Timestamping Schemes for MPEG-2 Systems Layer and Their Effect on Receiver Clock Recovery

Christos Tryfonas
Anujan Varma

UCSC-CRL-98-2
May 4, 1998

Board of Studies in Computer Engineering
University of California, Santa Cruz
Santa Cruz, CA 95064

ABSTRACT

We propose and analyze several strategies for performing timestamping of an MPEG-2 Transport Stream transmitted over a packet-switched network using the PCR-unaware encapsulation scheme, and analyze their effect on the quality of the recovered clock at the MPEG-2 Systems decoder. When the timestamping scheme is based on a timer with a fixed period, the PCR values in the packet stream may switch polarity deterministically, at a frequency determined by the timer period and the transport rate of the MPEG signal. This, in turn, can degrade the quality of the recovered clock at the receiver beyond acceptable limits. We consider three timestamping schemes for solving this problem: (i) selecting the deterministic timer period to avoid the phase difference in PCR values altogether, (ii) fine-tuning the deterministic timer period to maximize the frequency of PCR polarity changes, and (iii) selecting the timer period randomly to eliminate the deterministic PCR polarity changes. For the case of deterministic timer period, we derive the frequency of the PCR polarity changes as a function of the timer period and the transport rate, and use it to find ranges of the timer period for acceptable quality of the recovered clock. We also analyze a random timestamping procedure based on a random telegraph process and obtain lower bounds on the rate of PCR polarity changes such that the recovered clock does not violate the PAL/NTSC clock specifications. The analytical results are verified by simulations with both synthetic and actual MPEG-2 Transport Streams sent to an MPEG-2 Systems decoder.

Keywords: MPEG-2, clock recovery, ATM networks, set-top box, MPEG-2 Systems layer.

1 Introduction

MPEG-2 is the emerging standard for audio and video compression. By exploiting both spatial and temporal redundancies of the input signal, it achieves compression ratios of up to two orders of magnitude, enabling it to encode a video or audio signal to virtually any desired level of quality. The MPEG-2 Systems Layer specifies two mechanisms to multiplex elementary audio, video or private streams to form a *program*, namely the *MPEG-2 Program Stream* and the *MPEG-2 Transport Stream* formats. In jitter-prone environments such as a packet-switched network, the MPEG-2 Transport Stream is the preferred approach for transporting MPEG-2 streams. An MPEG-2 Transport Stream combines one or more programs into a single fixed-length packet stream. The use of explicit timestamps (called Program Clock References or PCRs in MPEG-2 terminology) within the packets facilitates the clock recovery at the decoder end ensures synchronization and continuity of MPEG-2 Transport Streams. For a brief tutorial of the MPEG-2 Systems Layer the interested reader is referred to [16, 20].

When transporting MPEG-2 encoded streams over packet-switched networks several issues must be taken into account. Some of these include the choice of the adaptation layer, method of encapsulation of MPEG-2 packets into network packets, provision of Quality-of-Service in the network to ensure control of delay and jitter, and the design of the decoder. At the decoder end, application-specific requirements such as accuracy and stability determine the approaches that should be taken to recover the system clock [9]. A certain category of applications uses the reconstructed system clock to directly synthesize a chroma sub-carrier for the composite video signal. The system clock, in this case, is used to derive the chroma sub-carrier, the pixel clock and the picture rate. The composite video sub-carrier must have at least sufficient accuracy and stability so that any normal television receiver's chroma sub-carrier PLL can lock to it, and the chroma signals which are demodulated using the recovered sub-carrier do not show any visible chrominance phase artifacts. There are often cases in which the application has to meet NTSC, PAL or SECAM specifications, which are even more stringent. As an example, NTSC requires a sub-carrier accuracy of 3 ppm with a maximum long-term drift of 0.1 Hz/sec. In such applications, a standard PLL is often used to recover the clock from the PCR timestamps transmitted within the stream.

The use of applications with stringent clock specifications requires careful design of the decoder since the decoder is responsible of feeding the TV set with a composite signal that meets the requirements. The demodulator in the TV set (Figure 1.1) has to extract clock information from this signal for the color sub-carrier regeneration process. The frequency requirements for NTSC specify a tolerance of ± 10 Hz (which corresponds to ± 3 ppm) [3]. The central sub-carrier frequency is 3.579545 MHz (or exactly $(63/88) \times 5$ MHz). The corresponding values for NTSC and other composite signals are summarized in Table 1.1. The requirements above define the precision of the oscillators for the modulator and thus, the minimum locking range for the PLL at the receiver end.

There are also requirements for the short- and long-term frequency variations. The maximum allowed short-term frequency variation for an NTSC signal is 56 Hz within a line (or 1 ns/64 μs) whereas the corresponding value for a PAL signal is 69 Hz. This corresponds to a variation of the color frequency of 16 ppm/line in both cases [2]. If this requirement is satisfied, we can obtain a correct color representation for each line.

The maximum long-term frequency variation (clock drift) that the composite NTSC or PAL signal must meet is 0.1 Hz/sec. The drift may be because of temperature changes at the signal generator and can be determined in an averaging manner over different time-window sizes. In fact,

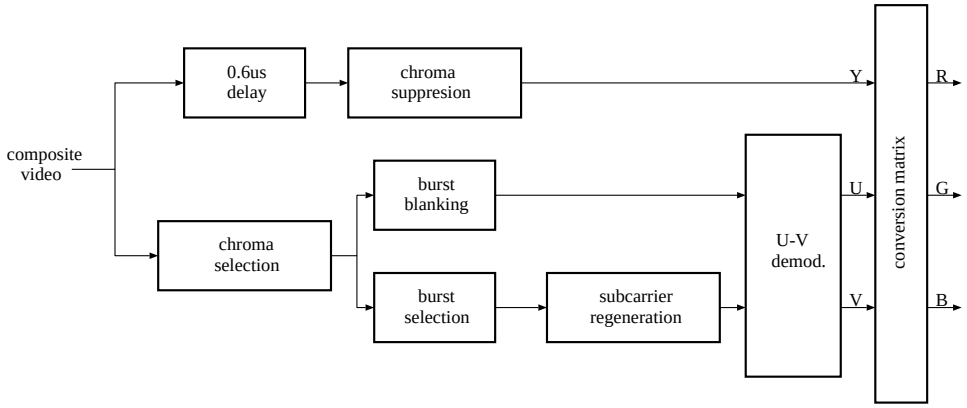

Figure 1.1: PAL/NTSC demodulator system block (from [2]).

Standard	NTSC	PAL (M)	PAL (B, D, G, H, N)
Frequency (Hz)	35795454	3575611.49	4433618.75
Tolerance (Hz)	±10	±10	±5
Tolerance (ppm)	±3	±3	±1

Table 1.1: Specifications for the color sub-carrier of different video formats (from [2]).

the actual requirement[1] for the color sub-carrier frequency (3.57 MHz ± 10 Hz for NTSC) is an average value which can be measured over any reasonable time period. Averaging intervals in the range from 0.1 sec to several seconds are common [6].

The degradation of the reconstructed clock at the receiver is caused primarily by the *packet delay variation (jitter)*. Three different causes contribute to the jitter experienced by an MPEG-2 transport stream as seen at the receiving end: The first is the frequency drift between the transmitter and the receiver clocks, which is usually small compared to the other two components. The second component of jitter is due to the packetization at the source, which may displace timestamp values within the stream. Finally, the network may introduce a significant amount of jitter, owing to the variations in queueing delays in the network switches. In this paper, our focus is in the second component, the packetization jitter.

The packet encapsulation procedure is the main cause for the packetization jitter. In the context of Asynchronous Transfer Mode (ATM) networks, two approaches have been proposed for encapsulation of MPEG-2 Transport Streams in ATM Adaptation Layer 5 (AAL5) packets: the *PCR-aware* and the *PCR-unaware* schemes [1]. In the PCR-aware scheme, packetization is done ensuring that when a Transport Stream packet contains a PCR value it will be the last packet encapsulated in an AAL-5 packet. This minimizes the jitter experienced by PCR values during packetization. In the PCR-unaware approach, the sender performs the encapsulation without checking whether PCR values are contained within a transport packet. Therefore, the encapsulation procedure may introduce significant jitter to the PCR values. The presence of jitter introduced by the adaptation

[1] All the requirements stated above have been defined for broadcasting applications. If the composite signal is to be displayed immediately, the requirements can be relaxed significantly. In fact, home receivers can handle a much wider range of frequencies and drift while ensuring good quality. However, the requirements are maintained in studio environments as good engineering practice.

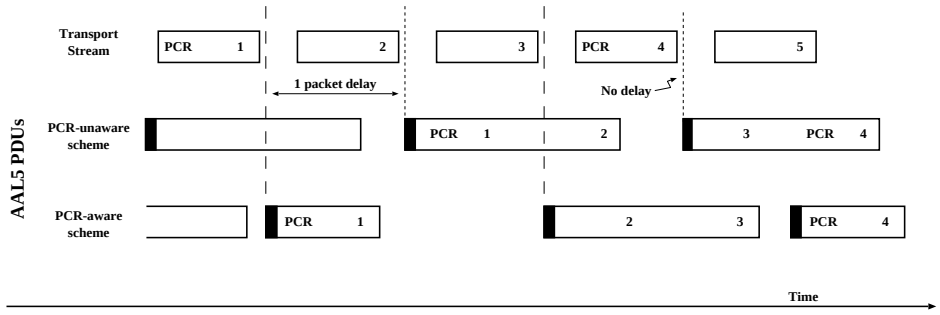

Figure 1.2: PCR packing schemes for AAL5 in ATM networks.

layer in this case, may distort the reconstructed clock at the MPEG-2 audio/video decoder. This, in turn, may degrade the quality when the synchronization signals for display of the video frames on the TV set are generated from the recovered clock.

The two schemes are illustrated in Figure 1.2. In the PCR-unaware case, the packetization procedure does not examine the incoming transport packets and therefore, the second AAL5 Protocol Data Unit (PDU) is the result of encapsulating transport packets 1 and 2, whereas the third AAL5 PDU results from the transport packets numbered 3 and 4. The PCR value in the second AAL5 PDU suffers a delay of one transport packet since it has to wait for the second transport packet to arrive before the PDU is formed. However, this is not the case for the third AAL5 PDU since the PDU becomes complete after the transport packet 4 arrives. On the other hand, the PCR-aware scheme completes a PDU if the current transport packet contains a PCR value. Thus, the second PDU is immediately formed as a result of transport packet 1 which contains a PCR value. The third PDU does not contain any PCR values since it contains transport packets 2 and 3. Finally, the fourth PDU is formed and completed by transport packet 4 in its payload without waiting to receive transport packet 5. It is evident that, for the PCR-unaware case, the process that inserts the PCR values into the MPEG-2 stream at the sender may introduce significant correlation on the resulting jitter of the outgoing transport packets containing PCR values. The PCR-unaware scheme is the recommended method of AAL encapsulation in the ATM Forum Video on Demand specification [19].

Several approaches have been proposed in the literature [1, 7, 8, 10, 18, 21] for the design of the MPEG-2 decoder to reduce the effects of jitter and provide acceptable quality for the decoded program. However, the impact of the timestamping process on the clock recovery at the decoder is not well understood. Insight on a few aspects of the timestamping process with a fixed-period timer, when transporting MPEG-2 over AAL5 using the PCR-unaware packing scheme, was first given by Akyildiz et al. [1]. In that work, it was shown that transport packets containing PCR values may switch indices (between odd and even) in a deterministic manner with a period that depends on both the transport rate and the timer period. This behavior was referred to as "pattern switch." This effect can be avoided by forcing all the PCR values to occupy the same phase in the encapsulated packet stream, or by compensating for the phase difference at the receiver.

In this paper, we analyze several strategies for performing timestamping of the outgoing MPEG-2 Transport Stream and analyze their effect on the clock recovery process of the MPEG-2 Systems decoder for applications with stringent clock requirements, assuming that the encapsulation procedure follows the PCR-unaware scheme. We start with an analysis of a timestamping process based on a timer with a fixed period. We derive the deterministic pattern switch frequency as a function of

the timer period and the transport rate, and use it to find ranges of the timer period for acceptable quality of the recovered clock. Next, we analyze a random timestamping procedure based on a *random telegraph* process [11] and obtain lower bounds on the rate of change of PCR polarity so that the recovered clock meets the PAL/NTSC specifications. Based on this analysis, we consider three timestamping schemes: (i) setting the deterministic timer period precisely to avoid the phase difference in PCR values altogether, (ii) fine-tuning the deterministic timer period to maximize the pattern switch frequency, and (iii) using a random distribution for the timer period to eliminate the deterministic pattern switch behavior. To verify our analytical results, we perform several simulation experiments with both synthetic and actual MPEG-2 Transport Streams sent to an MPEG-2 System decoder, over both a point-to-point link and a multi-hop ATM network. Our results show that, for certain ranges of timer periods, the low frequency of the pattern switch results in unacceptable quality of the reconstructed clock. This contradicts with the hypothesis of Akyildiz et al. [1], which states that reducing the frequency of the deterministic pattern switch improves the quality of the recovered clock. For deterministic timer periods, fine-tuning the timer to maximize the frequency of pattern switch results in the best quality of the receiver clock.

The rest of this paper is organized as follows: In Section 2, we analyze the effects of packetization jitter on the PAL/NTSC clock recovery process due to the PCR-unaware scheme, using a standard PLL in an MPEG-2 Systems decoder under both deterministic and probabilistic timestamping procedures. Based on this analysis, we propose a number of timestamping schemes for the sender so that the quality of the recovered clock at the decoder is acceptable in Section 3. In Section 4, we evaluate the schemes by simulation using a standard PLL employed in an MPEG-2 Systems decoder, with input from both synthetic MPEG-2 traces and actual MPEG-2 Transport Stream traces that are sent through a multi-hop ATM network. Finally, we conclude the paper in Section 5 with a summary of the results.

2 Analysis

In this section we provide an analysis of the effects of packetization jitter on the MPEG-2 decoder PLL. We first start with the modeling of the actual PLL and derive the transfer function to be used in the analysis. Next, we proceed to characterize the input signal at the PLL resulting from the timestamping and encapsulation schemes at the transmitter. We consider two distinct classes of timestamping schemes. In the first, timestamps are generated by a timer with a deterministic period; and in the second, the timer periods are drawn from a random distribution. In the first case, we derive the pattern switch frequency as a function of the timer period and transport rate of the MPEG-2 stream, which provides the phase of the input signal at the receiver PLL. In the second case, we use a random telegraph process to model the effect of the timestamping process, and use it to derive the variance of the recovered clock. This enables us to derive a lower bound on the required rate of change of PCR polarity in the packet stream to maintain the receiver PLL jitter within the specifications.

2.1 Derivation of Transfer Function of the Decoder PLL

We follow an approach similar to that in [12] for traditional PLLs. The main difference in our analysis is in the nature of the input signal. In our case, the input signal is a linear function as shown in Figure 2.1, whereas in the case of traditional PLLs, the input signal is considered to be a

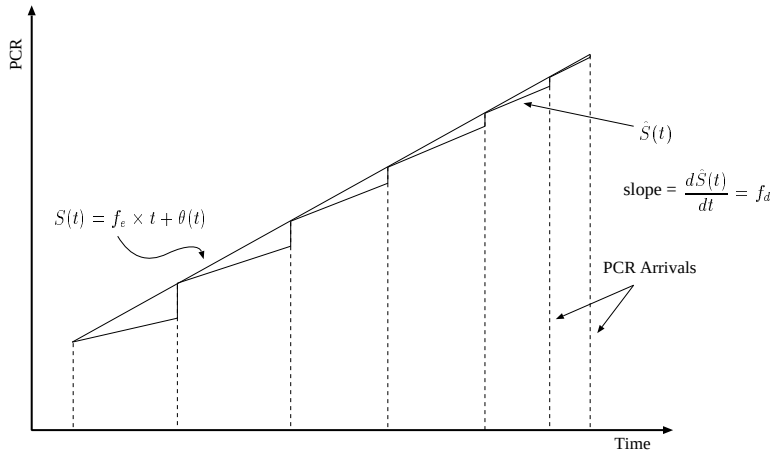

Figure 2.1: Actual PCR function and PCR function used in the analysis.

sinusoidal function. Before continuing with our analysis, we define some notations and assumptions used.

Although the PCRs arrive at discrete points in time, we can assume that the incoming PCRs form a continuous-time function $S(t)$ that is updated at the instants when a new PCR value is received. We can model the incoming clock with the function

$$S(t) = f_e \times t + \theta(t), \tag{2.1}$$

where f_e is the frequency of the encoder sending the MPEG-2 stream and $\theta(t)$ is the incoming clock's phase relative to a designated time origin. As indicated in Figure 2.1 there is a small discrepancy when modeling the incoming clock signal. The actual incoming clock signal $\hat{S}(t)$ is a function with discontinuities at the time instants at which PCR values are received, with slope equal to f_d for each of its segments, where f_d is the running frequency of the decoder. For the sake of convenience, however, we use $S(t)$ in place of the actual PCR function $\hat{S}(t)$, since the time between any two consecutive PCR arrivals is bounded by the MPEG-2 standard and equal to at most 0.1 second, which makes the two functions to be very close.

Analogously, the system time clock (STC) corresponds to the function

$$R(t) = f_d \times t + \hat{\theta}(t), \tag{2.2}$$

where $\hat{\theta}(t)$ is the incoming clock's phase relative to a designated time origin. Therefore, referring to model of the PLL in Figure 2.2, the error term after the subtractor is given by

$$e(t) = S(t) - R(t) = (f_e - f_d)t + \left(\theta(t) - \hat{\theta}(t)\right). \tag{2.3}$$

Without loss of generality, we can assume that $f_e = f_d$. Let us denote this with f_o and insert any frequency difference in the phase terms. We can now work with $\theta(t)$ as being the input to our control system and with $\hat{\theta}(t)$ as being the output of the counter as shown in Figure 2.2. Thus, Eq. (2.3) becomes

$$e(t) = \theta(t) - \hat{\theta}(t). \tag{2.4}$$

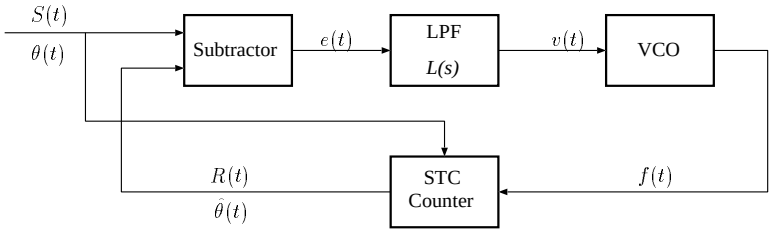

Figure 2.2: Equivalent model of the PLL used.

The frequency $f(t)$ of the VCO is a function of $v(t)$. The nominal value of this frequency is assumed to be f_0 and when $v(t)$ is applied, it becomes $f_0 + Kv(t)$ where K is the gain factor of the VCO. It is obvious that

$$\frac{dR(t)}{dt} = f_0 + Kv(t). \tag{2.5}$$

By definition,

$$R(t) = f_0 t + \hat{\theta}(t). \tag{2.6}$$

Combining Eq. (2.5) and (2.6) we get

$$\frac{d\hat{\theta}(t)}{dt} = Kv(t). \tag{2.7}$$

From Eq. (2.4) and (2.7) we obtain

$$\begin{aligned} \frac{de(t)}{dt} &= \frac{d\theta(t)}{dt} - Kv(t) \\ &= \frac{d\theta(t)}{dt} - K\int_0^\infty l(t-u)e(u)du. \end{aligned} \tag{2.8}$$

We assume that the Laplace transformations of $e(t)$ and $\theta(t)$ exist and they are $E(s)$ and $\Theta(s)$ respectively, and $L(s)$ is the low-pass filter's transfer function. Eq. (2.8), when transformed to the Laplace domain, becomes

$$sE(s) = s\Theta(s) - KL(s)E(s). \tag{2.9}$$

We assume that $\hat{\theta}(t)$ has a Laplace transform. Using $E(s) = \Theta(s) - \hat{\Theta}(s)$, where $\hat{\Theta}(s)$ is the Laplace transform of $\hat{\theta}(t)$, and Eq. (2.9) we can now derive the transfer function $H(s)$ of the closed-loop

$$H(s) = \frac{\hat{\Theta}(s)}{\Theta(s)} = \frac{KL(s)}{s + KL(s)}. \tag{2.10}$$

The Laplace transform $F(s)$ of the recovered frequency function $f(t)$ is given by

$$\begin{aligned} F(s) &= \left(\Theta(s) - \hat{\Theta}(s)\right) L(s)K \\ &= \left(1 - H(s)\right)\Theta(s)L(s)K \\ &= K\left(1 - \frac{KL(s)}{s + KL(s)}\right) L(s)\Theta(s) \qquad (2.11) \\ &= P(s)\Theta(s), \qquad (2.12) \end{aligned}$$

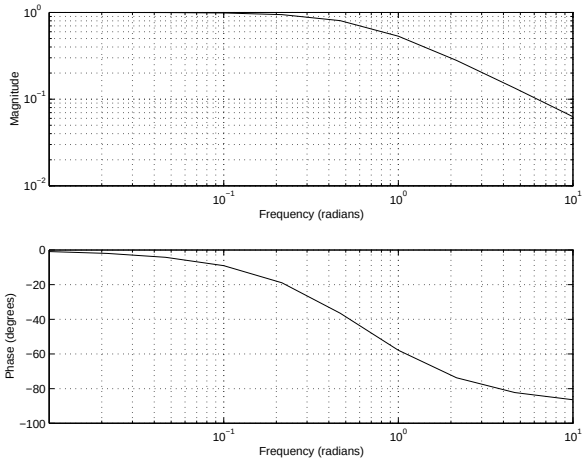

Figure 2.3: Frequency response of the first-order Butterworth LPF used in the analysis.

where $P(s)$ is given by

$$P(s) = K\left(1 - \frac{KL(s)}{s + KL(s)}\right) L(s). \tag{2.13}$$

In order to model the LPF, we used a first order Butterworth analog LPF with a cutoff frequency of 0.1 Hz. This cutoff frequency was derived from both the experimental results of [20] and the analysis done in [10]. Following the methodology presented in [15] we get

$$L(s) = \frac{0.2\pi}{s - s_1}, \tag{2.14}$$

with s_1 being the single real pole of the filter that has a value of

$$\begin{aligned} s_1 &= 2\pi 0.1 \exp(j\pi) \\ &= 0.2\pi \exp(j\pi) \\ &= -0.2\pi. \end{aligned} \tag{2.15}$$

Hence, from Eq. (2.14) and (2.15) we obtain

$$L(s) = \frac{0.2\pi}{s + 0.2\pi}. \tag{2.16}$$

The frequency response of the filter is shown in Figure 2.3.

2.2 Possible Input Processes due to PCR-Unaware Scheme

We now study the effect of possible input processes, resulting from the timestamping procedure at the source and the PCR-unaware scheme, on the clock recovery process at the decoder. Since we are interested in the tracking performance of the PLL, we can assume that the PLL is locked before the input process is applied as the input function $\theta(t)$ of the PLL.

Under the PCR-unaware scheme, an AAL packet containing two MPEG-2 transport packets may contain a PCR value either in the first or in the second MPEG-2 transport packet. Therefore, a PCR value will suffer either zero or one transport packet delay at the destination. Assuming that the PLL

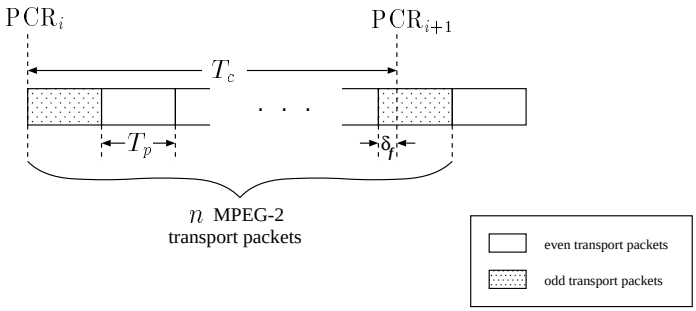

Figure 2.4: Illustration of forward drift of PCR values' packet boundaries when $0 \leq \tau < T_p/2$.

is locked before the input process is applied, the resulting phase difference values at its input $\theta(t)$ will be approximately $\pm f_o/2r$, where f_o is the central frequency in MPEG-2 Systems layer and r is the rate of the MPEG-2 transport stream in packets/second. We analyze two cases: a deterministic case in which a timer with a fixed-period is used to perform the timestamping procedure and a probabilistic case in which the PCR values are placed randomly in the MPEG-2 Transport Stream according to a random telegraph process.

Deterministic Case

When a timer with a constant period is used at the source to timestamp the MPEG-2 packet stream, the positions of the PCR values switch between even and odd boundaries in the AAL packets at a constant frequency. This effect was first observed by Akyildiz et al. [1], who termed it "pattern switch". In this section, we systematically derive the pattern switch frequency as a function of the timer period and the transport rate of the MPEG stream.

Let T_p denote the inter-arrival time of MPEG-2 transport packets, and T_c the period of the timer at the transmitter. Since $T_c > T_p$, we can express T_c in terms of T_p as

$$T_c = 2kT_p + \tau, \tag{2.17}$$

where k is a non-negative integer and $0 \leq \tau < 2T_p$. Since, in general, T_c is not an exact multiple of T_p, the actual time instants at which the PCR values are inserted into the MPEG-2 Transport Stream will drift relative to packet boundaries. More specifically, we need to consider three cases corresponding to different ranges of τ.

Case 1: $0 \leq \tau < T_p/2$**.** In this case, a forward drift of the resulting packet boundaries of the associated PCR values can be identified as illustrated in Figure 2.4. Let n denotes the integer number of transport packets included in a T_c period, that is,

$$n = \left\lceil \frac{T_c}{T_p} \right\rceil. \tag{2.18}$$

Let δ_f denote the forward drift, given by

$$\delta_f = T_c - (n-1)T_p. \tag{2.19}$$

The last equation can be easily derived by observing Figure 2.4.

From Eq. (2.18) and (2.19) we obtain

$$\begin{aligned} \delta_f &= T_c - (n-1)T_p \\ &= T_c - \left(\left\lceil \frac{T_c}{T_p} \right\rceil - 1 \right) T_p. \end{aligned} \tag{2.20}$$

As becomes evident from Eq. (2.20), that the number of continuous PCR packets falling into odd or even positions in the MPEG-2 transport stream n_c, is given by

$$\left\lfloor \frac{T_p}{\delta_f} \right\rfloor \leq n_c \leq \left\lceil \frac{T_p}{\delta_f} \right\rceil. \tag{2.21}$$

Thus, the polarity (even/odd) of timestamp values in the packet stream exhibits a square-wave pattern at the input of MPEG-2 decoder's PLL with a period of $2n_cT_c$ and peak-to-peak amplitude of f_oT_p. Therefore the phase of the input signal at the PLL is given by

$$\theta(t) = f_oT_p \sum_{\lambda=-\infty}^{\infty} \left(u(t + \frac{\lambda n_c T_c}{2}) - u(t - \frac{\lambda n_c T_c}{2}) \right) - \frac{f_oT_p}{2}, \tag{2.22}$$

in which $u(t)$ is the unit-step function. If the frequency of the above input signal becomes less than the bandwidth of the PLL, the output of the PLL will follow the pulse with a resulting degradation of the quality of the recovered clock. If we assume that the PLL has a perfect LPF of bandwidth B_L, then the period of $\theta(t)$ should be less than $1/B_L$. That is,

$$2n_cT_c < \frac{1}{B_L}. \tag{2.23}$$

Case 2: $T_p/2 \leq \tau < 3T_p/2$**.** In this case, at most two consecutive PCR values may fall into odd- or even-numbered MPEG-2 transport packets. In the specific case that $\tau = T_p$, the PCR values fall in alternate odd- and even-indexed transport packets producing the maximum frequency of changes in timestamp position in the packet stream. The resulting process has high-frequency components which are filtered by the decoder PLL and are unlikely to affect the quality of the recovered clock. This is verified in our simulation experiments of Section 4.

Case 3: $3T_p/2 \leq \tau < 2T_p$**.** This is similar to the first case, except that the drift of the packet boundaries of the PCR values is in the backward direction (Figure 2.5). In this case, let δ_b denote the backward drift, given by

$$\begin{aligned} \delta_b &= nT_p - T_c \\ &= \left\lceil \frac{T_c}{T_p} \right\rceil T_p - T_c. \end{aligned} \tag{2.24}$$

Again, the last equation can easily be derived from Figure 2.5. Similarly, the number of continuous PCR packets falling into only odd or only even positions in the MPEG-2 transport stream n_c, is bounded by the following inequality

$$\left\lfloor \frac{T_p}{\delta_b} \right\rfloor \leq n_c \leq \left\lceil \frac{T_p}{\delta_b} \right\rceil. \tag{2.25}$$

The resulting phase at the input of the PLL, in this case also, is a square-wave with a period of $2n_cT_c$ and peak-to-peak amplitude of f_oT_p. Therefore the input function at the PLL is again given by

$$\theta(t) = f_oT_p \sum_{\lambda=-\infty}^{\infty} \left(u(t + \frac{\lambda n_c T_c}{2}) - u(t - \frac{\lambda n_c T_c}{2}) \right) - \frac{f_oT_p}{2}, \tag{2.26}$$

with $u(t)$ being the unit-step function.

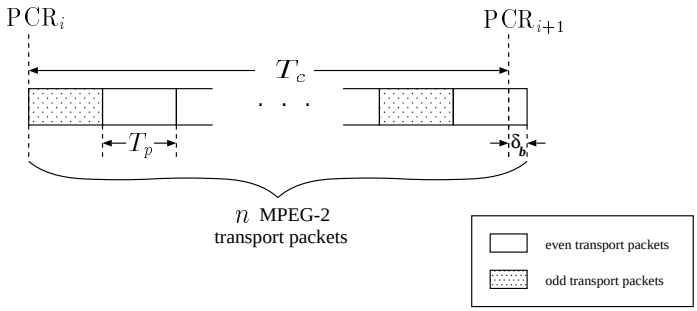

Figure 2.5: Illustration of backward drift of PCR values' packet boundaries when $3T_p/2 \leq \tau < 2T_p$.

Probabilistic Case

The generation of MPEG-2 Transport Streams with variable inter-PCR delay can be done by randomizing the timestamping procedure according to some distribution. In the probabilistic case, we assume that the PCR values fall completely in random places in the MPEG-2 Transport Stream. Without loss of generality, we can assume that they have the same probability of being in odd or even-indexed transport packets (as by using Bernoulli trials). For the sake of convenience, we analyze this behavior by modeling the input phase $\theta(t)$ as a random telegraph process [11].

The objective of our analysis is to obtain the variance or the actual function that describes the recovered clock, i.e., $f(t)$. We derive the variance of the recovered clock in the case that the sequence of $\pm f_o/2r$ values forms a scaled random telegraph process. The random telegraph process $T(t)$ is a random process that assumes values of ± 1, has a mean of zero, and is stationary or cyclostationary. Assuming that initially $T(0) = \pm 1$ with equal probability, $T(t)$ is generated by changing polarity with each occurrence of an event of a Poisson process of rate a. In our analysis, we use a scaled version of the random telegraph process $T(t)$ in which the process gets the values $\pm f_o/2r$. We refer to the scaled version by $T_s(t)$. A sample realization of this process is shown in Figure 2.6.

We first obtain the statistic measures of the scaled random telegraph process $T_s(t)$. Since the mean of the random telegraph process is zero, the mean of the scaled version is also zero. Let us now derive the autocorrelation function of $T_s(t)$.

$$\begin{aligned} R_{T_s}(t_1,t_2) &= E\left[T_s(t_1)T_s(t_2)\right], \quad \forall \text{ pair of real numbers } t_1, t_2 \\ &= \left(\frac{f_o}{2r}\right)^2 \text{Prob}[T_s(t_1) = T_s(t_2)] - \left(\frac{f_o}{2r}\right)^2 \text{Prob}[T_s(t_1) \neq T_s(t_2)] \\ &= \left(\frac{f_o}{2r}\right)^2 \frac{(1+e^{-2a|t_2-t_1|})}{2} - \left(\frac{f_o}{2r}\right)^2 \frac{(1-e^{-2a|t_2-t_1|})}{2} \\ &= \frac{f_o^2}{4r^2} e^{-2a|t_2-t_1|}. \end{aligned} \tag{2.27}$$

We also obtain the *power spectral density (psd)* of the input process. The psd function of the input process is given as the Fourier transform of the autocorrelation function $R_{T_s}(k)$. Thus,

$$S_{T_s}(w) = \int_{-\infty}^{\infty} R_{T_s}(\tau) e^{-jw\tau} d\tau$$

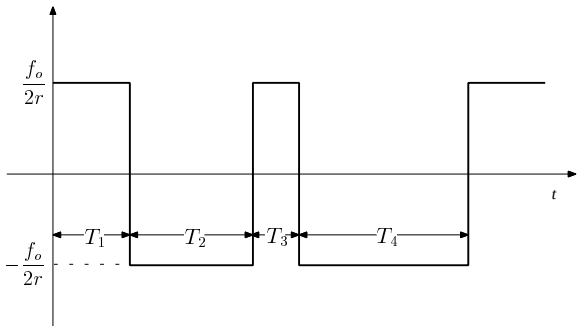

Figure 2.6: Sample realization of $T_s(t)$ process. T_i's are independent identically-distributed (i.i.d.) exponential random variables.

$$\begin{aligned} &= \int_{-\infty}^{\infty} \frac{f_o^2}{4r^2} e^{-2a|\tau|} e^{-jw\tau} d\tau \\ &= \frac{f_o^2}{4r^2} \left(\int_{-\infty}^{0} e^{2a\tau} e^{-jw\tau} d\tau + \int_{0}^{\infty} e^{-2a\tau} e^{-jw\tau} d\tau \right) \\ &= \frac{f_o^2}{4r^2} \left(\frac{1}{2a - jw} + \frac{1}{2a + jw} \right) \\ &= \frac{a f_o^2}{r^2 (4a^2 + w^2)}. \end{aligned} \tag{2.28}$$

The psd function of the recovered clock is given by

$$S_f(w) = S_{T_s}(w) |P(w)|^2, \tag{2.29}$$

where $|P(w)|$ is the magnitude of the Fourier transform of the function defined in Eq. (2.13). We can obtain the Fourier transform of the signal that has Laplace transform of $P(s)$ by substituting s with jw. After substitution, and using Eq. (2.16) we get

$$P(w) = \frac{Kw}{w + \left(\frac{w^2}{0.2\pi} - K\right) j}. \tag{2.30}$$

The square of the magnitude of $P(w)$ is given by

$$\begin{aligned} |P(w)|^2 &= P(w) P^*(w) \\ &= \left(\frac{Kw}{w + \left(\frac{w^2}{0.2\pi} - K\right) j} \right) \left(\frac{Kw}{w - \left(\frac{w^2}{0.2\pi} - K\right) j} \right) \\ &= \frac{(Kw)^2}{w^2 + \left(\frac{w^2}{0.2\pi} - K\right)^2}, \end{aligned} \tag{2.31}$$

where $P^*(w)$ is the conjugate of $P(w)$.

The variance of the output process is determined by the inverse Fourier transform of $S_f(w)$ at point zero. That is,

$$\begin{aligned} \sigma^2_{f(t)} &= \mathcal{F}\{S_f(w)\}|_0 \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} S_f(w) dw \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\frac{a f_o^2}{r^2 (4a^2 + w^2)} \right) \left(\frac{(Kw)^2}{w^2 + \left(\frac{w^2}{0.2\pi} - K\right)^2} \right) dw \end{aligned}$$

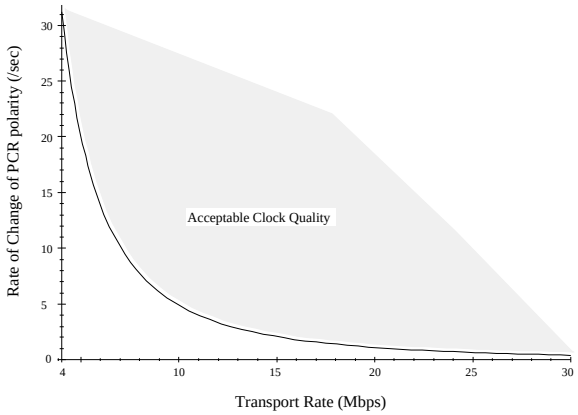

Figure 2.7: Minimum rate of change of PCR polarity to maintain acceptable clock quality as a function of the transport rate of the MPEG-2 stream under PAL.

$$= \frac{a f_o^2 \pi K^2}{r^2(40a^2 + 4\pi a + 2\pi K)}. \tag{2.32}$$

The above equation gives the variance of the clock at the MPEG-2 Systems decoder. The clock of the color sub-carrier is derived from this clock using a scaling factor which is different for PAL and NTSC. Since the scaled random telegraph process $T_s(t)$ is bounded, we can assume that the recovered clock deviates from its central frequency by at most $\sigma_{f(t)}$. Applying the requirements for the sub-carrier frequency shown in Table 1.1 we get

$$\sigma_{f_{\text{PAL}}} = \frac{A_{\text{PAL}} f_o K}{r} \sqrt{\frac{a\pi}{40a^2 + 4\pi a + 2\pi K}} \leq 5, \tag{2.33}$$

for the recovered PAL sub-carrier frequency and

$$\sigma_{f_{\text{NTSC}}} = \frac{A_{\text{NTSC}} f_o K}{r} \sqrt{\frac{a\pi}{40a^2 + 4\pi a + 2\pi K}} \leq 10, \tag{2.34}$$

for the recovered NTSC sub-carrier frequency with A_{PAL} and A_{NTSC} being equal to $\frac{4.43}{27}$ and $\frac{3.57}{27}$ respectively. We can now obtain a lower bound for the allowed rate r in packets/second in the case of PAL and NTSC sub-carriers, so that the clock remains within the specifications.

$$r \geq \frac{A_{\text{PAL}} f_o K}{5} \sqrt{\frac{a\pi}{40a^2 + 4\pi a + 2\pi K}}, \tag{2.35}$$

for the PAL case and

$$r \geq \frac{A_{\text{NTSC}} f_o K}{10} \sqrt{\frac{a\pi}{40a^2 + 4\pi a + 2\pi K}}, \tag{2.36}$$

for the NTSC case. Analogously, we can derive a bound on the rate of change of PCR polarity a so that the clock specifications are not violated under a specific transport rate.

$$a \geq \frac{-\pi(4r^2 - X^2) + \sqrt{(4\pi r^2 - \pi X^2)^2 - 320\pi K r^4}}{80r^2}, \tag{2.37}$$

where X is equal to $\frac{A_{\text{NTSC}} f_o K}{10}$ for the NTSC case and $\frac{A_{\text{PAL}} f_o K}{5}$ for the PAL case.

For the sake of convenience, we concentrate on the PAL case only and compute the numerical value for the minimum rate for a typical MPEG-2 decoder PLL. The constant K is used to scale the

input signal to the appropriate levels for the MPEG-2 frequency. More specifically, the design of the VCO takes into account the maximum difference in ticks of a 27 MHz clock when the jitter or the PCR inaccuracy due to remultiplexing operations is at its maximum allowed value, and the limits of the frequency of the decoder. Since, according to MPEG-2 standard [9], the maximum jitter expected is around ± 0.5 ms, the maximum allowable difference is 13500 ticks. For this maximum difference, the decoder must operate at a frequency within the limits specified in the MPEG-2 standard. That is,

$$27 \text{ MHz - } 810 \text{ Hz} \leq \text{decoder clock frequency} \leq 27 \text{ MHz} + 810 \text{ Hz}. \tag{2.38}$$

Therefore, the selection of K should be around the value of $\frac{810}{13500}$ or 0.06 in order for the decoder to operate correctly. We can also assume that a is greater or equal to one which corresponds to an underlying Poisson process that has a minimum average rate of one arrival every second. Then, the minimum transport rate for the stream, in order not to have any PAL clock violations, is approximately 19.476 Mbps. A similar result can be found for the NTSC case. Higher values for a produce even lower bounds. For example, a value of 20 gives a bound of approximately 4.970 Mbps. Figure 2.7 shows a plot of a versus the MPEG-2 transport rate, where the the shaded region corresponds to the range of a and transport rate for the reconstructed clock to stay within the PAL specifications.

3 Solutions for Providing Acceptable Clock Quality

In the last section, we analyzed and quantified the effect of the timestamping process at the transmitter on the quality of the recovered clock at the receiver. When the timer-period for timestamping is chosen deterministically, the pattern switch behavior may manifest itself as a periodic square-wave signal at the input of the decoder PLL in the MPEG-2 Systems Layer. One option to avoid the effect of this pattern switch signal is to eliminate it altogether by forcing all PCR values to occupy the same phase in the AAL packet stream. This would make the receiver clock quality under the PCR-unaware scheme identical to that under the PCR-aware scheme. A second alternative is to maximize the pattern switch frequency by causing the PCR values to switch between odd and even positions in the packet stream at the maximum rate. Finally, a third alternative is to use a random timestamping interval to avoid the deterministic pattern switch behavior. In this section, we discuss the tradeoffs associated with each of these approaches:

The MPEG standard [9] specifies a maximum interval of 0.1 seconds for transmission of timestamps in the MPEG-2 transport stream. Therefore, in all the schemes we consider below, we assume that the timestamping interval T_c is always chosen within this bound.

Scheme 1: Forcing PCR values to stay on one side: The best case in the timestamp process is when the timer period is selected such that the transport rate of the MPEG stream is an exact multiple of the timestamping rate, that is, the ratio T_c/T_p is an integer. In this case, the PCR values will always fall in either the odd-numbered or the even-numbered transport packets, thus eliminating packetization jitter altogether. Hence, the quality of the recovered clock is similar to that under the PCR-aware case. In practice, however, it is difficult to maintain the timestamping interval precisely as a multiple of the transport period, because of oscillator tolerances and various quantization effects. These effects may cause the PCR timestamp values to switch polarity at a very low frequency in the packet stream, degrading the quality of the recovered clock over the long term. In addition, loss of packets containing

PCR values may cause timestamps to change polarity, that is, an odd-indexed PCR packet may become even-indexed or vice-versa. These effects are studied in the simulation experiments of Section 4.

Scheme 2: Forcing PCRs to change boundary at high frequency: From the analysis of the previous section, it is clear that the maximum frequency of changes in timestamp position in the packet stream occurs when the timestamping interval T_c satisfies the equality

$$T_c = (2k+1)T_p, \tag{3.1}$$

where T_p is the transport period of the signal and k is any non-negative integer. If T_c can be chosen precisely to satisfy this inequality, the timestamped transport packets will occupy alternate (even/odd) positions in the AAL packet stream. The resulting pattern-switch signal is a square wave with the maximum possible frequency among all possible choices of T_c in the range from $2kT_p$ to $2(k+1)T_p$.

Just as in the previous scheme, it is difficult to set T_c precisely to satisfy Eq. (3.1). However, in this case it is not necessary to maintain T_c precisely. In the light of the analysis in the previous section, if the value of the timer period falls in the interval $[(2k+1/2)T_p, (2k+3/2)T_p]$, the frequency of the resulting pattern-switch pulse is still close to the case when $T_c = (2k+1)T_p$. This allows some tolerance for the clocks. Another significant advantage of this scheme is that random losses of packets containing timestamps are unlikely to affect the quality of the reconstructed clock. These hypotheses are verified in our simulation experiments in the next section.

Scheme 3: Random setting of timer period: In this case, the period of the timestamping timer is set to an arbitrary value, chosen randomly. The same timestamping interval is chosen for the entire packet stream, resulting in a deterministic pattern-switch signal at the input of the receiver.

From our analysis of the previous section, the frequency of the pattern switch signal depends on the relative magnitudes of T_c and T_p. Thus, this scheme needs to be used only when the transport rate of the MPEG signal is not known, since a more intelligent choice can be made when T_c is known.

Scheme 4: Random timer period: Another alternative when the transport rate is not known is to randomize the timestamping interval, by setting the timer each time to a value drawn from a random distribution. In the previous section, we showed that adequate quality can be maintained for the receiver clock when the timestamping interval is chosen such that the resulting PCR polarity changes in the packet stream exceeds a minimum rate. Although the analysis was based on modeling the PCR polarity changes with a random telegraph process, in practice similar results can be obtained by choosing the timer period from an exponential distribution. Results from our simulation experiments in the next section indicate that an exponentially-distributed timer period results in almost the same quality for the recovered clock as compared to the case when the PCR polarity changes according to the random telegraph process.

Similar to Scheme 2, this solution does not suffer from degradation of clock quality in the presence of random packet losses. Thus, Scheme 4 is useful when the transport rate T_p is not known with adequate precision.

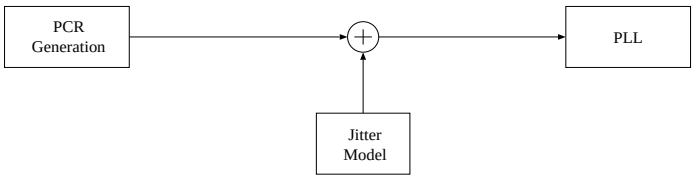

Figure 4.1: Simulation model used in the first set of experiments to study the impact of PCR insertion rate on the receiver clock.

In summary, Scheme 2 is the preferred scheme when the transport rate of the MPEG signal is known precisely, while Scheme 4 may be used when the transport rate is not known. In the next section, we evaluate the four schemes using both synthetic and real MPEG-2 traces to investigate the characteristics of the recovered clock signal at the receiver under various conditions.

4 Experimental Results

To evaluate the timestamping schemes discussed in the last section, we conducted a number of simulation experiments with both synthetic and real MPEG-2 traces. In this section, we describe the simulation experiments and compare the schemes based on the quality of the recovered clock at the receiver.

Our experiments in this section fall into two sets: the first set is based on a synthetic trace in which the PCR timestamps were generated using the particular timestamping scheme being evaluated, and the second based on actual MPEG-2 traces which were pre-processed by inserting PCR timestamps based on the various timestamping schemes. In all cases, the traces were fed to an MPEG-2 Systems decoder and the behavior of the reconstructed clock was observed. In the second set of experiments, the packet streams were transported over an ATM network and the PCR-unaware scheme was used for encapsulation at the Adaptation Layer. Finally, to study how the jitter generated by the network interacts with the packetization jitter introduced by the timestamping process, we conducted simulations in which the MPEG-2 streams were transported over a multi-hop ATM network with significant amount of cross-traffic.

4.1 Experiments with Synthetic Traces

We first describe the models used and then present results from the simulations.

PCR Generation and Jitter Models

Since our interest is only in studying the effect of the timestamp distribution in the MPEG-2 stream on the behavior of the recovered clock, we constructed a model of the timestamp generation process and used it to drive a model of the MPEG-2 Systems decoder. The PCR insertion frequency was varied based on the specific timestamping scheme simulated, with a lower bound of 10 Hz to meet the MPEG-2 specifications [9]. The transport rate was set to 4 Mbps, which results in an inter-packet time T_p of $188 * 8/(4 \times 10^6) = 376$ μseconds. Thus, the maximum packetization jitter under the PCR-unaware scheme is 376 μs.

The model used in the first set of experiments is illustrated in Figure 4.1. Two different approaches can be used to model the packetization jitter resulting from the timestamping process: in

the first, packets are delayed according to the jitter model before they are delivered to the decoder. This is the traditional way of generating jitter in which the jitter changes the actual arrival times of the packets. Another way of modeling jitter is to assume that packets are delivered with equal spacing, and simulate the jitter by changing the PCR values in the packets once they are received by the decoder. Although the two methods are not equivalent, the convergence time and the steady-state error of the receiver PLL are almost the same in both cases [2]. For the sake of convenience, we chose the second method to model jitter. In our experiments, the jitter is added directly to the incoming PCR values at the decoder, in terms of the number of ticks of a 27 MHz clock.

Deterministic Timer Period

We first used a timer with a deterministic period to timestamp the MPEG-2 transport packets, and ran several experiments for various choices of the timer period. the results are summarized in Figures 4.3 and 4.2. Recall that the transport rate is 4 Mbps, corresponding to an inter-packet time of $T_p = 376$ μs. We set the timestamping timer period T_c close to or exactly at the transport packet boundaries to examine the three cases considered in Section 2.

In the first case, the timer period was chosen very close, but not exactly equal to, a multiple of the inter-packet time T_p. The value chosen was $T_c = 50T_p + 10^{-7}$, which corresponds to a timer frequency of $f_c = 1/T_c = 53.1912$ Hz. According to the analysis in Section 2, the PCR timestamps in the resulting trace exhibit a forward drift of $\delta_f = 10^{-7}$ seconds. This, in turn, causes the PCR timestamp positions to switch at intervals of $2n_cT_c$, where $n_c = T_p/\delta_f$, that is, approximately 141.4 seconds. Thus the packetization delay experienced by transport packets containing PCR values form a square wave as illustrated in Figure 4.2.

In the second case, we set the timer period T_c precisely to $(50 + 1)T_p$, or $f_c = 52.1485$ Hz. According to the analysis of Section 2, this maximizes the frequency of PCR polarity changes in the packet stream, for the range of $50T_p \leq T_c \leq 52T_p$. Thus, the frequency of PCR polarity changes in $f_c/2 = 26.0743$ Hz.

In the third case, T_c was set slightly lower than an integer multiple of the inter-packet time. by choosing $T_c = 52T_p - 10^{-7}$, or $f_c = 51.1459$ Hz. This results in a backward drift of $\delta_b = 10^{-7}$ seconds for the timestamp values. The delay distribution resulting from this choice is identical to that shown in Figure 4.2.

Assuming the MPEG-2 stream carried NTSC video, Figure 4.3 shows the frequency of the recovered NTSC clock at the receiver, in terms of deviation from the ideal 3.57 MHz value. When the low frequency drift is present in the timestamp values, that is, in the first and third cases above, the frequency of the recovered clock does not meet the maximum allowed variation of ± 10 Hz in NTSC. In the second case, however, the recovered clock stayed well within the specifications.

For completeness, we also simulated two additional cases: the first was with a timer period of exactly $50T_p$ ($f_c = 53.1914$ Hz), which forces all the PCR timestamp values to stay on one side of the packet boundary. As seen from Figure 4.3, this resulted in a perfect clock at the receiver. In the second case, we increased the timestamping rate almost by a factor of 10, but still maintaining the drift in timestamp values. This did not improve the clock behavior in Figure 4.3, asserting that the solution to the jitter problem lies in fine-tuning the timestamping frequency, rather than choosing a high frequency of timestamping.

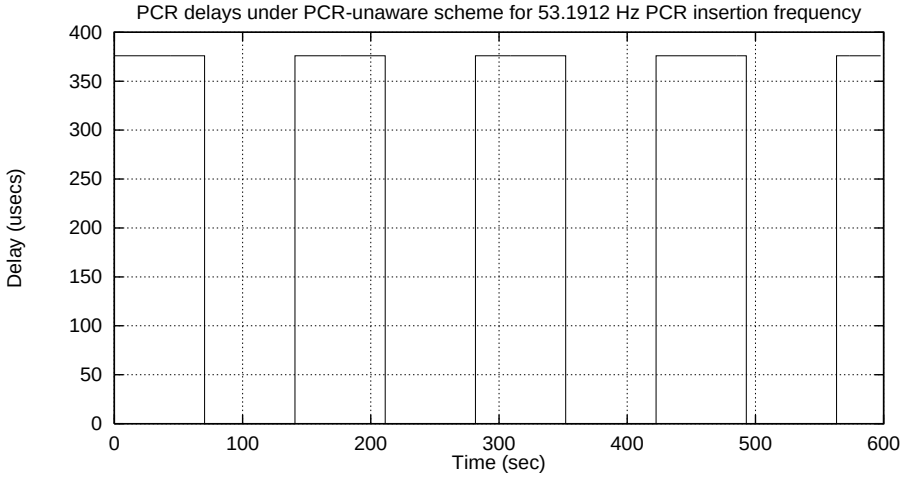

Figure 4.2: Delays of transport packets containing PCR values due to packetization jitter with a PCR insertion rate of 53.1912 Hz.

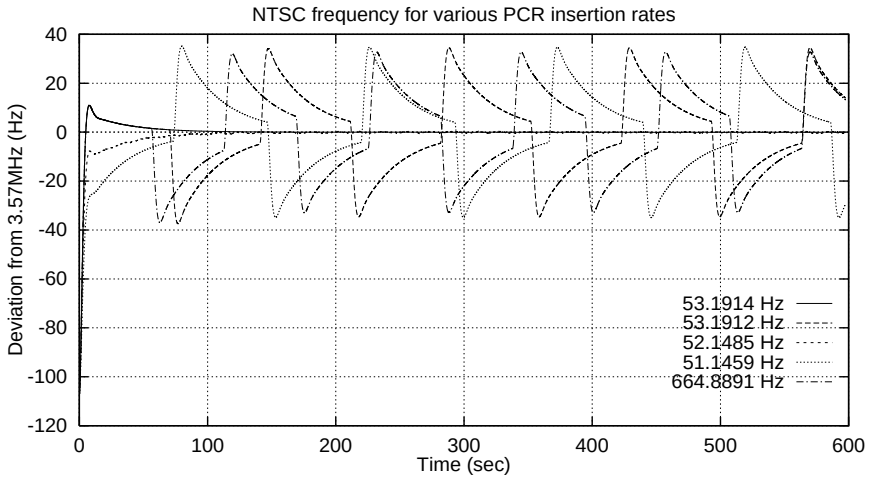

Figure 4.3: NTSC color sub-carrier generation frequency for varying PCR insertion frequencies. Transport rate is 4 Mbps.

Experiments with Random Timestamping Intervals

We also conducted several experiments with the synthetic trace where the timestamping intervals were chosen randomly. First, to validate the results for the random telegraph process in Section 2, we generated a trace in which the PCR values switched polarity alternately, at intervals that are exponentially distributed. The frequency of timestamping, that is, the rate of the random telegraph process a was varied to study its effect on the quality of the recovered clock at the receiver. We started from a rate of 10 per second (the minimum specified rate in MPEG-2), and increased it to 20 and 40.

The model based on random telegraph process enabled us to characterize the effect of timestamp jitter on the behavior of the recovered clock, but may not adequately describe the timestamp distribution resulting from a timer with exponentially distributed period. Generating timestamp distributions according to the random telegraph process may not be practical in a real system, as

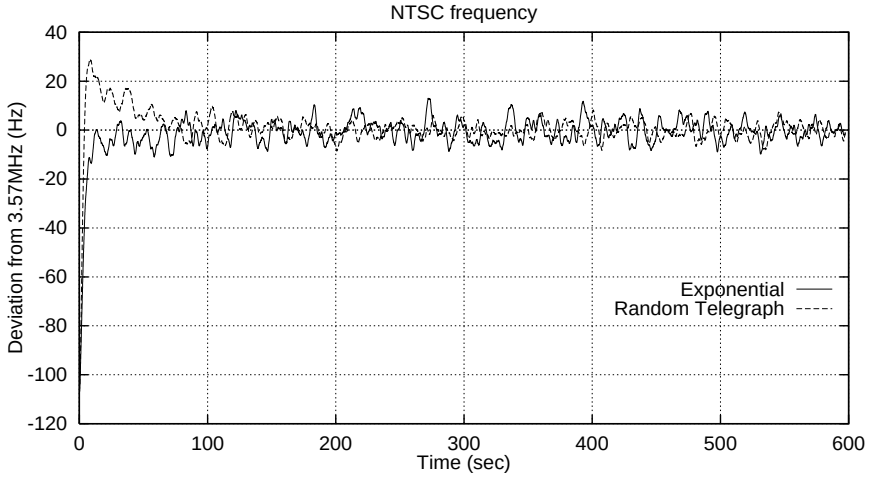

Figure 4.4: NTSC color sub-carrier generation frequency for both exponential distribution and the random telegraph process. The average rate for the exponential timestamping process is 10 PCRs/sec, same as the rate for the random telegraph process.

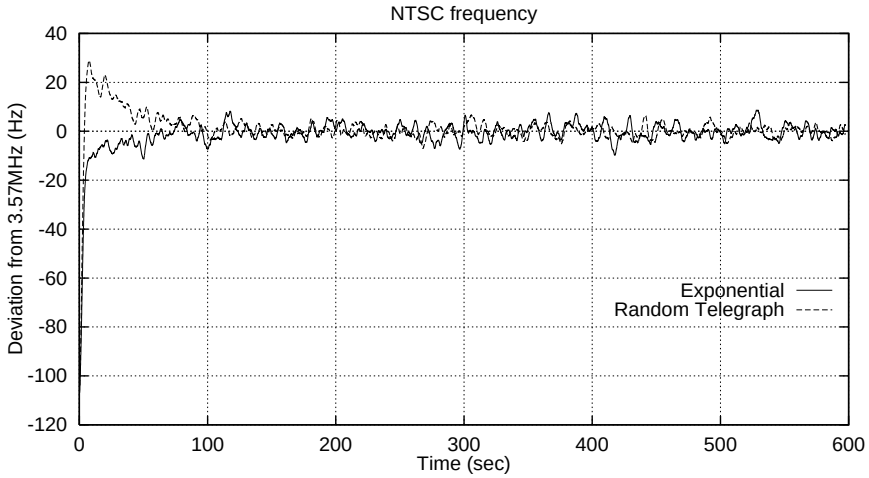

Figure 4.5: NTSC color sub-carrier generation frequency for both exponential distribution and the random telegraph process. The average rate for the exponential timestamping process is 20 PCRs/sec, same as the rate for the random telegraph process.

it requires the transmitter to switch positions in the packet stream with each inserted timestamp. When an exponentially-distributed timer period is used to generate timestamps, the polarity of the timestamps may change at a much lower rate than the timestamping rate. To study the clock quality with exponentially distributed timer periods, we performed the same experiments discussed in the last paragraph with a timestamping process driven by an exponential distribution of the timestamping interval. The same timestamping rates of 10, 20, and 40 were chosen, so as to compare them with the case of random telegraph process.

The results from these experiments are shown in Figures 4.4, 4.5 and 4.6, for the timestamping rates of 10, 20 and 40, respectively. With a timestamping rate of 10 per second, the clock signal obtained from exponentially distributed timer periods slightly violated the NTSC specifications

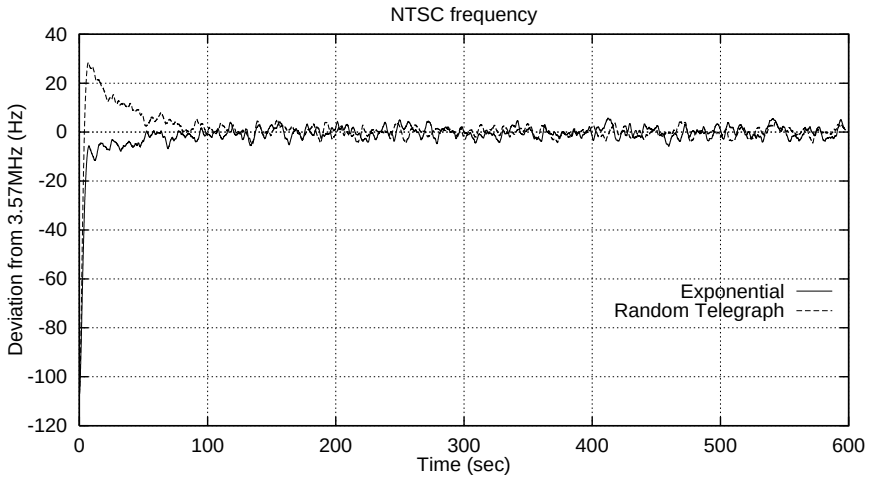

Figure 4.6: NTSC color sub-carrier generation frequency for both exponential distribution and the random telegraph process. The average rate for the exponential timestamping process is 40 PCRs/sec, same as the rate for the random telegraph process.

(±10 Hz), while the timestamp distribution driven by the random telegraph process provided adequate quality. This can be considered an anomaly. When the timestamping rate was increased to 20 and 40, both schemes performed very close and maintained the receiver clock well within the specified ±10 Hz tolerance.

In summary, the results from these experiments enable us to conclude that a random timestamping process with an adequately large timestamping rate can provide the desired quality for the recovered clock. An interesting observation from these experiments is that, in general, a higher rate of the exponential distribution (smaller timer period) results in better quality of the reconstructed clock, which may not be true in the case when the timer period is chosen deterministically. The reason is that the random timestamp distributions resulting from the former avoids the low-frequency patterns caused by the latter.

4.2 Simulations with Real MPEG-2 Traces in an ATM Network

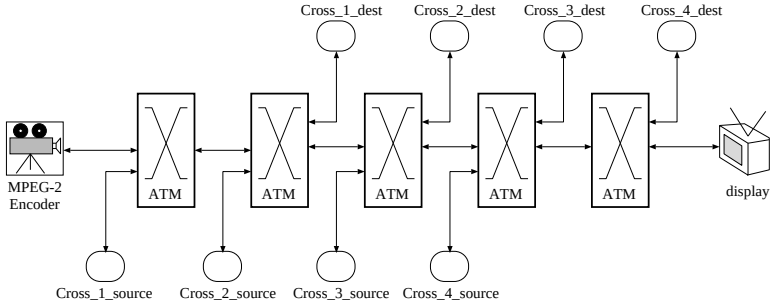

Figure 4.7: Network topology used in the experiment.

To further validate our results, we performed a number of simulation experiments with real MPEG-2 Transport Stream traces. To simulate the effect of the timestamp distribution on the clock recovery process, the PCR values were inserted in the traces using a timer with tunable frequency.

Three traces were used throughout the experiments; in the first one, all the transport packets were inserted into even-numbered packets (referred to as "One Side" in the experiments). In the second trace, the PCR values were added so as to introduce a small forward drift when encapsulated using the PCR-unaware scheme (referred to as "Forward Drift" in the experiments). Finally, in the third trace, the transport packets containing PCR timestamps fall into alternating odd and even indices (referred to as "Max. Freq." in the experiments). The oscillator used for the timer was simulated with a 2 ppm frequency variation, to model the effect of clock drift. The transport rate of each trace was 6.5 Mbps. Each trace contained a single PAL video signal.

To model the influence of network-induced jitter, the trace was sent through an ATM network model with a large number of ON-OFF sources whose ON and OFF periods were distributed exponentially. The network topology used is shown in Figure 4.7. It consists of five cascaded ATM switches. The switch nodes are non-blocking, output-buffered crossbar switches. The MPEG-2 Transport Stream traces are sent through all the cascaded switches to the display device at the other end. At each hop of the network, the end-to-end MPEG stream shares the network link with cross traffic generated by a set of cell sources. All the cross-connections are between nodes that are connected to adjacent ATM switches. The propagation delay for each network link was set to 1 msec.

Experiment 1: No Cross-Traffic

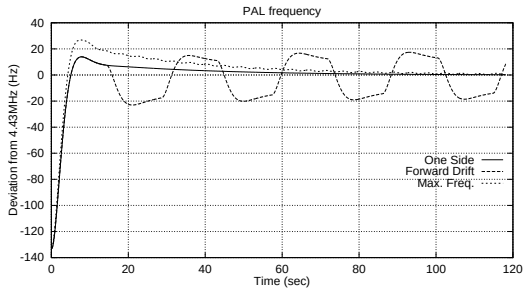

Figure 4.8: PAL color sub-carrier generation frequency for varying PCR insertion frequencies for experiment 1.

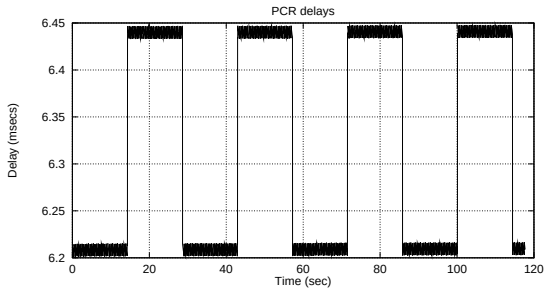

Figure 4.9: PCR delays of "Forward Drift" trace for experiment 1.

The goal of this experiment is to study the behavior of the clock recovery process at the decoder due to packetization jitter only, resulting from the use of different schemes to timestamp the MPEG-2 trace. Since all cross-traffic sources are turned off in this experiment, the network does not introduce any additional jitter.

The recovered clock at the decoder for the various cases is shown in Figure 4.8. It is evident that the trace with the forward drift results in unacceptable quality of the recovered clock since it violates the bounds for the PAL frequency. The clock frequency contains low-frequency components with the same frequency as the square wave representing the delay distribution of the transport packets containing PCR values (Figure 4.9). The jitter observed at the receiver is due to the packetization at the source, and is approximately 231 μsecs.

As expected, the trace containing PCR values only in even-indexed transport packets ("One Side") results in perfect clock recovery. the "Max. Freq." case also gives close to perfect results after the PLL becomes locked (Figure 4.8). The initial overshoot in this case comes from the fact

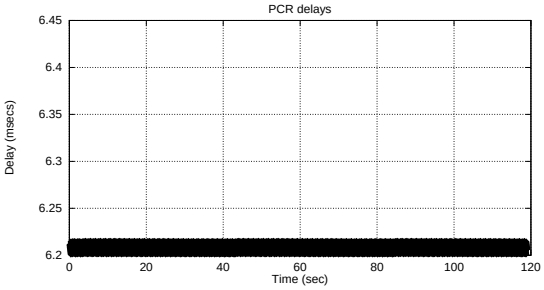

Figure 4.10: PCR delays of "One Side" trace for experiment 1.

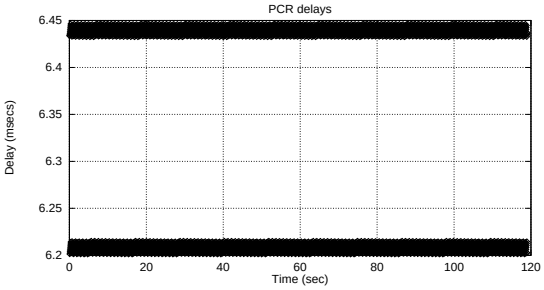

Figure 4.11: PCR delays of "Max. Freq." trace for experiment 1.

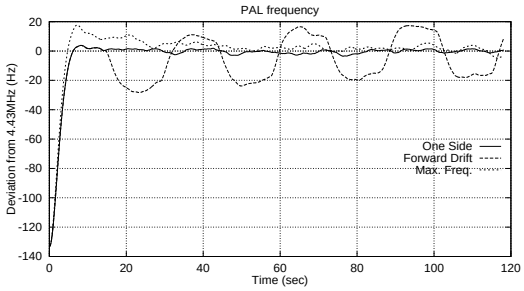

Figure 4.12: PAL color sub-carrier generation frequency for varying PCR insertion frequencies for experiment 2.

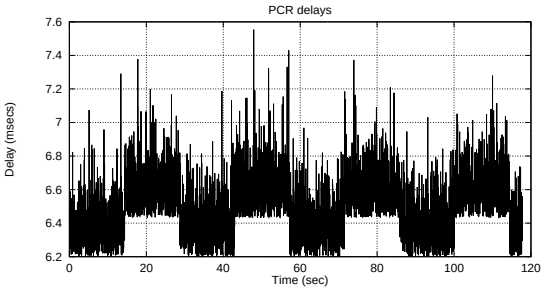

Figure 4.13: PCR delays of "Forward Drift" trace for experiment 2.

that the resulting packetization jitter is interpreted as higher frequency difference between sender and receiver. The delay experienced by transport packets containing PCRs for the "One Side" and the "Max. Freq." cases are plotted in Figures 4.10 and 4.11, respectively.

Experiment 2: Moderate Cross-Traffic

This experiment was performed to test whether jitter introduced by the network can mask the effect of the square wave pattern generated by the packetization jitter. In this experiment, thirty ON-OFF sources from each cross-traffic node were multiplexed with the MPEG-2 stream at each network link. The overall load on any downstream output link of the ATM switches was set to approximately 60%, resulting in a 10 Mbps aggregate rate of the ON-OFF sources per hop, or 0.334 Mbps load per source.

As seen in Figure 4.12, the recovered clock is still affected by the square-wave pattern of the packetization jitter when the "Forward Drift" trace was used, resulting in unacceptable quality of the recovered clock. The recovered PAL color sub-carrier generation frequencies for the other traces are also shown in Figure 4.12. The "One Side" and "Max. Freq." traces produce similar variations in the recovered clock which stays within the acceptable levels when the PLL becomes locked.

The plot of the delays experienced by transport packets containing PCRs for the "Forward Drift" case contains the square shape due to the forward drift described in Section 2 and is not masked by the jitter induced by the network. This is illustrated in Figure 4.13. However, the PCR delays in the "Max. Freq." case are spread out and the two delay zones due to the PCR-unaware scheme can no longer be distinguished clearly (Figure 4.14). The maximum jitter observed was approximately 1.5 msec. Finally, in the "One Side" case, the PCR delays have similar shape as in the "Max. Freq."

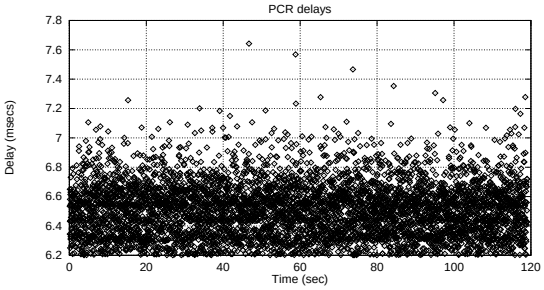

Figure 4.14: PCR delays of "Max. Freq." trace for experiment 2.

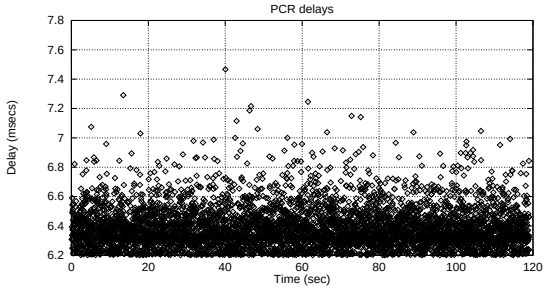

Figure 4.15: PCR delays of "One Side" trace for experiment 2.

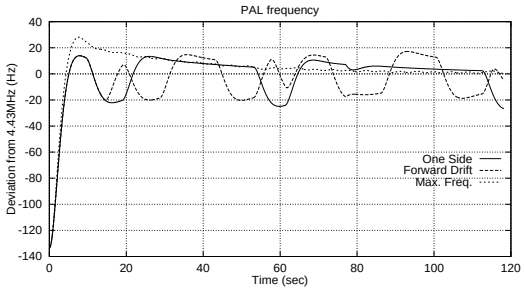

Figure 4.16: PAL color sub-carrier generation frequency for varying PCR insertion frequencies for experiment 3.

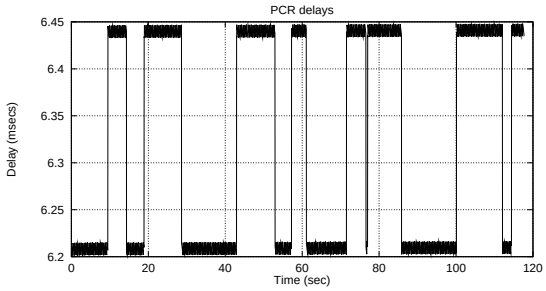

Figure 4.17: PCR delays of "Forward Drift" trace for experiment 3.

case but with a lower average value (Figure 4.15). This results in the slightly better quality of the reconstructed clock in the "One Side" case compared to the "Max. Freq" case, although in both cases the PAL specifications were not violated (Figure 4.12).

Experiment 3: Random Losses and No Cross-Traffic

In the previous experiments, we observed that the "One Side" and "Max. Freq." cases produce similar results for the decoder clock, with the first one being slightly better. However, this behavior occurred in a loss-free environment. When packets containing timestamp values are lost, the phase of the incoming timestamps can still switch, producing low-frequency components in the recovered clock signal. To investigate this effect, we simulated losses in the ATM network model by periodically dropping packets containing timestamp values. To simulate the worst-case behavior, we kept the packet loss artificially high, with a probability of 10^{-3} for loss of packets containing PCR values. The results are shown in Figure 4.16.

The "Max. Freq." case is the only one in which the reconstructed clock is not influenced by the losses in PCR packets. The "Forward Drift" case results in even worse performance than the loss-free case, with more peaks in its plot. A loss of a PCR packet in the "One Side" trace causes a major disturbance of the recovered clock since the PCRs change boundaries after the loss. The losses introduce very low frequency components at the input of the decoder, which cannot be filtered.

The delays experienced by transport packets containing PCRs for the "Forward Drift" case do not have a periodic pattern in this case, since a loss results in an additional polarity change for the delay (Figure 4.17). In the case of the "One Side" trace, the delays do not stay on the same side because of the same reason. The number of edges in that delay plot is equal to the number of PCR

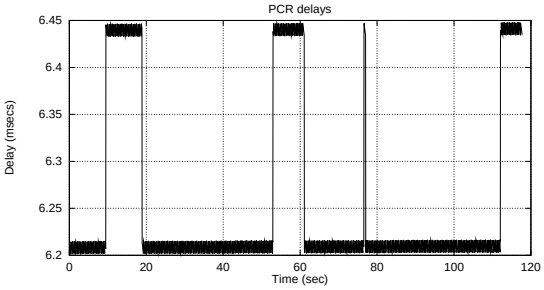

Figure 4.18: PCR delays of "One Side" trace for experiment 3.

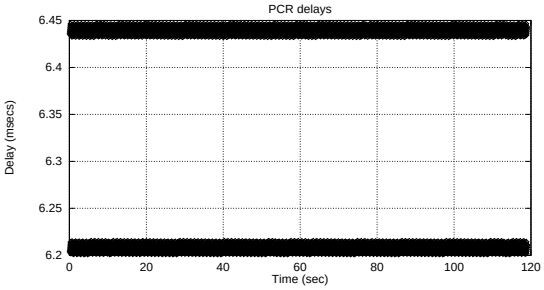

Figure 4.19: PCR delays of "Max. Freq." trace for experiment 3.

packets that were discarded in the experiment and is illustrated in Figure 4.18. Finally, the delay plot of the "Max. Freq." case (Figure 4.19) remains roughly the same with the corresponding delay plot when no losses are present (Figure 4.11).

5 Conclusions

Our primary contribution in this paper is in providing guidelines for selecting the timestamping interval for transmission of PCR timestamps in a packetized MPEG-2 transport stream. Based on a systematic analysis of the jitter introduced by the timestamping process at the receiver, we identified three approaches for setting the timer used to derive the timestamps. In the first approach, the timer period is set precisely so that the transport rate of the MPEG stream is an exact multiple of the timestamping rate. This completely eliminates packetization jitter, but is difficult to implement in practice because of the precision required in the timer setting. In addition, loss of packets carrying timestamp values can cause the PCR values in the packet stream to switch position, affecting the quality of the recovered clock.

The second approach is to fine-tune the timer period to maximize the frequency of changes in PCR polarity. To maximize the frequency, the timestamping interval must ideally be set to $T_c/(2k+1)$, where k is any non-negative integer and T_c the inverse of the transport rate in packets per second. This causes consecutive PCR values in the packet stream to alternate in polarity. This scheme has the advantage that, even when the timer cannot be set precisely to $T_c/(2k+1)$, the frequency of PCR polarity changes in the packet stream is still close to ideal. In addition, the scheme is robust in the presence of packet losses. Hence, this is the preferred scheme when the timestamps are generated with a fixed period.

When the transport rate of the MPEG-2 stream is not known and/or when a deterministic timer period is not practical, generating timestamping intervals randomly (with a certain minimum rate) can still provide adequate quality for the recovered clock. The quality of the receiver clock in this case depends on the process of PCR polarity changes, which, in turn is dependent on the distribution of the timestamping interval. By modeling the PCR polarity changes with a random telegraph process, we were able to derive lower bounds on the rate of PCR polarity changes to achieve the desired clock quality. Although derivation of a corresponding bound for the timestamping interval is not straightforward, the former provides us with a guideline for setting the random timestamping interval.

6 Acknowledgments

The authors would like to thank Yiannis Kontoyiannis at Stanford University for his valuable input on the analysis.

References

[1] I. F. Akyildiz, S. Hrastr, H. Uzunalioglu, and W. Yen. Comparison and evaluation of packing schemes for MPEG-2 over ATM using AAL5. In *Proceeding of ICC '96*, volume 3, pages 1411–1415, June 1996.

[2] G. F. Andreotti, G. Michieletto, L. Mori, and A. Profumo. Clock recovery and reconstruction of PAL pictures for MPEG coded streams transported over ATM networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 5(6):508–514, December 1995.

[3] CCIR 624. *Specifications for the color TV signal.* tab. II, item 2.11.

[4] M. De Prycker. *Asynchronous Transfer Mode : Solution for Broadband ISDN.* Ellis Horwood, second edition, 1993.

[5] S. Dixit and P. Skelly. MPEG-2 over ATM for video dial tone networks: issues and strategies. *IEEE Network*, 9(5):30–40, September–October 1995.

[6] D. Fibush. *Subcarrier Frequency, Drift and Jitter in NTSC Systems.* ATM Forum, July 1994. ATM94-0722.

[7] P. Hodgins and E. Itakura. *The Issues of Transportation of MPEG over ATM.* ATM Forum, July 1994. ATM94-0570.

[8] P. Hodgins and E. Itakura. *VBR MPEG-2 over AAL5.* ATM Forum, December 1994. ATM94-1052.

[9] International Organization for Standardization. *Information Technology — Generic Coding of Moving Pictures and Associated Audio: Systems, Recommendation H.222.0, ISO/IEC 13818-1*, draft international standard edition, November 1994.

[10] Y. Kaiser. Synchronization and dejittering of a TV decoder in ATM networks. In *Proceedings of PV '93*, volume 1, 1993.

[11] A. Leon Garcia. *Probability and Random Processes for Electrical Engineering.* Addison-Wesley Publishing Company, second edition, May 1994.

[12] H. Meyr and G. Ascheid. *Synchronization in Digital Communications*, volume 1 of *Wiley Series in Telecommunications.* John Wiley & Sons, 1990.

[13] M. Nilsson. MPEG-2 over ATM. In *IEE Colloqium on "MPEG-2 – what it is and what it isn't"*, number 1995/012. The Institute of Electrical Engineers, January 1995.

[14] M. Perkins and P. Skelly. *A Hardware MPEG Clock Recovery Experiment in the Presence of ATM Jitter.* ATM Forum, May 1994. ATM94-0434.

[15] J. Proakis and D. G. Manolakis. *Introduction to Digital Signal Processing.* Macmillan, 1988.

[16] P. A. Sarginson. MPEG-2 – a tutorial introduction to the systems layer. In *IEE Colloqium on "MPEG-2 – what it is and what it isn't"*, number 1995/012. The Institute of Electrical Engineers, January 1995.

[17] M. Schwartz and D. Beaumont. *Quality of Service Requirements for Audio-Visual Multimedia Services.* ATM Forum, July 1994. ATM94-0640.

[18] R. P Singh, Sang-Hoon Lee, and Chong-Kwoon Kim. Jitter and clock recovery for periodic traffic in broadband packet networks. *IEEE Transactions on Communications*, 42(5):2189–2196, May 1994.

[19] The ATM Forum Technical Committee. *Audiovisual Multimedia Services: Video on Demand Specification 1.0*, December 1995.

[20] C. Tryfonas. MPEG-2 transport over ATM networks. Master's thesis, University of California at Santa Cruz, September 1996. Available at `http://www.cse.ucsc.edu/research/hsnlab`.

[21] C. Tryfonas and A. Varma. A restamping approach to clock recovery in MPEG-2 systems layer. Technical Report UCSC-CRL-98-4, University of California at Santa Cruz, Dept. of Computer Engineering, May 1998.