS	SCR	PCR	MBS
1	187	279	31091	171	244	29813
2	161	279	23281	149	244	28593
3	186	279	12380	171	244	19281

Table 5-1: Contract Parameter Requirements for testing video streams (SCR and PCR in no. of ATM cells per 1/15 s; MBS in no. of ATM cells)

By using the ATM traffic parameters of the 484848 case as the baseline, 10 different test cases are developed with more restrictive UPC parameters for each video stream. In the first 5 cases, MBS setting varies from 90% down to 50% of the baseline value. In the last 5 cases, the SCR setting varies from 95% down to 75% of the baseline value. The test cases for testing video streams are shown in Table 5-2 to Table 5-4, respectively.

Test No.	SCR	PCR	MBS
1-1	171	250	26832 (90%)
1-2	171	250	23850 (80%)
1-3	171	250	20869 (70%)
1-4	171	250	17888 (60%)
1-5	171	250	14907 (50%)
1-6	162 (95%)	250	29813
1-7	154 (90%)	250	29813
1-8	145 (85%)	250	29813
1-9	137 (80%)	250	29813
1-10	128 (75%)	250	29813

Table 5-2: Parameters of Test Cases for Stream #1

Test No.	SCR	PCR	MBS
2-1	149	250	25734 (90%)
2-2	149	250	22874 (80%)
2-3	149	250	20015 (70%)
2-4	149	250	17156 (60%)
2-5	149	250	14297 (50%)
2-6	142 (95%)	250	28593
2-7	134 (90%)	250	28593
2-8	127 (85%)	250	28593
2-9	119 (80%)	250	28593
2-10	112 (75%)	250	28593

Table 5-3: Parameters of Test Cases for Stream #2

Test No.	SCR	PCR	MBS
3-1	171	250	17353 (90%)
3-2	171	250	14625 (80%)
3-3	171	250	13497 (70%)
3-4	171	250	11569 (60%)
3-5	171	250	9641 (50%)
3-6	162 (95%)	250	19281
3-7	154 (90%)	250	19281
3-8	145 (85%)	250	19281
3-9	137 (80%)	250	19281
3-10	128 (75%)	250	19281

Table 5-4: Parameters of Test Cases for Stream #3

For each case, three methods to transmit the data through the ATM network, as described in section 4.9, are compared. In the following presentation, the " uncontrolled" method uses the raw MPEG2 bit-stream with 64, 64, 46 for I-, P-, and B- frame BPs, the "fixed controlled" method reduces these BPs to 48, 48, 48, and "MPEG2-to-ATM" or "converter" refers to using the proposed converter with variable BPs.

5.2 Overhead Introduced By LP Data Structure

Table 5-5 to Table 5-7 show the number of LP ATM cells introduced by the MPEG2-to-ATM converter in each test case. Figures 5-1 to 5-3 are the graphic representation of the original number of ATM cells required to transmit all the low priority coefficients data when they are not differentiated with high priority data, versus the new number of ATM cells required for transmitting these low priority data in the proposed LP AAL5 PDU format. In simulation results of test no. 3-1, with the restricted service contract, 387970 HP ATM cells are output by the MPEG2-to-ATM converter. At the same time, 77382 LP ATM cells are introduced by following the LP data structure. If there were enough channel bandwidth for 100% transmission of the video data, only 35386 extra HP ATM cells would have been required to transmit the rest of the information. The overhead introduced by the LP data structure is therefore calculated as (77382 – 35386) / 35386 = 119%. This is an example of a test case with the lightly restricted service contract parameters situation.

The graphs show that there is a roughly constant number of LP ATM cells required when the number of original ATM cells is below a certain level. Once the number of original ATM cells is over this level, the new number of ATM cells required increases rapidly and converges to the original requirement. This is because of the 254-bit limit for AC coefficients in each block in the data structure (please refer to section 3.3 for data structure). Test no. 1-10 and test no. 2-10 both have 0% overhead, and test no. 3-10 has 4% overhead. Their service contract parameter settings are heavily restricted. This means a lot of extra HP ATM cells are required to have 100% video data transmission and about the same number of LP ATM cells are produced by the MPEG2-to-ATM converter.

In this thesis, the simulation model assume that no smart dropping is implemented [2] in the UPC Police function. With the number of LP ATM cells introduced by LP data structure, it is found that most of the LP AAL5 PDU cannot be reassembled at the receiver perfectly. One ATM cell of an ALL5 PDU is dropped, the whole PDU is useless. For simplicity, all the ATM LP packets are assumed dropped during the transmission in this thesis. Therefore, no ATM network resource is consumed by LP ATM cells.

Test No.	HP ATM cells transmitted	Extra ATM cells required, Without LP data structure.	LP ATM cells introduced.
1-1	432940	38755	80674 (208%)
1-2	430220	41475	79653 (192%)
1-3	426758	44937	80891 (180%)
1-4	421555	50140	84139 (168%)
1-5	415240	56455	87134 (154%)
1-6	418863	52832	92315 (175%)
1-7	399462	72233	102054 (141%)
1-8	377251	94444	112611 (119%)
1-9	358008	113687	122952 (108%)
1-10	336199	135496	135510 (100%)
Total ATM cells required for 100% transmission is 471695 ATM cells			

Table 5-5: No. of LP ATM Cells Introduced for Stream #1

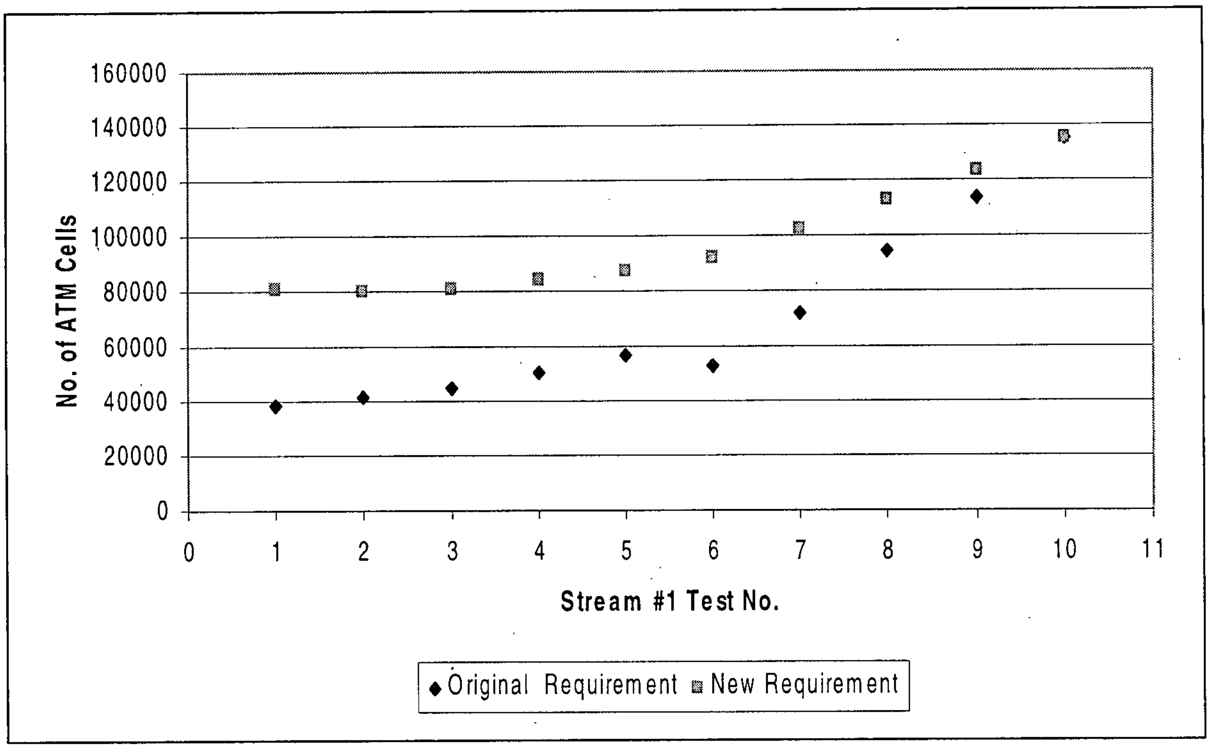

Figure 5-1: Original Requirements vs. New Requirement for Stream #1

Test No.	HP ATM cells transmitted	Extra ATM cells required, Without LP data structure.	LP ATM cells introduced.
2-1	317412	18338	31764 (173%)
2-2	314315	21435	37142 (173%)
2-3	311117	24633	40427 (164%)
2-4	307579	28171	46159 (164%)
2-5	302359	33391	52089 (156%)
2-6	308755	26995	41035 (152%)
2-7	295686	40064	49721 (124%)
2-8	282711	53039	60756 (115%)
2-9	268237	67513	70936 (105%)
2-10	255103	80647	80355 (100%)
Total ATM cells required for 100% transmission is 335750 ATM cells			

Table 5-6: No. of LP ATM Cells Introduced for Stream #2

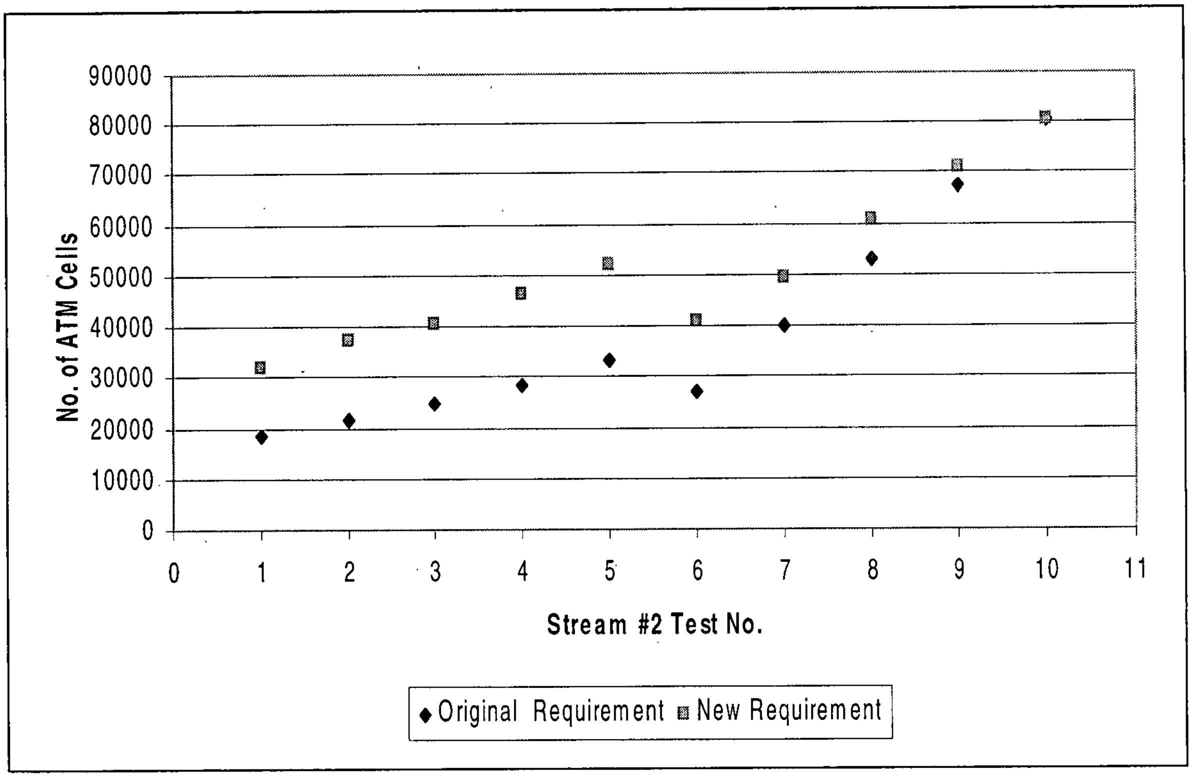

Figure 5-2: Original Requirements vs. New Requirement for Stream #2

Test No.	HP ATM cells transmitted	Extra ATM cells required, Without LP data structure.	LP ATM cells introduced.
3-1	387970	35386	77382 (219%)
3-2	385274	38082	80071 (210%)
3-3	383878	39478	81912 (207%)
3-4	381648	41708	84738 (203%)
3-5	379733	43623	85853 (197%)
3-6	369715	53641	89582 (167%)
3-7	352178	71178	100896 (142%)
3-8	332103	91253	112492 (123%)
3-9	314745	108611	122192 (113%)
3-10	295150	128206	132833 (104%)
Total ATM cells required for 100% transmission is 423356 ATM cells			

Table 5-7: No. of LP ATM Cells Introduced for Stream #3

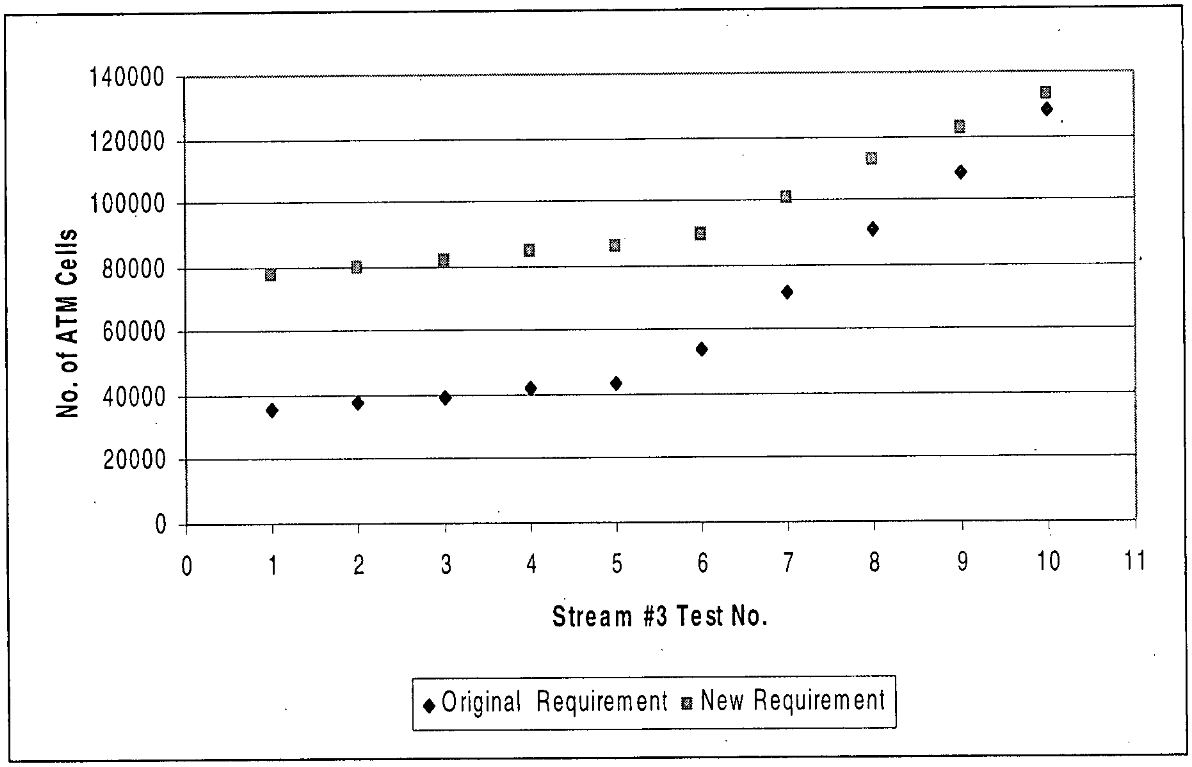

Figure 5-3: Original Requirements vs. New Requirement for Stream #3

5.3 Preset vs. Effective ATM Traffic Parameters

Tables 5-8 to 5-10 show the effective values and the preset values for the three ATM service contract parameters for the test cases involving the three video streams. From the simulation results, it is observed that the effective PCR and effective MBS are closely matched to their corresponding settings. Therefore, we focus on comparisons between the effective SCR and preset SCR. Figures 5-4 to 5-6 give the graphical comparison for the effective SCR versus preset SCR. From the figures, the proposed MPEG2-to-ATM converter produces video streams which closely follow the preset SCR. However, for cases 1-6 to 1-10, the effective SCR is slightly higher than the preset SCR. In these cases, the capability of the MPEG2-to-ATM converter has reached its limit. The same applies to video stream #2 and stream #3. This functional limit can be changed by adjusting the design or system parameters of the MPEG2-to-ATM converter. For example, a longer buffering period can be used instead of the 2 second period used currently. The minimum I-frame break point can be set to a lower value such as 12 (value used is 16). The minimum token pool level can be set to a higher value such as 200 (value used is 25). Values of Reduction_step and Max_reduction can also be increased.

Test No.	SCR Setting	SCR Effective	PCR Setting	PCR Effective	MBS Setting	MBS Effective
Converter 1-1	171	171	250	249	26382	26828
Un-Controlled 1-1	171	147	250	250	26382	26832
Fixed-Controlled 1-1	171	164	250	244	26382	26832
Converter 1-2	171	171	250	249	23850	23847
Un-Controlled 1-2	171	143	250	250	23850	23850
Fixed-Controlled 1-2	171	159	250	244	23850	23850
Converter 1-3	171	169	250	249	20869	20865
Un-Controlled 1-3	171	136	250	250	20869	20869
Fixed-Controlled 1-3	171	155	250	244	20869	20869

Converter 1-4	171	167	250	249	17888	17885
Un-Controlled 1-4	171	128	250	250	17888	17888
Fixed-Controlled 1-4	171	150	250	244	17888	17888
Converter 1-5	171	165	250	249	14907	14904
Un-Controlled 1-5	171	117	250	250	14907	14907
Fixed-Controlled 1-5	171	143	250	244	14907	14907
Converter 1-6	162	166	250	249	29813	29811
Un-Controlled 1-6	162	119	250	250	29813	29813
Fixed-Controlled 1-6	162	142	250	244	29813	29813
Converter 1-7	154	159	250	247	29813	29813
Un-Controlled 1-7	154	101	250	262	29813	29813
Fixed-Controlled 1-7	154	114	250	244	29813	29813
Converter 1-8	145	150	250	247	29813	29794
Un-Controlled 1-8	145	86	250	250	29813	29813
Fixed-Controlled 1-8	145	95	250	244	29813	29813
Converter 1-9	137	142	250	247	29813	29804
Un-Controlled 1-9	137	74	250	250	29813	29813
Fixed-Controlled 1-9	137	81	250	244	29813	29813
Converter 1-10	128	134	250	247	29813	29802
Un-Controlled 1-10	128	62	250	250	29813	29813
Fixed-Controlled 1-10	128	67	250	244	29813	29813

Table 5-8: Effective UPC parameters vs. preset UPC parameters for stream #1

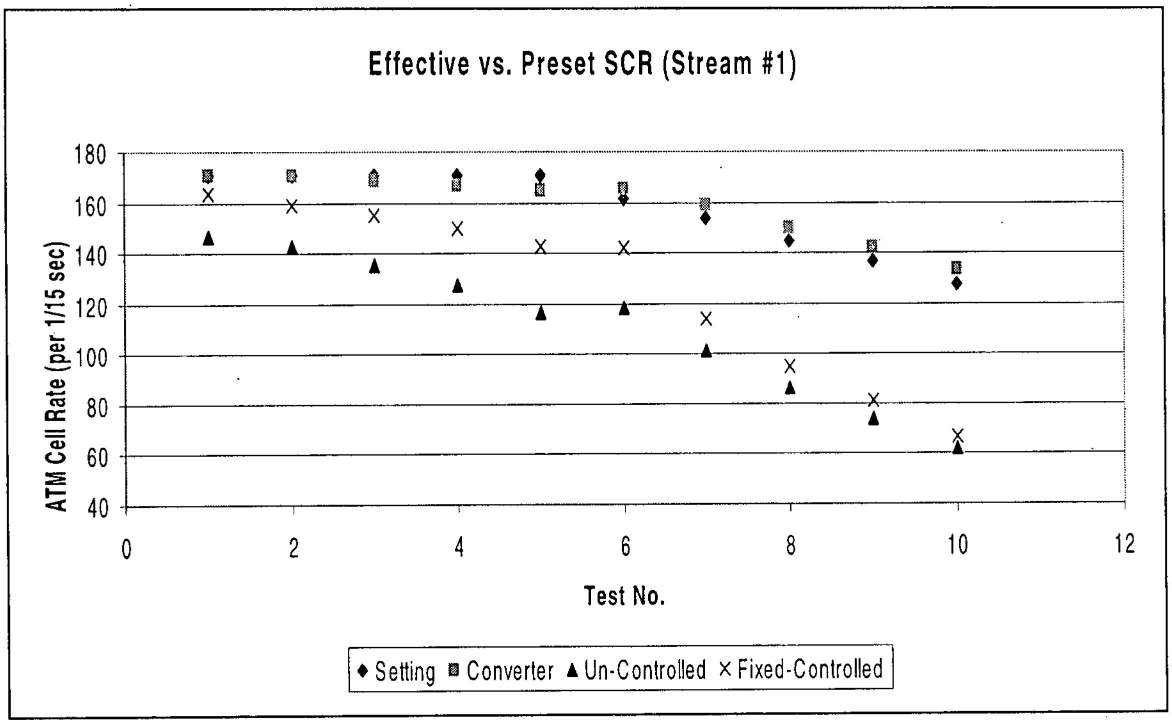

Figure 5-4: Effective vs. Preset SCR for Stream #1

Test No.	SCR Setting	SCR Effective	PCR Setting	PCR Effective	MBS Setting	MBS Effective
Converter 2-1	149	151.29	250	249.16	25734	25428
Un-Controlled 2-1	149	115.34	250	249.56	25734	25734
Fixed-Controlled 2-1	149	141.41	250	243.7	25734	25734
Converter 2-2	149	149.82	250	249.16	22874	22846
Un-Controlled 2-2	149	112.61	250	249.56	22874	22874
Fixed-Controlled 2-2	149	132	250	243.7	22874	22874
Converter 2-3	149	148.29	250	249.16	20015	20011
Un-Controlled 2-3	149	108.36	250	249.56	20015	20015
Fixed-Controlled 2-3	149	123.65	250	243.7	20015	20015
Converter 2-4	149	146.61	250	249.16	17156	17063
Un-Controlled 2-4	149	102.72	250	249.56	17156	17156
Fixed-Controlled 2-4	149	120.03	250	243.7	17156	17156
Converter 2-5	149	144.12	250	247.98	14297	14293
Un-Controlled 2-5	149	90.59	250	249.56	14297	14297
Fixed-Controlled 2-5	149	105.83	250	243.7	14297	14297

Converter 2-6	142	147.17	250	249.16	28593	28590
Un-Controlled 2-6	142	109.81	250	249.56	28593	28593
Fixed-Controlled 2-6	142	121.43	250	243.7	28593	28593
Converter 2-7	134	140.94	250	249.16	28593	28598
Un-Controlled 2-7	134	101.14	250	249.56	28593	28593
Fixed-Controlled 2-7	134	103.73	250	243.7	28593	28593
Converter 2-8	127	134.75	250	249.16	28593	28591
Un-Controlled 2-8	127	90.9	250	249.56	28593	28593
Fixed-Controlled 2-8	127	94.67	250	243.7	28593	28593
Converter 2-9	119	127.85	250	249.16	28593	28567
Un-Controlled 2-9	119	79.72	250	249.31	28593	28593
Fixed-Controlled 2-9	119	83.13	250	243.7	28593	28593
Converter 2-10	112	121.59	250	247.98	28593	28586
Un-Controlled 2-10	112	67.05	250	249.31	28593	28593
Fixed-Controlled 2-10	112	72.51	250	243.7	28593	28593

Table 5-9: Effective UPC parameters vs. preset UPC parameters for stream #2

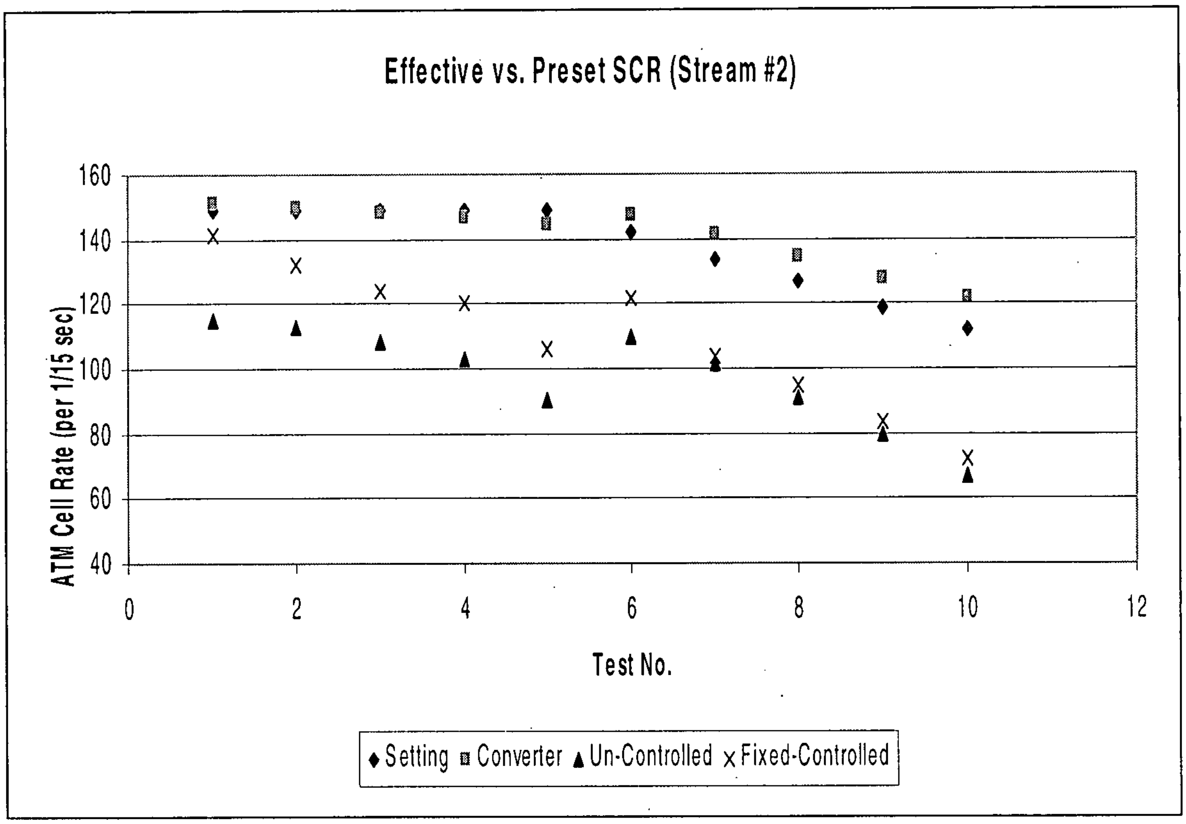

Figure 5-5: Effective vs. Preset SCR for Stream #2

Test No.	SCR Setting	SCR Effective	PCR Setting	PCR Effective	MBS Setting	MBS Effective
Converter 3-1	171	170.31	250	247.43	17353	17349
Un-Controlled 3-1	171	132.74	250	249.92	17353	17353
Fixed-Controlled 3-1	171	165.58	250	243.7	17353	17353
Converter 3-2	171	169.13	250	247.43	14625	14595
Un-Controlled 3-2	171	124.83	250	249.92	14625	14625
Fixed-Controlled 3-2	171	154.24	250	243.7	14625	14625
Converter 3-3	171	168.52	250	247.43	13497	13493
Un-Controlled 3-3	171	121.58	250	249.92	13497	13497
Fixed-Controlled 3-3	171	149.85	250	243.7	13497	13497
Converter 3-4	171	167.54	250	248.29	11569	11565
Un-Controlled 3-4	171	119.39	250	249.92	11569	11569
Fixed-Controlled 3-4	171	141.91	250	243.7	11569	11569
Converter 3-5	171	166.7	250	244.37	9641	9637
Un-Controlled 3-5	171	117.09	250	249.92	9641	9641
Fixed-Controlled 3-5	171	133.87	250	243.7	9641	9641
Converter 3-6	162	162.3	250	247.43	19281	19279
Un-Controlled 3-6	162	88.58	250	249.92	19281	19281
Fixed-Controlled 3-6	162	109.51	250	243.7	19281	19281
Converter 3-7	154	154.6	250	247.43	19281	19278
Un-Controlled 3-7	154	65.78	250	249.92	19281	19281
Fixed-Controlled 3-7	154	79.82	250	243.7	19281	19281
Converter 3-8	145	145.79	250	247.43	19281	19233
Un-Controlled 3-8	145	50.59	250	249.92	19281	19281
Fixed-Controlled 3-8	145	64.02	250	243.7	19281	19281
Converter 3-9	137	138.17	250	247.43	19281	19276
Un-Controlled 3-9	137	40.21	250	249.92	19281	19281
Fixed-Controlled 3-9	137	52.8	250	243.7	19281	19281
Converter 3-10	128	129.57	250	247.43	19281	19263
Un-Controlled 3-10	128	33.33	250	249.92	19281	19281
Fixed-Controlled 3-10	128	41.22	250	243.7	19281	19281

Table 5-10: Effective UPC parameters vs. preset UPC parameters for stream #3

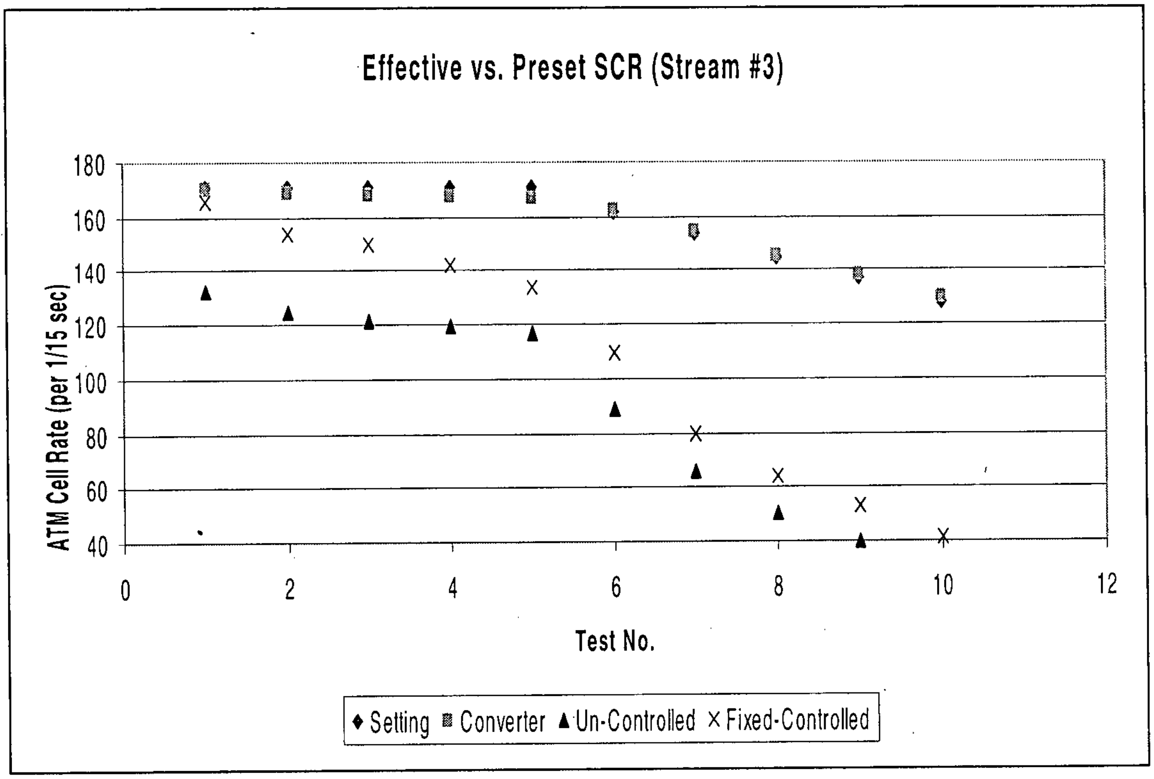

Figure 5-6: Effective vs. Preset SCR for Stream #3

5.4 Average PSNRs and Channel Efficiency at Receiver

In each case, after converting the received ATM cells into an MPEG2 stream, the average PSNRs for the entire video sequence and each individual frame are calculated. The data for average PSNR are shown in Figures 5-7 to 5-9 and tabulated in Tables 5-11 to 5-13, which also show the corresponding channel efficiency at the receiver. The channel efficiency at the receiver is defined as 1-(X/Y), where X = no. of received cells discarded by receiver due to missing video headers; Y = total no. of cells received, and reflects the number of ATM cells that are wasted due to the loss of the corresponding video headers. Since the outputs of the MPEG2-to-ATM converter follow the UPC service contract, all the ATM cells at the received end can be used by the decoder, resulting in 100% channel efficiency. However, the outputs of both the un-controlled and fixed-controlled methods may exceed (or violate) the established UPC service contract, in which case the nonconforming cells are dropped by UPC, resulting in incomplete AAL5 PDUs at the receiver. The decoder will not be able to used these incomplete AAL5 PDUs for video stream reconstruction.

For stream #2, the MPEG2-to-ATM converter method has the best average PSNR performance (Figure 5-8 and Table 5-12). This is an expected result, but the uncontrolled method performs better than fixed controlled method in terms of average PSNR. The reason is because the fixed controlled method wastes available tokens when token generation rate is greater than consumption rate and the Token Pool is full (referred to as Token Pool overflow period). However, the MPEG2-to-ATM converter method does not always perform better than the other two methods. For stream #1 (Figure 5-7, table 5-11), the uncontrolled method had the best average PSNR performance in the first three test cases. Figures A-1, A-3, and A-5 shows that the uncontrolled method has perfect transmission of video frames expect ranges from frame number 367 to 489, frame number 700 to 740, and frame number around 800, which yields higher average PSNR value than the MPEG2-toATM converter method, which has finite frame PSNR value for most of the video frames. However, this is not the case for stream #3 (Figure 5-9 and Table 5-

13). Figure A-13 shows that the uncontrolled method has a number of almost zero transmissions of video frames (frame PSNR is around 20 dB) that yields lower average PSNR. Although the MPEG2-toATM converter has a much better frame PSNR performance, when fixed-controlled method has the frame PSNR around 20 dB, most of the time, fixed-controlled method has higher frame PSNR Therefore, the fixed-controlled method has the best average PSNR performance in the first test case.

It was observed that the average PSNR does not match the results of subjective evaluations of the received video playback through a hardware MPEG2 decoder (please refer to the graphs in the appendix). Therefore, we introduce another method to evaluate the picture quality, which, instead of using the average PSNR, uses a frame by frame PSNR comparison.

Test No.	Converter PSNR (channel efficiency)	Un-controlled PSNR (channel efficiency)	Fixed-controlled PSNR (channel efficiency)
1-1	38.45 dB (100%)	41.6 dB (85.6%)	38.78 dB (95.5%)
1-2	38.35 dB (100%)	40.54 dB (83.9%)	37.88 dB (93.1%)
1-3	38.14 dB (100%)	38.66 dB (80.2%)	37.14 dB (91.2%)
1-4	37.74 dB (100%)	37.15 dB (76.4%)	36.5 dB (89.8%)
1-5	37.46 dB (100%)	35.12 dB (70.9%)	35.45 dB (86.5%)
1-6	37.45 dB (100%)	36.06 dB (71.8%)	35.2 dB (85.1%)
1-7	35.51 dB (100%)	33.26 dB (63.7%)	32.11 dB (71.8%)
1-8	34.11 dB (100%)	31.51 dB (57.5%)	30.26 dB (63.3%)
1-9	33.10 dB (100%)	30.17 dB (52.0%)	29.18 dB (56.9%)
1-10	32.02 dB (100%)	29.05 dB (46.2%)	28.11 dB (50.2%)
PSNR for the 484848 case is 40.40 dB			

Table 5-11: Average PSNR and channel efficiency for Stream #1

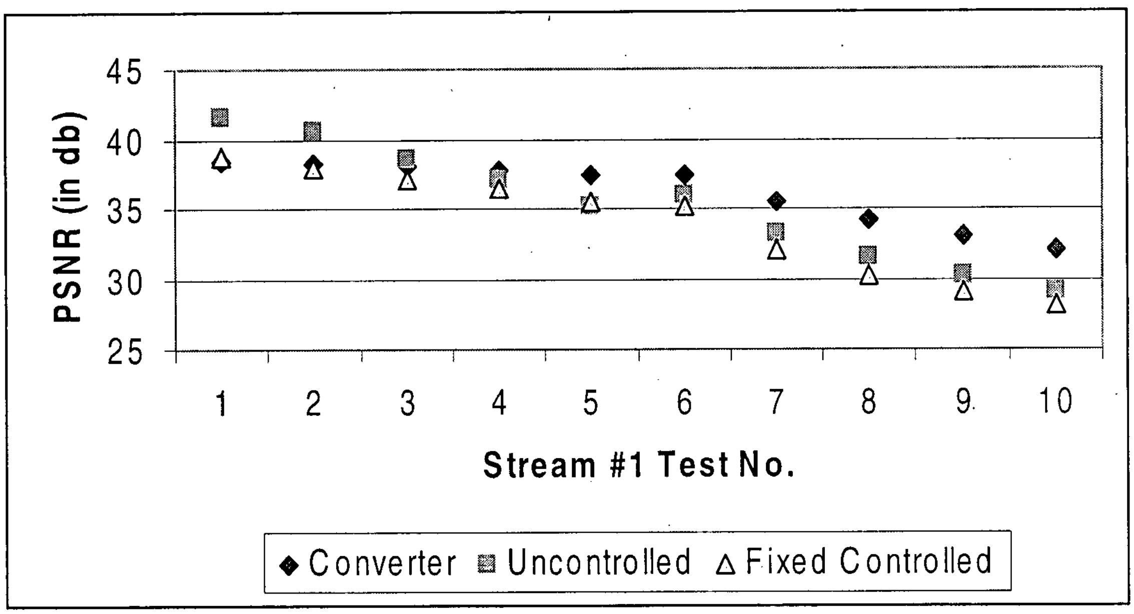

Figure 5-7: Average PSNR for Stream #1

Test No.	Converter PSNR (channel efficiency)	Un-controlled PSNR (channel efficiency)	Fixed-controlled PSNR (channel efficiency)
2-1	49.53 (100%)	40.09 (76.2%)	38.19 (93.5%)
2-2	45.93 (100%)	39.58 (75.2%)	36.63 (88.1%)
2-3	44.49 (100%)	38.53 (73.1%)	35.37 (83.4%)
2-4	43.25 (100%)	37.35 (70.1%)	34.97 (81.9%)
2-5	42.03 (100%)	35.48 (62.9%)	33.23 (73.4%)
2-6	44.87 (100%)	38.9 (74.6%)	35.05 (82.5%)
2-7	41.75 (100%)	36.98 (71.8%)	32.88 (73.6%)
2-8	39.51 (100%)	35.37 (67.5%)	32.04 (70.3%)
2-9	37.75 (100%)	33.47 (62.4%)	30.94 (65%)
2-10	36.41 (100%)	31.82 (55.1%)	29.87 (59.6%)
484848 case is 39.92 dB			

Table 5-12: Average PSNR and channel efficiency for Stream #2

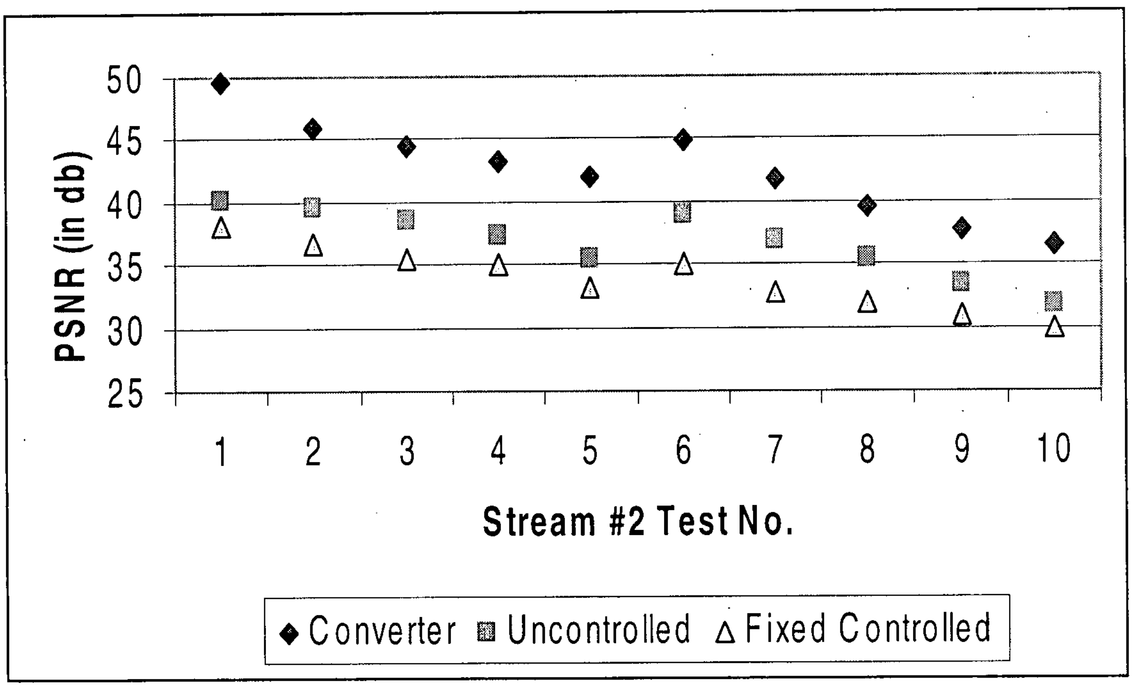

Figure 5-8: Average PSNR for Stream #2

Test No.	Converter PSNR (channel efficiency)	Un-controlled PSNR (channel efficiency)	Fixed-controlled PSNR (channel efficiency)
3-1	38.28 (100%)	37.97 (77.9%)	39.47 (97.2%)
3-2	37.98 (100%)	36.42 (73.8%)	37.22 (91.2%)
3-3	37.81 (100%)	35.81 (72.1%)	36.31 (88.9%)
3-4	37.5 (100%)	35.46 (71.3%)	35.21 (84.7%)
3-5	37.37 (100%)	35.14 (70.2%)	33.98 (80.3%)
3-6	36.74 (100%)	31.93 (54.6%)	31.85 (67.5%)
3-7	34.93 (100%)	29.61 (42.5%)	29.22 (51.6%)
3-8	33.4 (100%)	28.23 (34.7%)	27.83 (43.9%)
3-9	32.35 (100%)	27.63 (29.1%)	27.14 (38.2%)
3-10	31.18 (100%)	27.06 (25.7%)	26.45 (31.8%)
484848 case is 39.92 dB			

Table 5-13: Average PSNR and channel efficiency for Stream #3

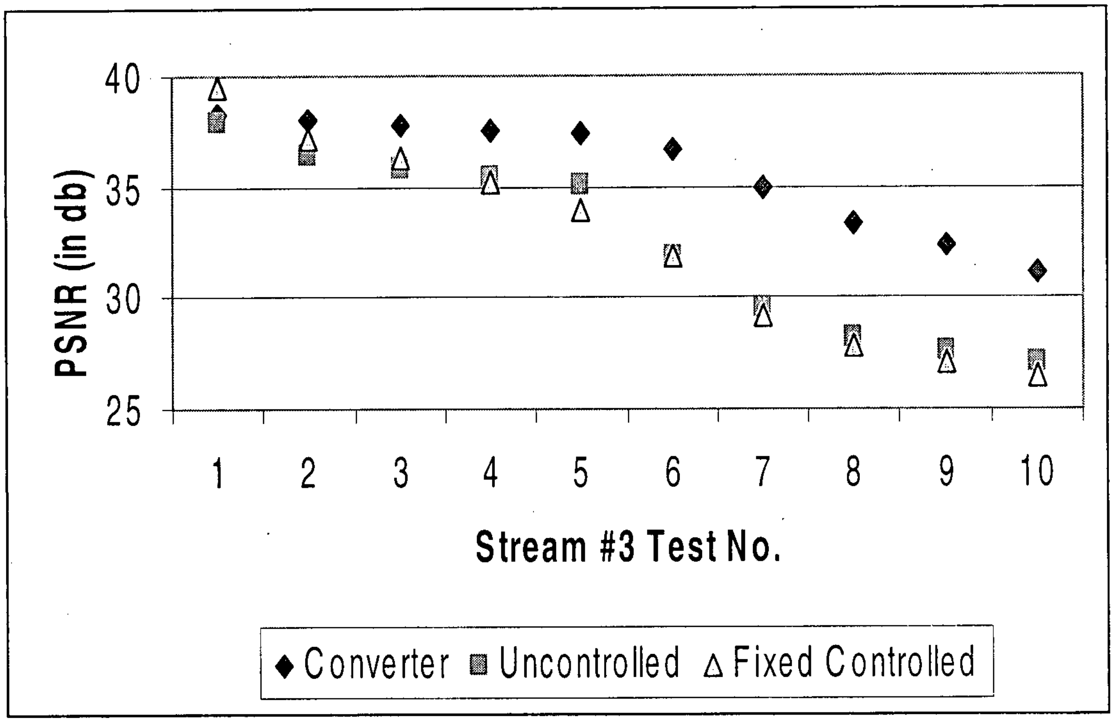

Figure 5-9: Average PSNR for Stream #3

5.5 Frame by Frame PSNR Evaluation

Subjective evaluations show that the converter produces the best picture quality, while the other two methods yield substantially lower subjective picture quality. We hypothesize that the received video quality is heavily influenced by the frames that have poor PSNR. Therefore, the frame by frame PSNRs of the three methods considered are compared.

The PSNR values are calculated for individual frames. From subjective tests, it was found that the 242424 case still gives a good picture quality while the 161616 case gives only a marginally acceptable picture quality when the video stream is decoded without subjecting to ATM network degradations. Therefore, the frame-based PSNR of 242424 and 161616 cases are used as references for comparison between the three methods. Please find the graphs of frame by frame PSNR of the whole transmission period for these three video streams in the appendix. Tables 5-14 to 5-16 show the percentages of number of frames which PSNRs fall below the 242424 PSNR and 161616 PSNR. Figure 5-10 to Figure 5-15 are the graphical representations.

All the results of frame by frame PSNR comparison show that the MPEG2-to-ATM converter gives the least number of frames which PSNRs are lower than those of the corresponding frames in the 242424 case (i.e., below the 242424 PSNR) and no frames below the 161616 PSNR. On the other hand, both uncontrolled and fixed-controlled methods have most of the low PSNR frames falling below the 161616 PSNR. This is in agreement with subjective evaluating.

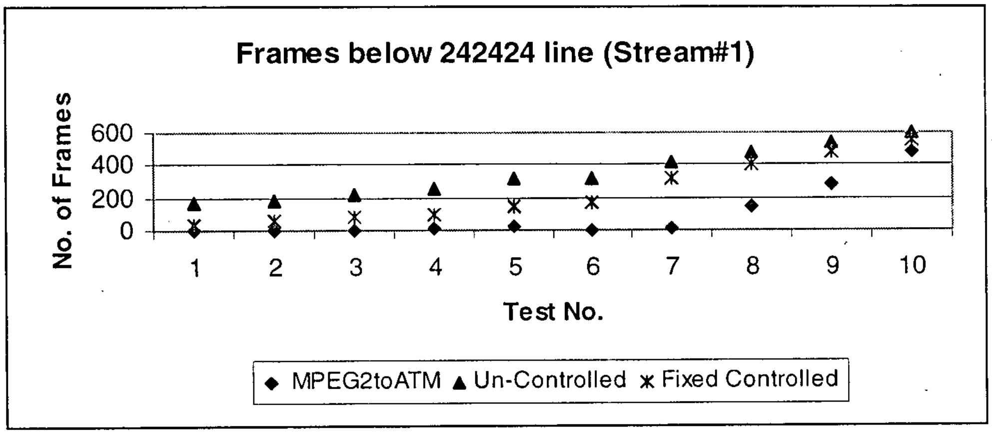

Figure 5-10: No. of Frames Below 242424 PSNR Line for Stream #1

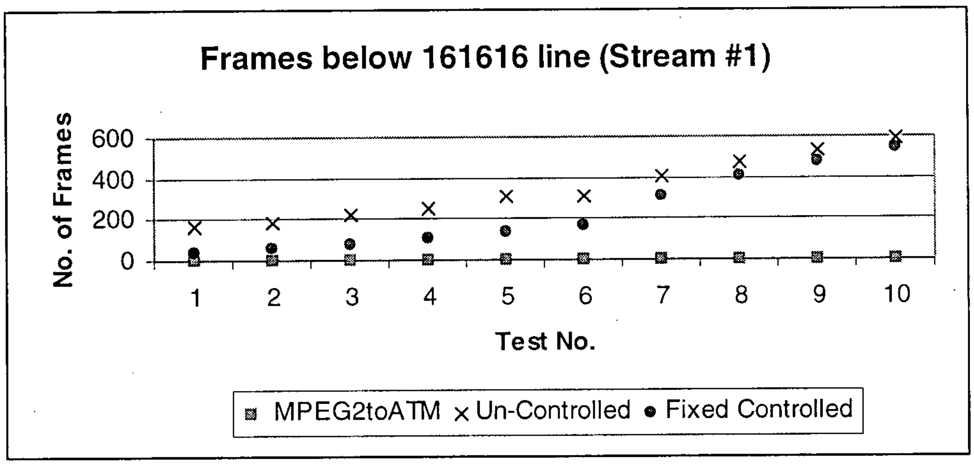

Figure 5-11: No. of Frames Below 161616 PSNR Line for Stream #1

Test No.	Converter		Uncontrolled		Fixed-Controlled	
	Below 242424	*Below 161616*	Below 242424	*Below 161616*	Below 242424	*Below 161616*
1-1	0%	*0%*	13%	*13%*	3%	*3%*
1-2	0%	*0%*	14%	*14%*	5%	*5%*
1-3	0%	*0%*	17%	*17%*	6%	*6%*
1-4	1%	*0%*	20%	*19%*	8%	*8%*
1-5	2%	*0%*	24%	*23%*	11%	*10%*
1-6	0%	*0%*	24%	*24%*	13%	*12%*

1-7	1%	*0%*	31%	*31%*	24%	*24%*
1-8	11%	*0%*	36%	*36%*	31%	*31%*
1-9	21%	*0%*	41%	*41%*	36%	*36%*
1-10	36%	*0%*	46%	*45%*	42%	*41%*

Table 5-14: Percentage of Number of Frames below 242424 PSNR and 161616 PSNR Lines for Stream #1

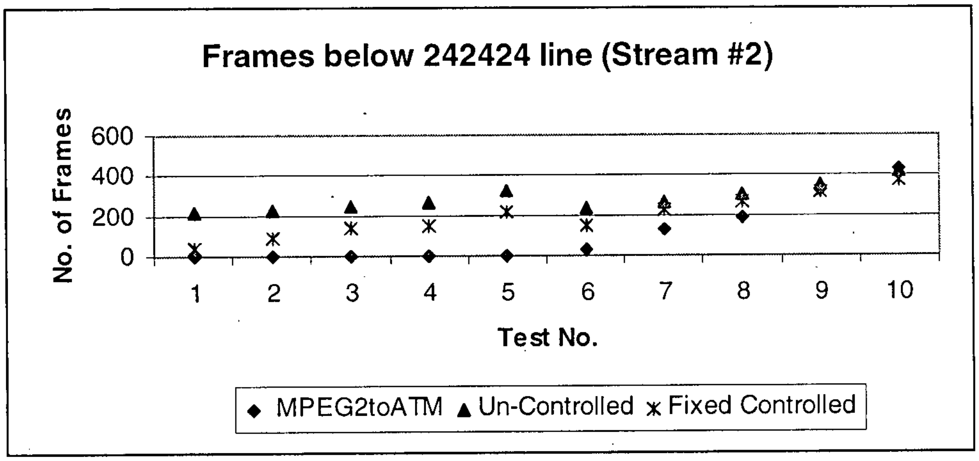

Figure 5-12: No. of Frames Below 242424 PSNR Line for Stream #2

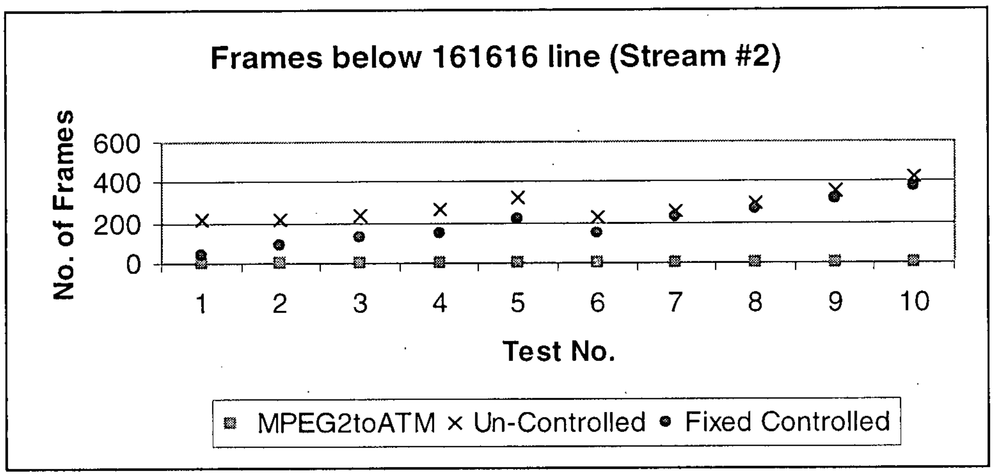

Figure 5-13: No. Of Frames Below 161616 PSNR Line for Stream #2

Test No.	Converter		Uncontrolled		Fixed-Controlled	
	Below 242424	*Below 161616*	Below 242424	*Below 161616*	Below 242424	*Below 161616*
2-1	0%	*0%*	20%	*21%*	4%	*4%*
2-2	0%	*0%*	21%	*21%*	9%	*9%*
2-3	0%	*0%*	23%	*23%*	13%	*12%*
2-4	0%	*0%*	25%	*25%*	14%	*14%*
2-5	0%	*0%*	31%	*31%*	21%	*21%*
2-6	3%	*0%*	22%	*22%*	14%	*14%*
2-7	12%	*0%*	25%	*25%*	22%	*22%*
2-8	18%	*0%*	29%	*28%*	25%	*25%*
2-9	31%	*0%*	34%	*34%*	30%	*30%*
2-10	42%	*0%*	40%	*40%*	35%	*35%*

Table 5-15: Percentage of Number of Frames below 242424 PSNR and 161616 PSNR Lines for Stream #2

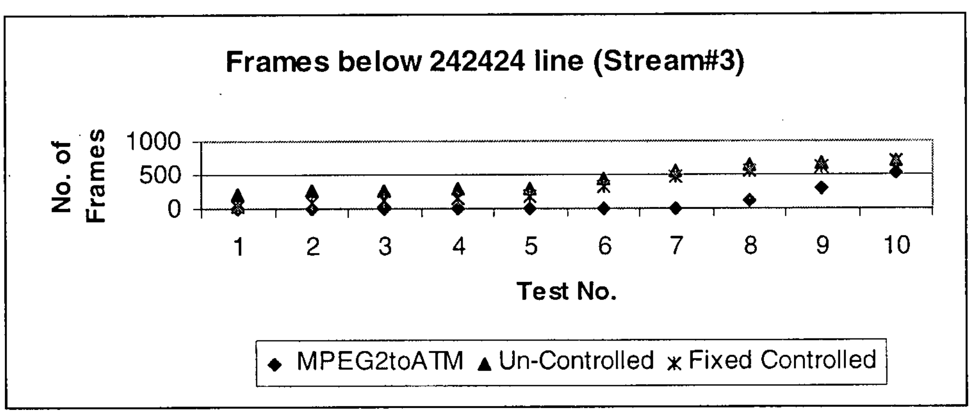

Figure 5-14: No. Of Frames Below 242424 PSNR Line for Stream #3

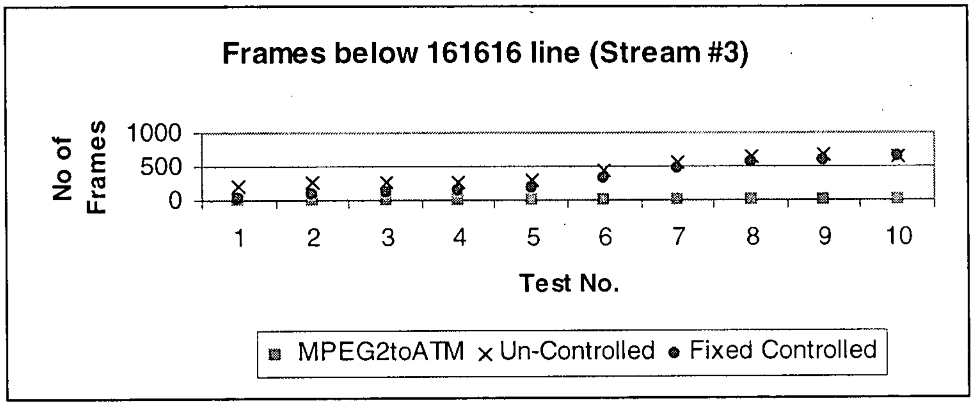

Figure 5-15: No. Of Frames Below 161616 PSNR Line for Stream #3

Test No.	Converter		Uncontrolled		Fixed-Controlled	
	Below 242424	*Below 161616*	Below 242424	*Below 161616*	Below 242424	*Below 161616*
3-1	0%	0%	19%	19%	2%	2%
3-2	0%	0%	23%	22%	7%	7%
3-3	0%	0%	24%	24%	9%	9%
3-4	0%	0%	25%	25%	13%	12%
3-5	0%	0%	26%	25%	17%	16%
3-6	0%	0%	40%	39%	28%	28%
3-7	1%	0%	50%	50%	42%	42%
3-8	9%	0%	57%	57%	49%	49%
3-9	25%	0%	61%	61%	54%	53%
3-10	48%	0%	64%	58%	63%	58%

Table 5-16: Percentage of Number of Frames below 242424 PSNR and 161616 PSNR Lines for Stream #3

Chapter 6 Conclusions

6.1 Summary

We have presented a novel method for post-encoding source rate control of VBR MPEG2 video data to conform to ATM UPC policies at the network interface. This MPEG2-to-ATM converter functions as an external post-coding source rate controller between any MPEG2 source (video archives, life programs, etc.) and an ATM network.

Dynamic break points are employed to shape and partition the video data into high priority ATM cells which conform to the prevailing UPC contract with the network, and nonconforming low priority cells which minimize the effects of cell loss due to network congestion on the subjective quality of the received video. Both high and low priority cells are transmitted over a common ATM virtual connection to achieve high quality rt-VBR video transmission. Since only one ATM transmission channel is required, there is no channels synchronization issue at the receiver end. Furthermore, our designed data structure introduces almost zero overhead for the high priority data.

This MPEG2-to-ATM converter is designed in such a way that the receiving end may need no special equipment (in our case, the ATM-to-MPEG2 converter) in order minimize the impact of the actual implementation. Unlike many proposed methods, current MPEG2 decoders without extra hardware or major software upgrade can decode the video streams from this MPEG2-to-ATM converter.

In teams of quality comparison, a new method, individual frame by frame's PSNR comparison, has been proposed as an alternative of the conventional average PSNR comparison method. From the simulation results, it is apparent that our MPEG2-to-ATM converter significantly improves the picture quality of video by maintaining a high PSNR for the majority of received frames. Because frames with very low PSNR have a much greater effect in degrading the subjective quality of the received video, the MPEG2-to-ATM converter gives the best performance in terms of subjective quality, as confirmed by subjective evaluations.

6.2 Further Work

The block-based design (Figure 4-1) of the simulation model provides scalability and portability. Different system component models can be used in the simulation model for further research and evaluation. There are also a number of user-defined system parameters. For different kind of applications such as TV news, sports, and games, corresponding set of system parameters may be obtained. Different channel simulation models such as satellite channel can also be put into the ATM channel module. Different break points assignment algorithms, either based on BP look up table or other methods can be investigated. Another interesting area is that determination and establishment of ATM service contract based on limited information, for example, the video data in the ES buffer. ATM service contract re-negotiation is another one. The methods of changing the system parameters such as Min_Ipb dynamically to overcome the current capability limit of MPEG2-to-ATM converter are interesting as well. In short, there are many ways to enhance and modify this converter for further research and evaluations.

References

[1] F. Akyildiz, S. Hrastar, H. Uzunalioglu, and W. Yen, "*Comparison and Evaluation of Packing Schemes for MPEG2 over ATM using AAL5*", Proc. IEEE Int'l Conf. on Communications. pp.1411-1415, 1996

[2] G. J. Armitage and K.M. Adams, "*Packet reassembly during cell loss*", IEEE Network, vol. 7, no. 9, pp.26-34, Sept. 1993

[3] ATM Forum, "*Traffic Management Specification Version 4.0*", ATMF95-0013R10, 1995

[4] J. Brenneman, "*Comments on MPEG-2/AAL5 encapsulation*", ATM Forum Contribution ATM95-1265, Oct. 1995

[5] D.M. Cohen and D.P. Heyman, "*A simulated Study of Video Teleconferencing Traffic in ATM Networks*", Proc. IEEE INFOCOM, pp. 894-901, 1993

[6] R. Coelho and S. Tohme, "*Video coding mechanism to predict video traffic in ATM network*" Proc. IEEE GLOBECOM, pp.447-451, 1993

[7] P. Cuenca, L. Orozco-Barbosa, L. Wang, A. Garrido and F. Quiles, "*An Error Concealment Scheme for MPEG-2 Video Transmission Over ATM-Based Networks*", Proc. IEEE CCECE, 1997

[8] S. Dizit and P. Skelly, "*MPEG2 over ATM for Video Dial Tone Networks: Issues and Strategies*", IEEE Network Magazine, Vol. 9, No. 5, pp. 30-40, September 1995

[9] M. Ghanbari, "*Two-Layer Coding of Video Signals for VBR Networks*", IEEE JSAC, Vol 7, No. 5, Jun. 1989

[10] M. R. Grasse and J. F. Fraster, "*Traffic Characteristics of MPEG2 Variable Bit Rate Video*", Australian Telecommunication Networks & Applications Conference, pp. 473-478, Dec. 1994

[11] M. Hamdi, J. W. Roberts, and P. Rolin, "*Rate Control for VBR Video Coders in Broad-Band Networks*", IEEE Journal on Selected Areas in Communications, vol. 15, pp.1040-1051, August 1997

[12] B. G. Haskell, A. Puri and A. N. Netravali, "*Digital Video: An Introduction to MPEG-2*", Chapman & Hall, 1997

[13] H. Heeke, "*Traffic control algorithm for ATM networks*", IEEE Trans. On Circuits and Systems Video Technology, Vol. 3, pp.182-189, June 1993

[14] D.P. Heyman, A. Tabatabai, and T. V. Lakshman, "*Statistical Analysis and Simulation Study of Video Teleconference Traffic in ATM Networks*", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 2, pp.49-59, Mar. 1992

[15] M. R Ismall, I.E. Lambadaris, M. Devetsikiotis and A. R. Kaye, "*Modelling Prioritized MPEG Video Using TES and a Frame Spreading Strategy for Transmission in ATM Networks*", Proc. IEEE INFOCOM. Vol. 2, pp. 762-770, 1995

[16] ISO/IEC, "*Generic Coding of Moving Picture and Associated Audio, ISO/IEC 13818-2, MPEG-2 Draft International Standard*", May 1994

[17] ISO/IEC, "*Motion Pictures Expert Group 2, Recommendation H.262, ISO/IEC 13818-2, Committee Draft, Generic Coding of Moving Pictures and Associated Audio*", Nov. 1993

[18] ITU, "*Draft Recommendation H. 263: Video Coding for Low Bitrate Communication. ITU-T (CCITT)*", Dec. 1995

[19] ITU, "*Traffic Control and Congestion Control in B-ISDN*", ITU Study Group 13, Perth, Australia, Nov. 1995

[20] ITU, "*B-ISDN protocol reference model and its application*", ITU-T Recommendation I.321, June 1993

[21] B. Jabbari, F. Yegenoglu, Y. Kuo, S. Zafar and Y. Zhang "*Statistical Characterization and Block-Based Modeling of Motion-Adaptive Coded Video*" IEEE Transactions on Circuits and Systems for Video Technology, Vol.3, p199-207, June 1993

[22] H. Kanakia, P.P. Mishra, A. Reibman, "*An Adaptive Congestion Control Scheme for Real-Time Packet Video Transport*", ACM SIGCOMM '93 Ithaca, New York, Sept. 1993

[23] J.P. Leduc, "*Digital Moving Pictures - Coding and Transmission on ATM Networks*". New York: Elsevier, 1994

[24] B. G. Lee, M. Kang, and J. Lee, “*Broadband Telecommunications Technology*”, Artech House, Boston, 1993

[25] W. Luo and M. E. Zarki, "*Quality Control for VBR Video over ATM Network*", IEEE Journal on Selected Areas in Communications, Vol. 15, pp. 1029-1039, Aug. 1997

[26] W. Luo and M. E. Zarki, "*Adaptive Data Partitioning For MPEG-2 Video Transmission Over ATM Based Networks*", Proc. IEEE Int'l Conf. on Image Processing, Vol1, pp.17-20, 1996

[27] J. L. Mitchell, W. B. Pennebaker, C. E. Fogg, and D. J. LeGall, "*MPEG Video Compression Standard*", Chapman & Hall, New York, 1996

[28] P. Nasiopoulos, "*A course on DVD Authoring*", UBC Graduate Course material, 1997

[29] J. Ni, T. Yang and D.H.K. Tsang, "*CBR Transportation of VBR MPEG2 Video Traffic for Video-On-Demand in ATM Networks*", Proc. ICC/SUPERCOMM '96, Vol.3 pp1391-1395, June 1996

[30] P. Pancha and M. E. Zarki, "*MPEG Coding for Variable Bit Rate Video Transmission*", IEEE Communications Magazine, pp54-66, May 1994

[31] P. Pancha and M. E. Zarki, "*Leaky bucket access control for VBR MPEG video*", Proc. IEEE INFOCOM, pp.796-803, Boston, Mar-Apr. 1995

[32] P. Pancha and M. E. Zarki, "*Prioritized Transmission of VBR MPEG Video*", Proc. IEEE GLOBECOM, pp.1135-1139, 1992

[33] M. R. Pickering and J. F. Arnold, "*A perceptually efficient VBR rate control algorithm*", IEEE Transactions on Image Processing, vol. 3, pp.527-532, Sept. 1994

[34] M. D. Prycker, "*Asynchronous Transfer Mode, solution for broadband ISDN*", second Edition, New York: Ellis Horwood, 1993

[35] E. Rathgeb, "*Policing of realistic VBR video traffic in ATM networks*", Int'l J. Digital Analog Communication System, vol. 6, pp. 213-226, 1993

[36] A. R. Reibman and B.G. Haskell, "*Constraints on variable bit rate video for ATM networks*", IEEE Transactions on Circuits and Systems Video Technology, vol. 2, pp.

361-372, Dec. 1992

[37] D. Reininger and D. Raychaudhuri, "*Bit-Rate Characteristics of a VBR MPEG Video Encoder for ATM Networks*", Proc. IEEE ICC, pp517-521, 1993

[38] R. J. Siracusa, K. Joseph, J. Zdepski and D. Raychaudhuri, "*Flexible and Roburst Packet Transport for Digital HDTV*", IEEE JSAC, Vol. 11, No. 1, pp.88-98, Jan. 1993

[39] P. N. Tudor, "*MPEG-2 Video Compression Tutorial*", IEE, London WC2R 0BL, UK. 1995

[40] S. Varma, "*MPEG-2 over ATM: System Design Issues*", Proc. IEEE COMPCON, pp. 26-31, 1996

[41] S. Vuong, "*Multimedia Communications*", DMS'97 conference tutorial material. 1997

[42] M. Zrunz, R. Sass, and H. Hughes, "*Statistical Characteristics and Multiplexing of MPEG-coded Video Streams*", Proc. IEEE INFOCOM, 1995

Appendix:

In this appendix, the detailed frame by frame comparison for each test case are shown in graphs. Also the virtual token pool levels before and after BP assignment in the MPEG2-to-ATM converter for each test case are shown.

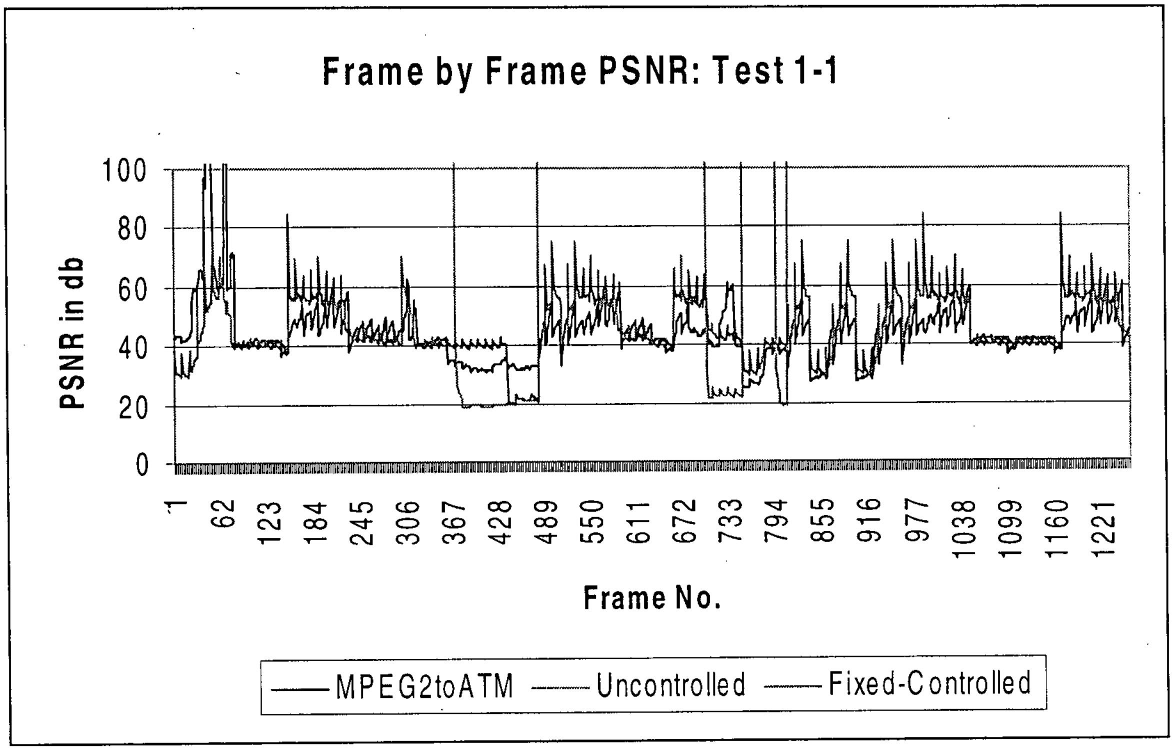

Figure A-1: Frame by Frame PSNR Comparison for Test 1-1

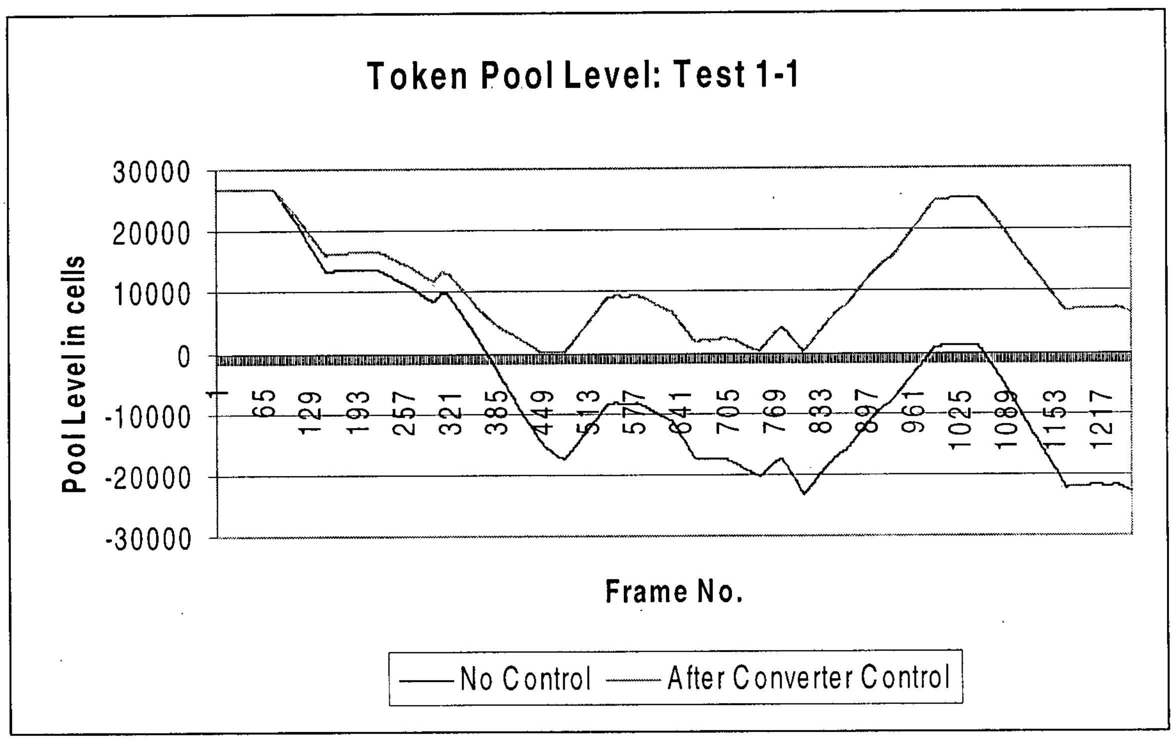

Figure A-2: Token Pool Levels Comparison of No Control and Converter Controlled for Test 1-1

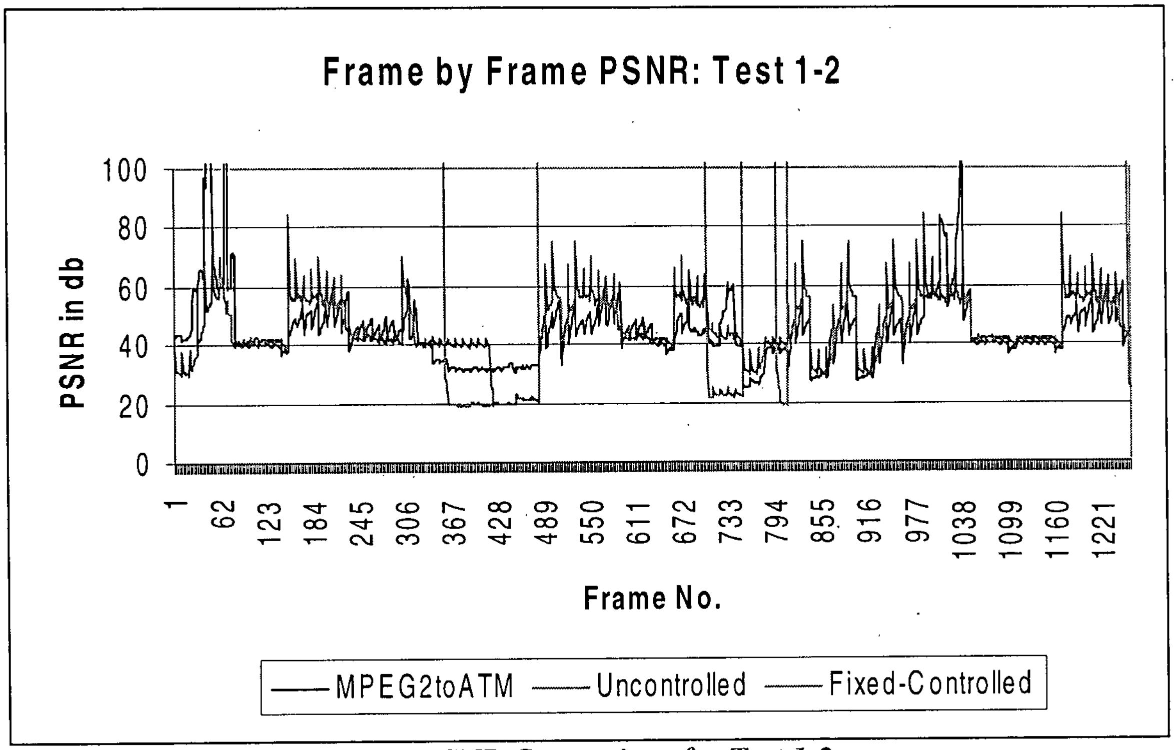

Figure A-3: Frame by Frame PSNR Comparison for Test 1-2

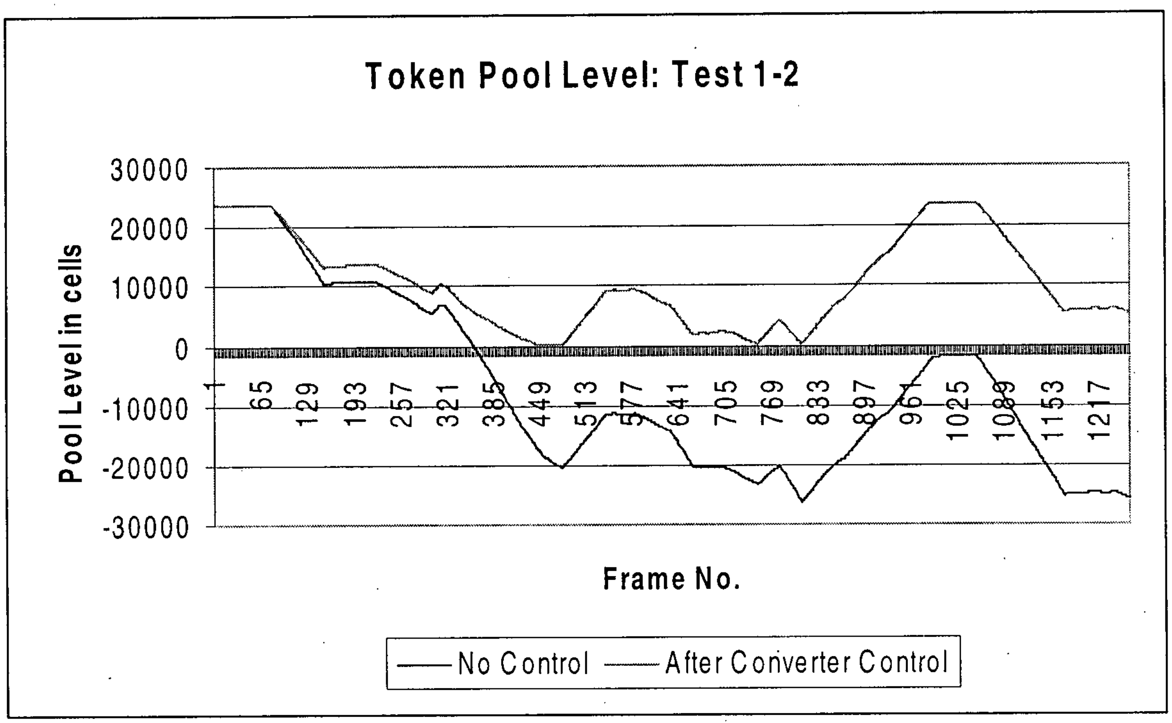

Figure A-4: Token Pool Levels Comparison of No Control and Converter Controlled for Test 1-2

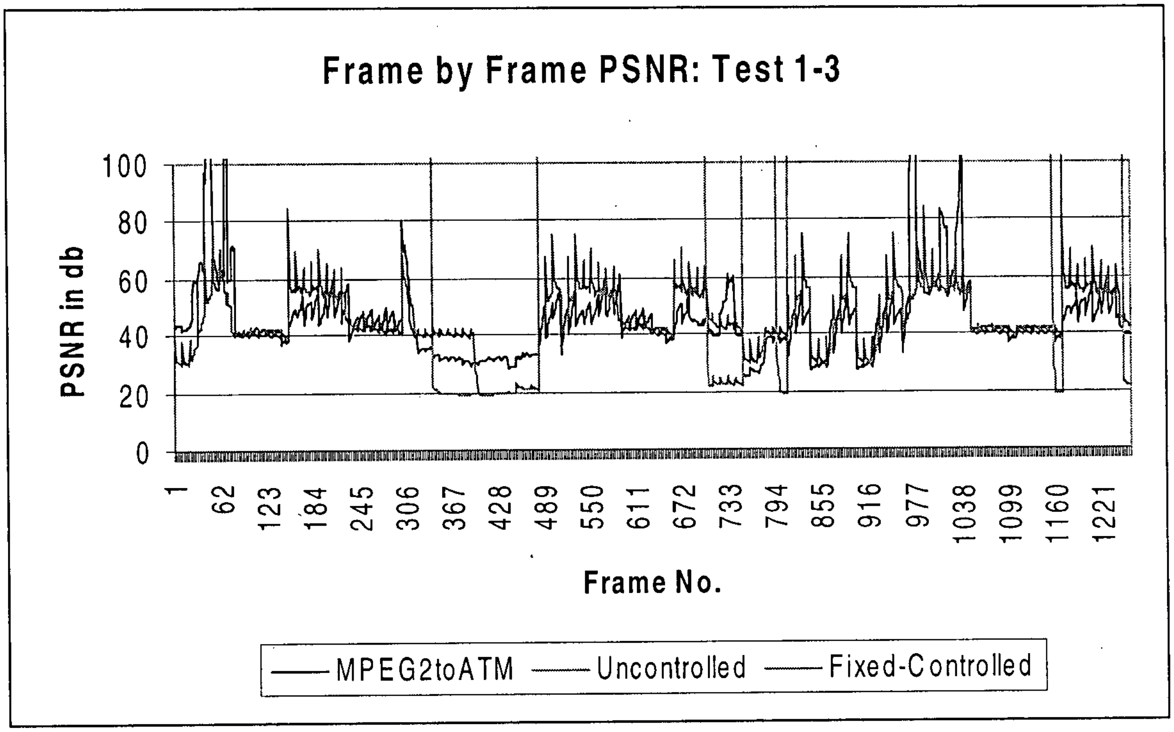

Figure A-5: Frame by Frame PSNR Comparison for Test 1-3

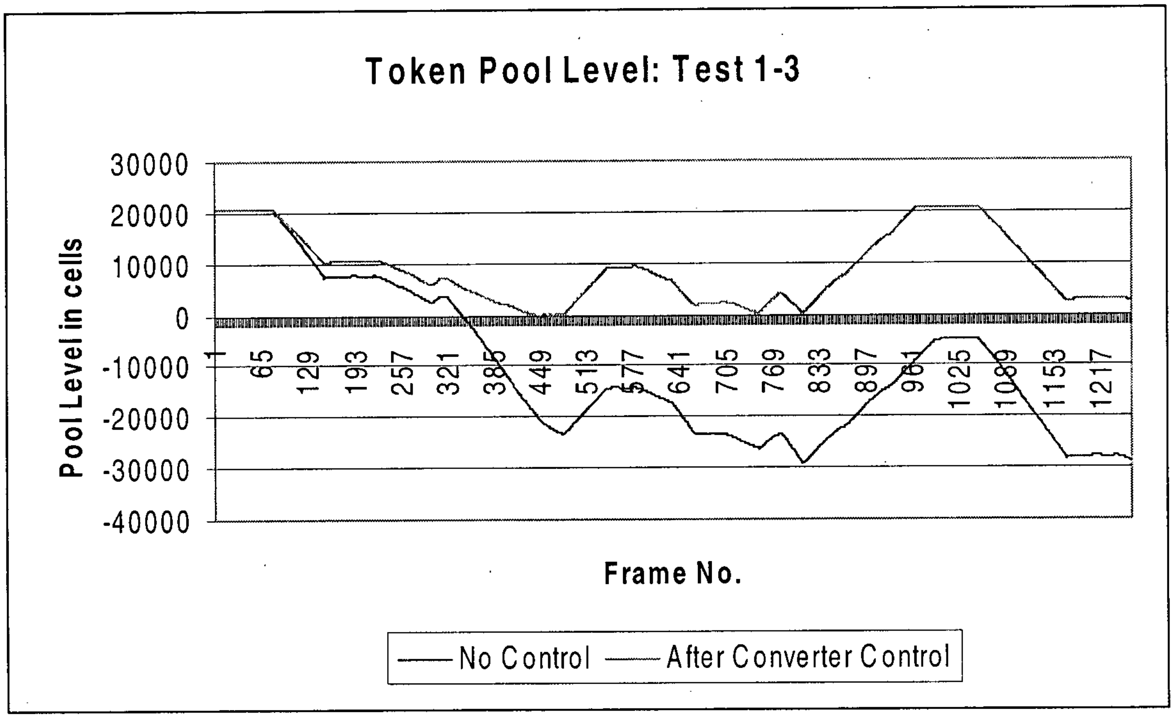

Figure A-6: Token Pool Levels Comparison of No Control and Converter Controlled for Test 1-3

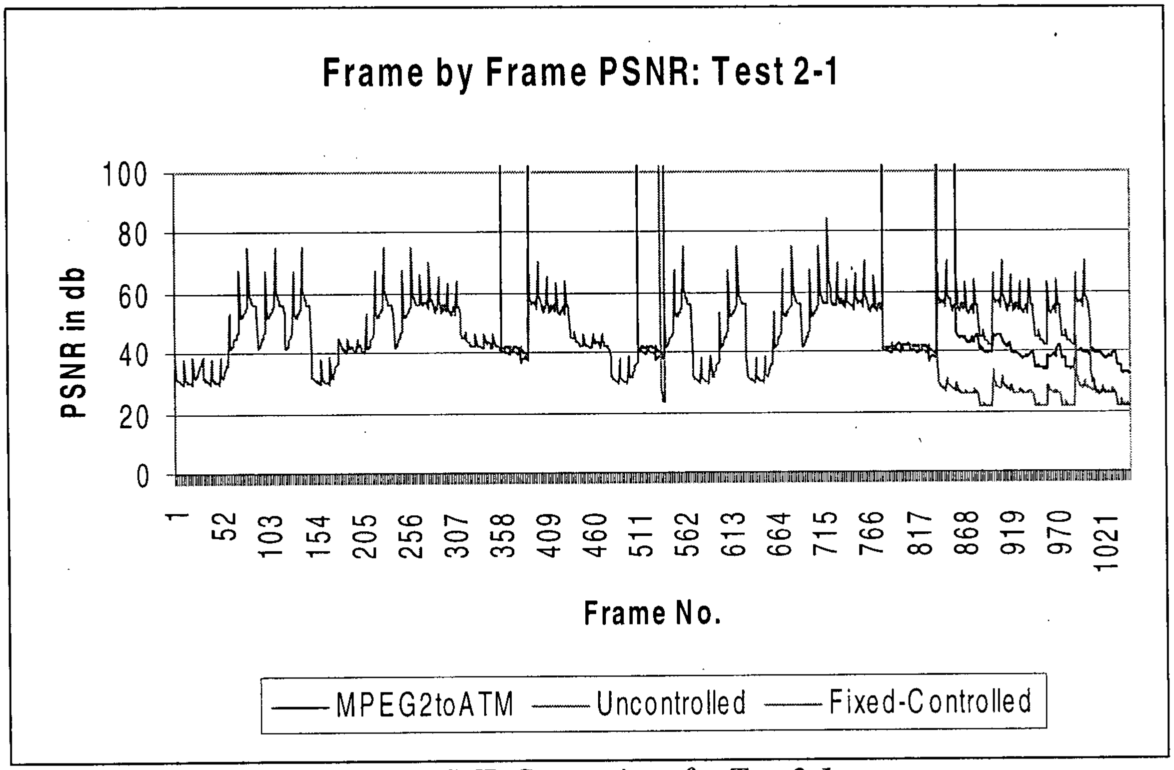

Figure A-7: Frame by Frame PSNR Comparison for Test 2-1

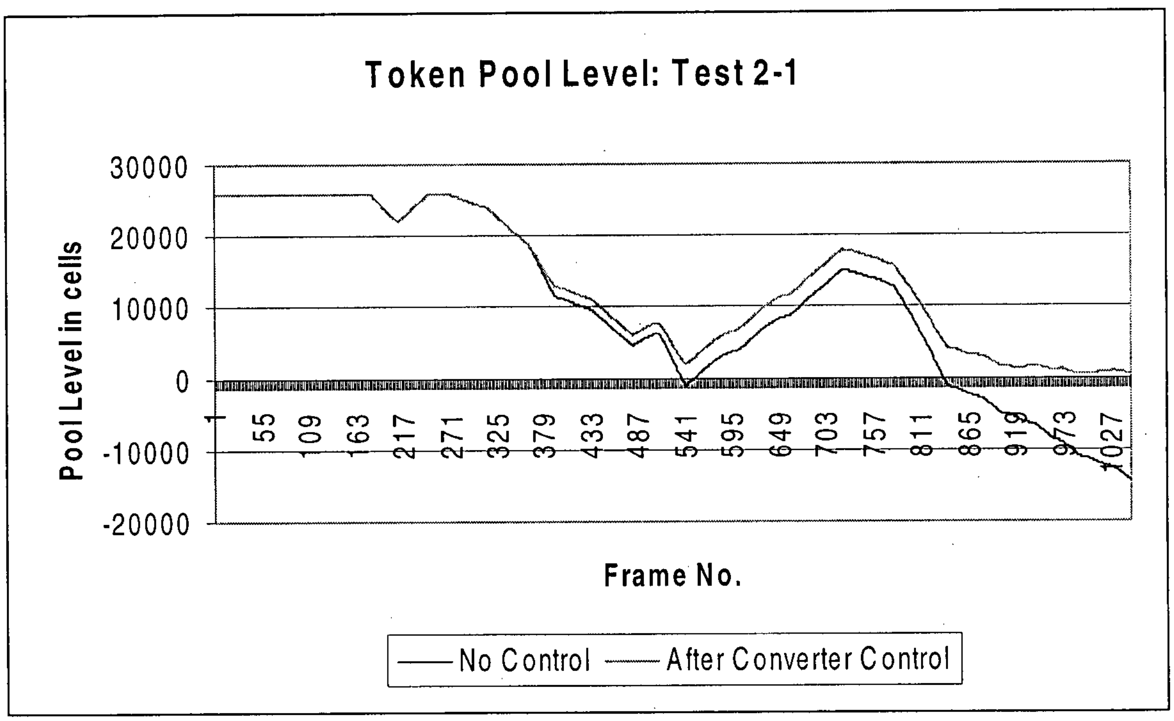

Figure A-8: Token Pool Levels Comparison of No Control and Converter Controlled for Test 2-1

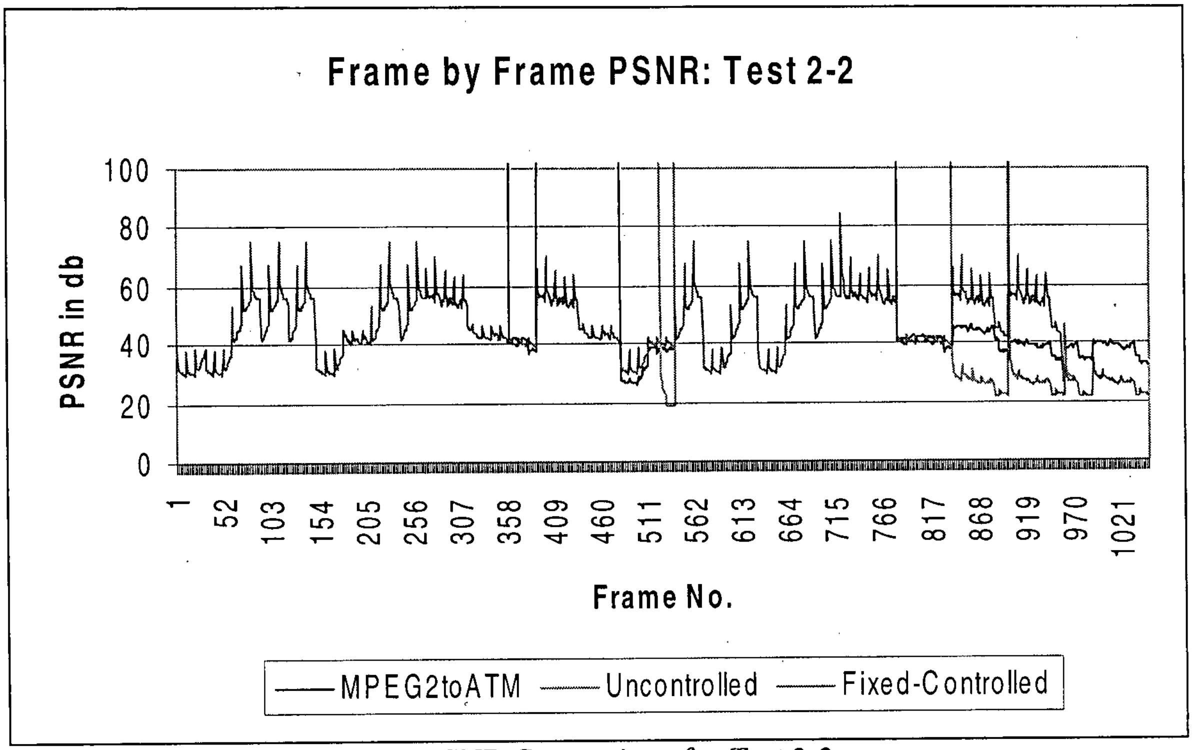

Figure A-9: Frame by Frame PSNR Comparison for Test 2-2

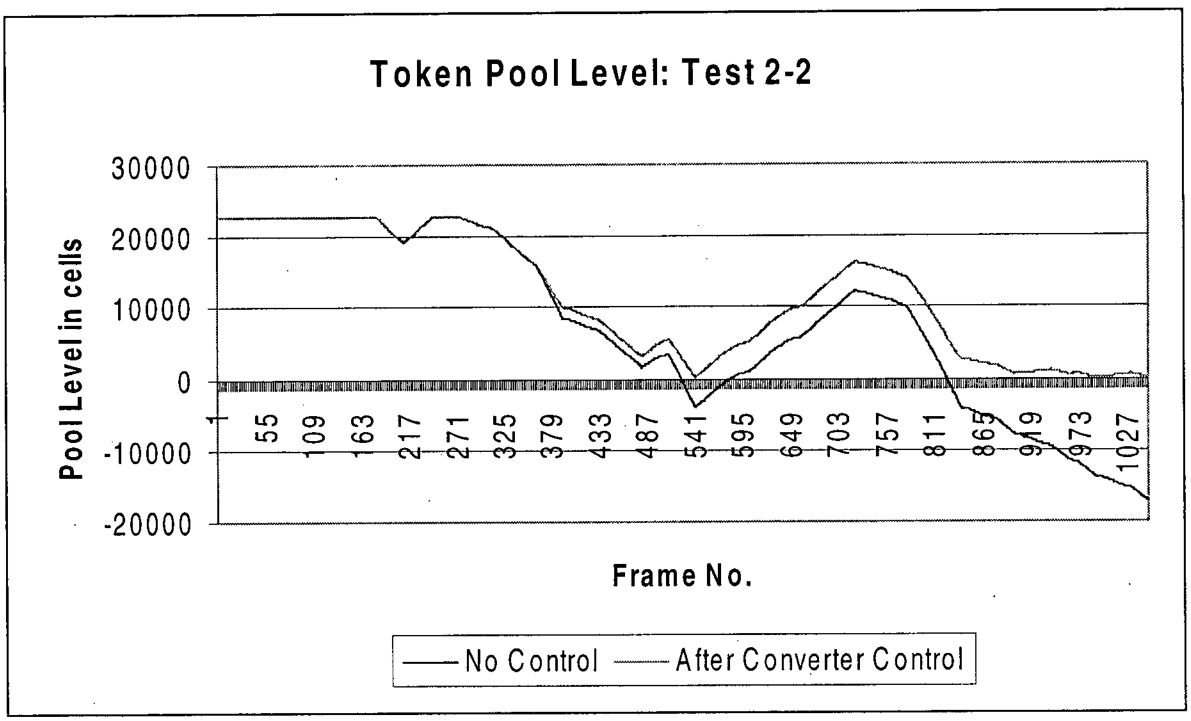

Figure A-10: Token Pool Levels Comparison of No Control and Converter Controlled for Test 2-2

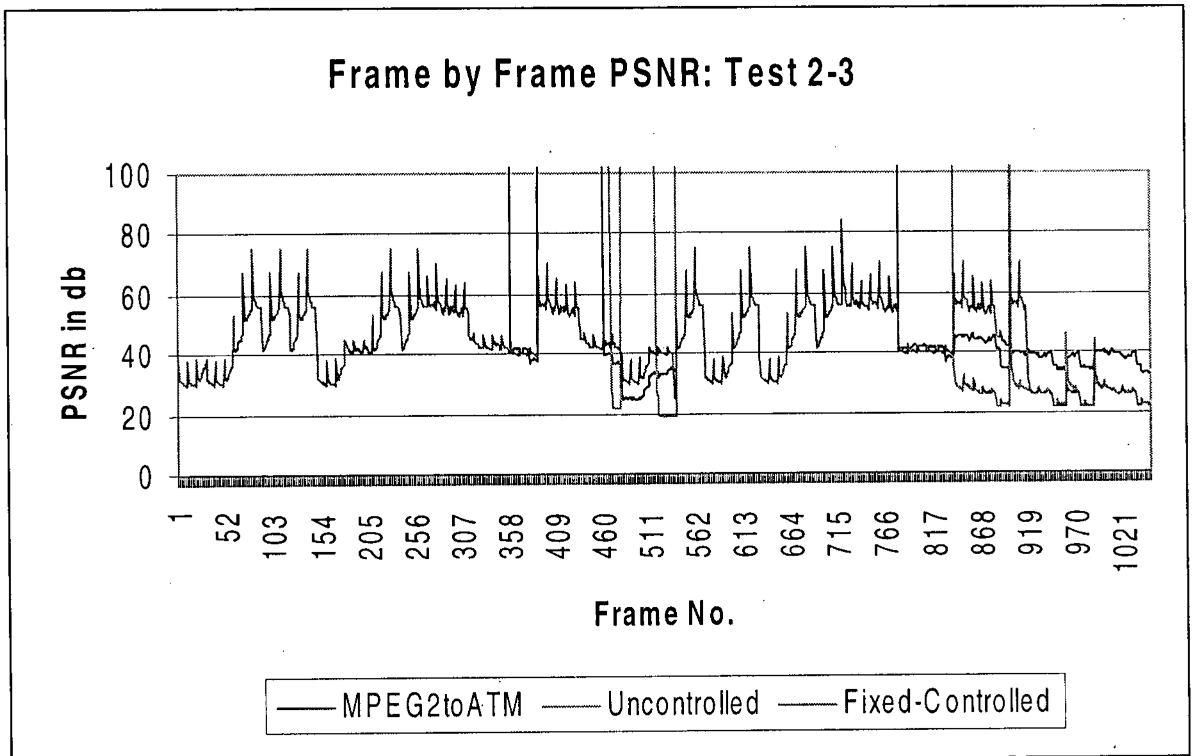

Figure A-11: Frame by Frame PSNR Comparison for Test 2-3

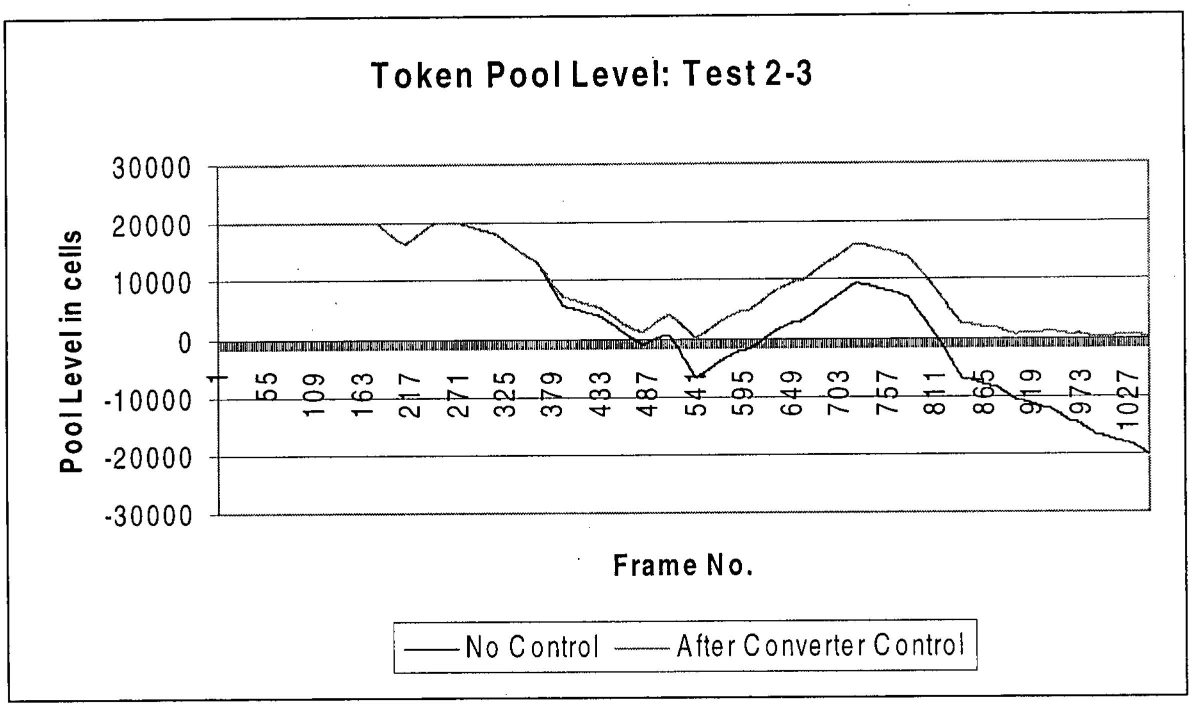

Figure A-12: Token Pool Levels Comparison of No Control and Converter Controlled for Test 2-3

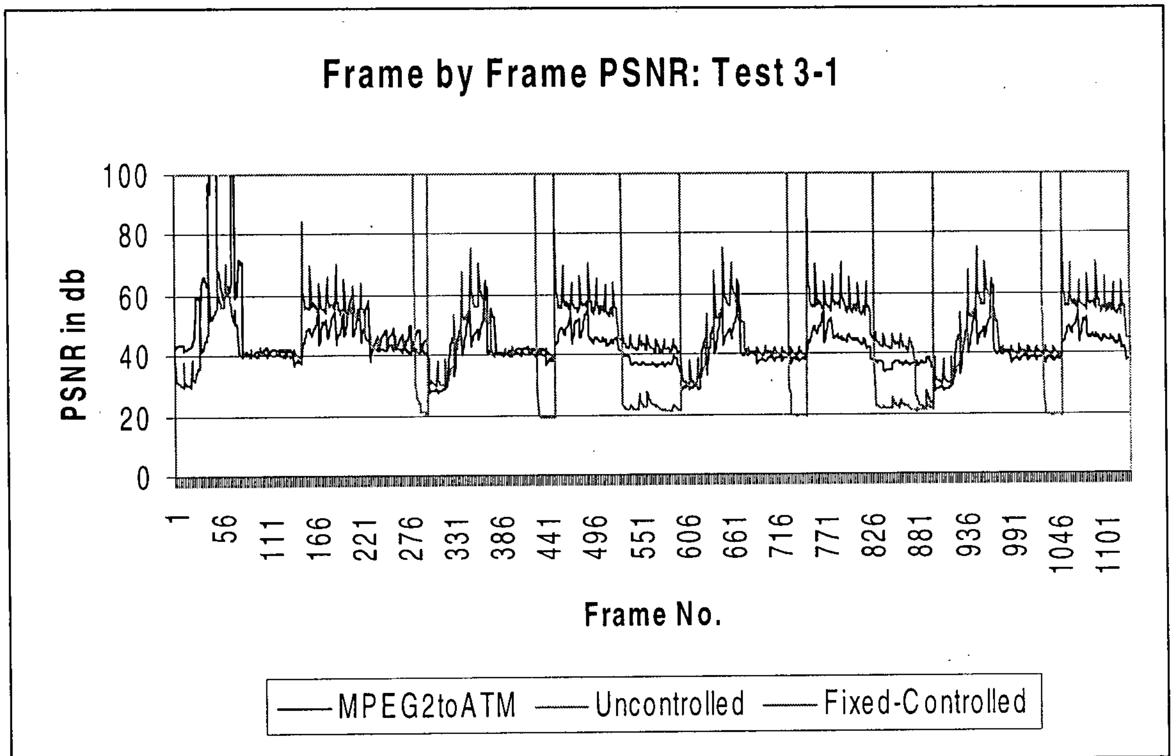

Figure A-13: Frame by Frame PSNR Comparison for Test 3-1

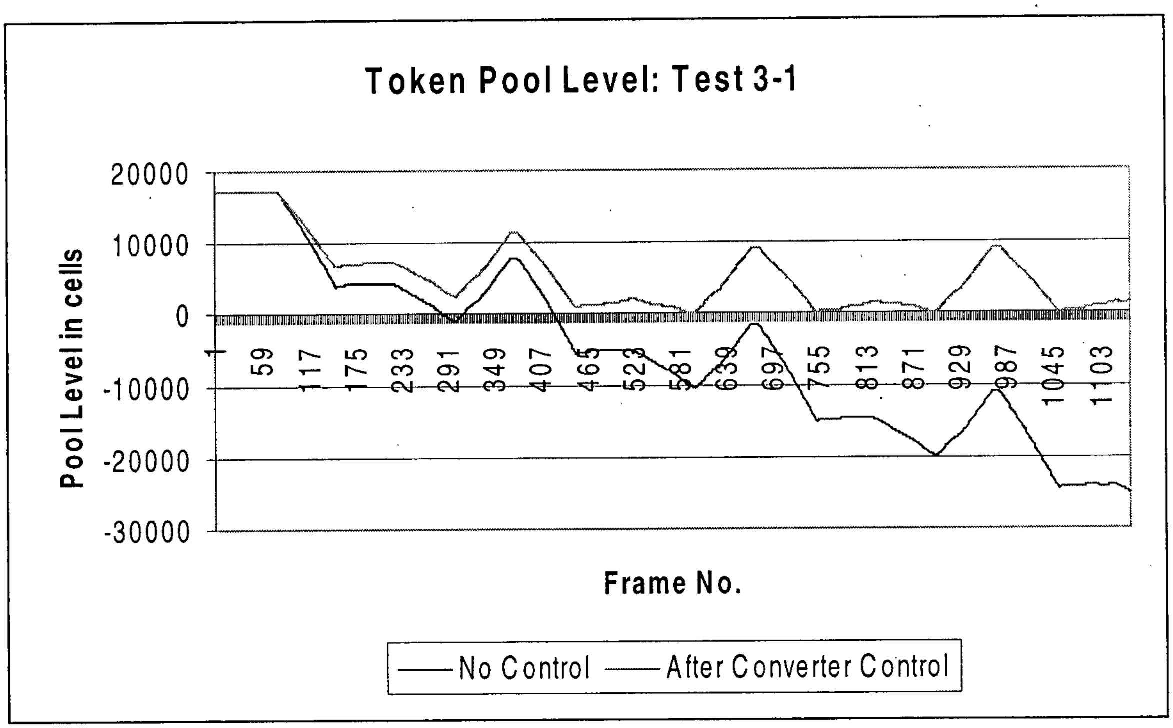

Figure A-14: Token Pool Levels Comparison of No Control and Converter Controlled for Test 3-1

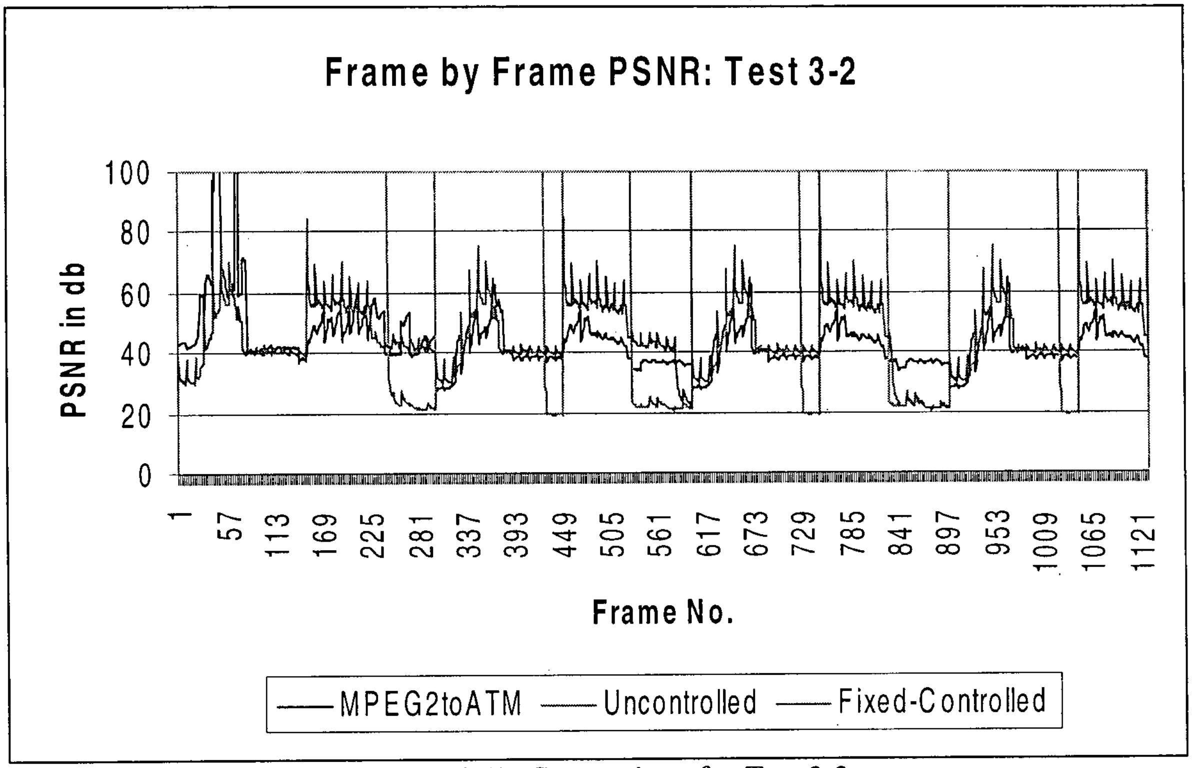

Figure A-15: Frame by Frame PSNR Comparison for Test 3-2

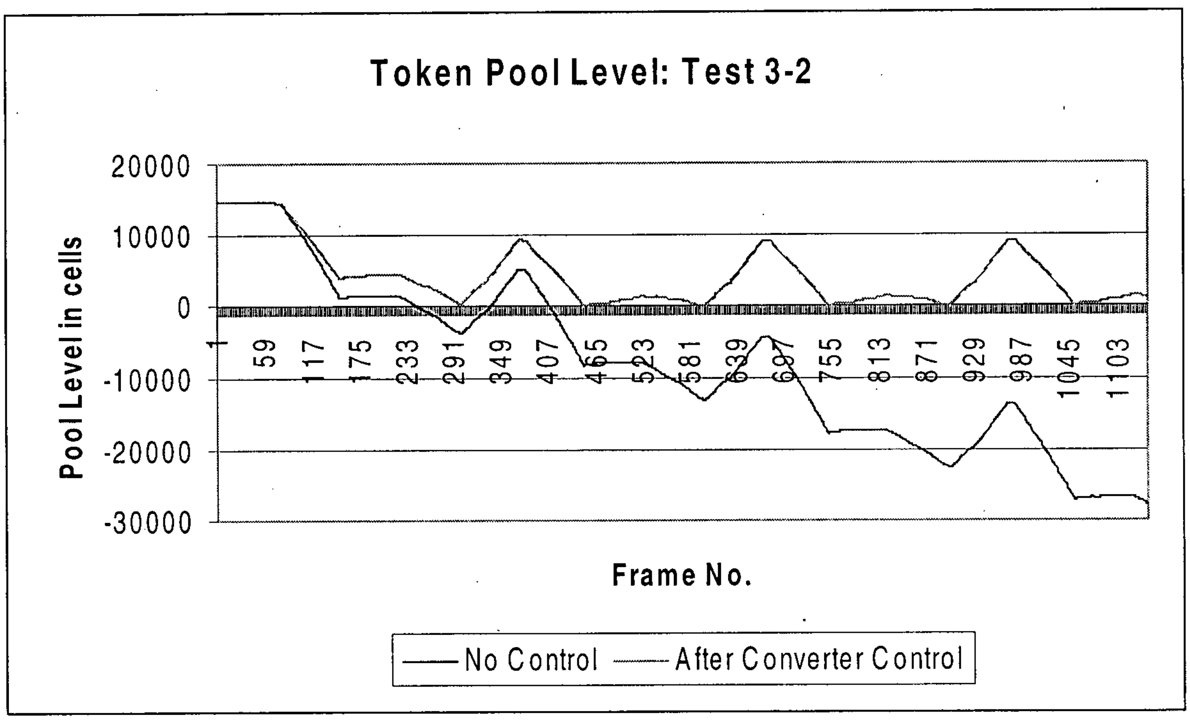

Figure A-16: Token Pool Levels Comparison of No Control and Converter Controlled for Test 3-2

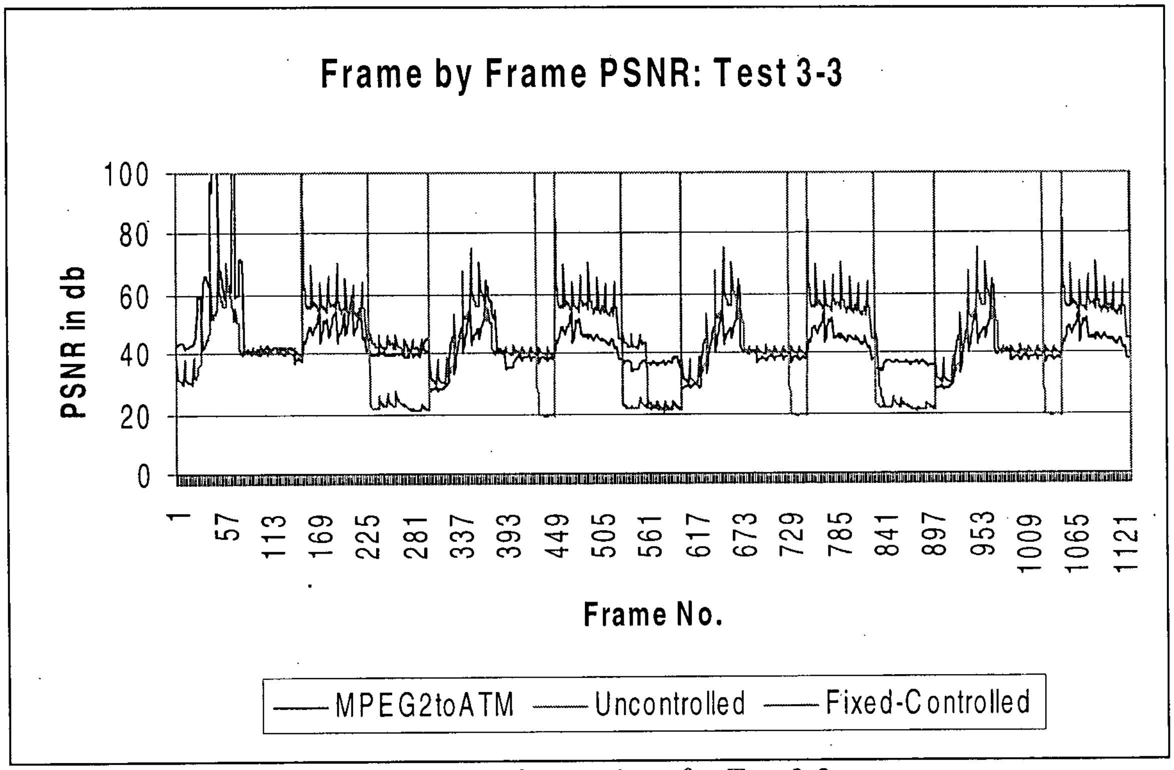

Figure A-17: Frame by Frame PSNR Comparison for Test 3-3

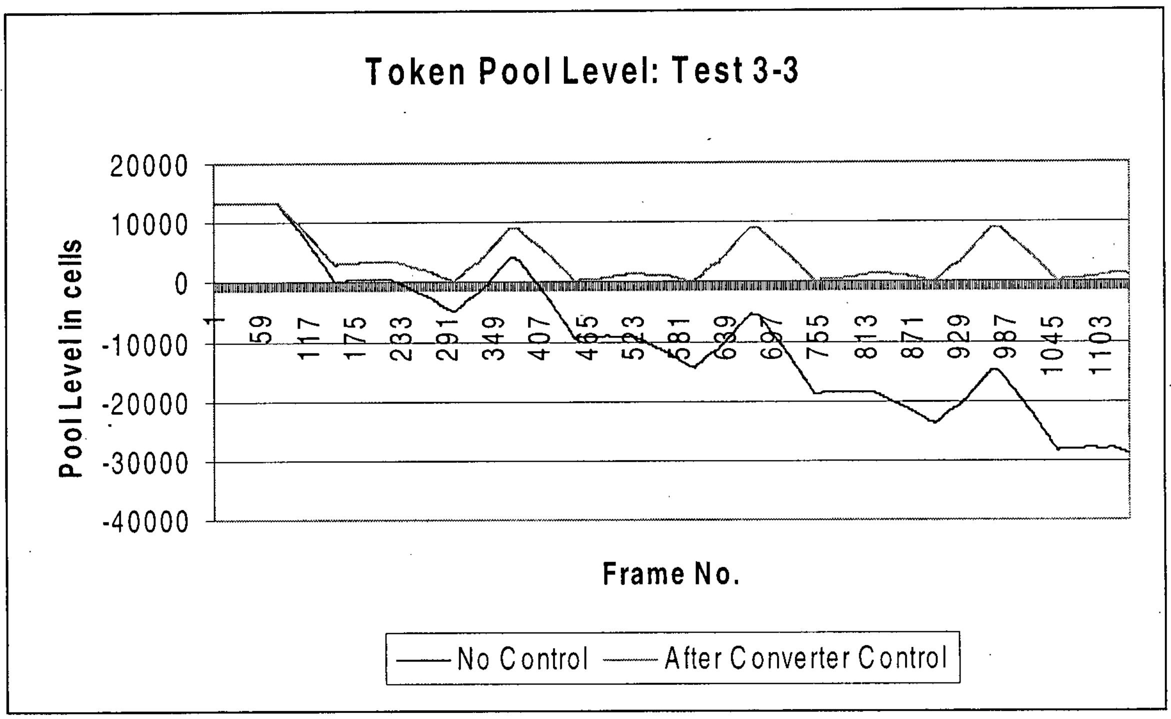

Figure A-18: Token Pool Levels Comparison of No Control and Converter Controlled for Test 3-3