Design of Optimal Playout Schedulers for Packet Video Receivers*

Nikolaos Laoutaris, George Boukeas, and Ioannis Stavrakakis

Department of Informatics, University of Athens, 15784 Athens, Greece
{laoutaris,boukeas,istavrak}@di.uoa.gr

Abstract. One way to reduce, or avoid, the loss of intrastream synchronization due to the delay variability introduced by best-effort networks, is by employing application layer buffering and scheduling at a Packet Video Receiver (PVR), resulting in a higher end-to-end delay. In this paper an analytical model is presented that captures the essential trade-off between stream continuity and stream latency. Unlike past related work, stream continuity is not expressed as the accumulated amount of synchronization loss, but as a combination of the accumulated amount, and the variation of the duration of synchronization loss occurrences. This approach allows for a fine grained optimization of stream continuity which has the potential of providing an improved perceptual quality. It is shown that the minimization of the accumulated amount of synchronization loss, and the minimization of the variance of the duration of synchronization loss occurrences, are two competing objectives; the minimization of variance is desirable because it leads to the concealment of discontinuities. The aforementioned presentation quality metrics are considered by the optimal playout policy, which is derived by means of Markov decision theory and linear programming.

1 Introduction

In recent years multimedia services such as internet telephony, desktop videoconference, and video on demand (VOD), have found a place next to traditional data applications like telnet, ftp or the world wide web. These new services require high transmission reliability and stringent end-to-end delay and delay jitter, to be able to maintain intrastream synchronization between successive media units. The networking community is currently focused on developing mechanisms that will enhance the Quality of Service capabilities of best effort network [1, 2]. The main effort in providing current best effort networks with QoS mechanism has been undertaken by the IETF which is standardizing the Integrated Services and Differentiated Services architectures. Nevertheless, it is realized that the deployment of new protocols will be a slow process, so much effort is being put

* This work and its dissemination efforts have been supported in part by the General Secretariat for Research and Technology of Greece under grants 99PENED-92 and 99PENED-193 and the IST Program of the European Union under contract IST-1999-10160 (VideoGateway).

in coping with current network limitations by incorporating intelligent adaptive algorithms at the application layer.

Adaptive rate applications fall into two general categories depending on which end of the communicating parties is adapting its rate or buffering capacity. In source rate adaptation [3, 4], it is the sending system that adapts to the time-varying bandwidth availability by regulating the rate of its output video stream. On the other hand, Packet Video Receiving systems (PVR's), may buffer some frames in a playout buffer or even make small adjustments in their playout rate in an effort to conceal the effects of jitter (lack of a frame to display caused by excessively delayed frames).

All PVR's buffer incoming frames as a measure against network jitter. Buffering frames in the playout buffer increases the end-to-end latency at the end-user level. The intrastream synchronization improvement that is gained with the addition of the playout buffer is bounded due to the need to keep the buffering delay below a threshold that is specified by the available end-to-end delay budget.

Different applications tolerate different maximum end-to-end latencies. Bidirectional applications such as desktop video conferencing place very strict latency requirements, typically of few hundreds of milliseconds. On the other hand, unidirectional applications, for example video on demand (VOD), allow for much larger latencies in the order of seconds. In a VOD application, the PVR can buffer massive numbers of frames, thus ensure an almost pauseless video presentation across the widest range of network jitter. The absence of critical latency requirements also allows for the incorporation of techniques such as data proxying and client-server feedback which can help in using network resources more efficiently, especially in the case of Variable Bit Rate (VBR) encoded video [5–7].

The hard real time requirements of interactive applications [8] are met by the *absolute delay method* which delivers frames at a constant end-to-end latency and drops late frames. Applications with looser delay constraints (soft real time) may present a late frame and thus gain in stream continuity by not discarding a frame which has already harmed the continuity of the stream by causing an underflow with its late arrival. Keeping the late frame increases the end-to-end latency of all subsequent frames resulting in a playout policy with variable overall latency for different frames. Of course, even soft real time applications have an upper bound on latency or a limited buffer capacity so eventually the increase of buffering delay will lead to frame droppings.

The family of playout schedulers that we study in this work does not guarantee a constant end-to-end latency nor a constant buffering delay. The scheduler guarantees only a statistically constant (mean value) buffering delay and is thus more suitable for soft real time applications. The gain from the relaxation of the constant latency requirement, is the ability to react better to bursty frame-arrival sequences forming due to network jitter. Perceptual quality can be improved by implementing *mild* latency control methods, which harm stream continuity less than the *harsh* deadline discard of late frames under the absolute delay method. At the same time, the buffering delay is not ignored – in favor of stream continuity – but is kept statistically constant, below an acceptable, user-defined level. The scheduler increases the buffer occupancy as a measure against jitter, and decreases the buffer occupancy, in order to control the buffering delay and avoid overflows. The playout buffer occupancy is controlled by regulating the playout duration of frames in a per frame basis. This necessity is not present in packet audio systems, where the existence of silence periods gives the system the ability to change the size of the de-jitter buffer by modifying the duration of the silence periods, in a per talkspurt basis, without modifying the duration of media units [9–14].

Analytical studies for PVR's that employ dynamic playout schedulers, have recently appeared in the literature. Yuang et al. [15] proposed a dynamic playout policy based on *slowdown* thresholds. In their work, frames are presented at a linearly decreasing rate when the playout buffer occupancy drops below TH – the slowdown threshold – and at a constant rate μ, faster than the mean frame arrival rate λ, when the occupancy exceeds TH. We have modified the threshold-based scheduler of [15] in [16], by looking at the case where the scheduler applies a playout rate that never exceeds the mean frame arrival rate. This modification dictates that the buffering delay will only increase (it decreases when the playout rate exceeds the normal encoding rate) resulting in a scheduler that is more suitable for unidirectional, soft-real-time applications, such as web-based video distribution systems. The design of the scheduler has been simplified by the introduction of a compact and fair continuity metric – the Distortion of Playout (DoP) – which combines all causes of media asynchrony. The study has limited the range of the threshold parameter, TH, by identifying a range of values where there is no beneficial tradeoff between continuity and reduction of mean playout rate – the two antagonistic metrics of interest. Interestingly, it has been shown that this range of values changes with the burstiness of the frame arrival process, revealing the danger of an initially meaningful TH appearing in the undesirable area due to the change of arrival burstiness. The work is supplemented by online algorithms for the detection and the maintenance of the operational parameter TH within the area of beneficial tradeoff across unknown, non-stationary, delay jitter.

With the current work we improve previous heuristic frame-playout policies by formulating an optimization problem that involves the two main metrics of interest; the intrastream synchronization and the buffering delay. Furthermore, the intrastream synchronization metric used in this work is more fair and more general than previously used synchronization metrics [15–18]. In addition, the buffering delay is controlled in a way that harms stream continuity as little as possible.

The remainder of the paper is organized as follows. Some key concepts and definitions are presented is Section 2. A number of optimal playout adaptation policies for packet video receivers are derived is Section 3 by introducing some interesting metrics and employing Markov decision theory and linear programming techniques. Numerical results are presented in Section 4 together with a comparative analysis with a non-optimal scheduler from the literature. Section 5 comments on implementation issues and future development issues. Section 6 concludes the paper.

2 Definitions

In the following sections we construct a mathematical model for the derivation of the optimal frame-playout policy. Among others, the need for a method to control the duration of frames will arise. Under R_T – the usual static playout policy– all frames are presented with an equal duration T, given by the frame production rate λ, via $T = 1/\lambda$. In some cases, we will need to expand the presentation duration of a frame beyond T, the normal duration. When this decision is repeated for successive frames, it constitutes a transient reduction of playout rate (we will refer to this as a *slowdown*). On the opposite, the presentation of a sequence of frames, with durations that are shorter than T, constitutes a transient increase of playout rate (much like a *fast forward* operation on a VCR). The most general way to regulate B, the duration of a frame, is to allow it to take all non-negative values. This approach gives the utmost freedom in the search for an optimal playout policy. In this paper, however, we limit the possible values for B to a countable set of values that follow:

$$B_k \triangleq k \cdot c$$

where c is an fraction of the normal frame duration T, such that $T = a \cdot c$. The value of a is the *cutting factor* of T. Using the last relationship B_k becomes:

$$B_k = \frac{k}{\lambda \cdot a} \tag{1}$$

$\lambda = 1/T$ is the normal frame rate, typically 25 or 30 frames/sec. c will be the basic unit for shortening and expanding the duration of a frame. The reduction of freedom in the search for optimal policies is negligible for small values of c, e.g., $c = T/10$. In the following, the choice of an appropriate value of k will be referred to as an *action* or a *decision*.

Expanding or shortening the duration of a frame presentation introduces a discontinuity – a loss of intrastream synchronization – quantified by the difference between the selected frame duration and the normal frame duration T. Let d_{ik} denote the *discontinuity* that is incurred when the next frame is presented with a duration B_k and the current buffer occupancy is i.

$$d_{ik} \triangleq |B_k - T| + S \cdot I_{\{i=0\}} \qquad 0 \leq i \leq N \tag{2}$$

S is a random variable that adds to d_{ik} the effect of buffer underflows. S represents the time interval between a buffer underflow instant and the next arrival instant. I is the indicator function. We define the Distortion of Playout as:

$$DoP_{ik} \triangleq d_{ik} + \bar{l}_{ik} \cdot T \tag{3}$$

where $\bar{l}_{ik}$ [1] is the expected number of lost frames due to buffer overflow over the next presentation interval, given that i is the current buffer occupancy, and decision k is made. $\bar{l}_{ik}$ is given by:

$$\bar{l}_{ik} = \sum_{m=1}^{\infty} m \cdot P\{N - i + 1 + m, B_k\} \tag{4}$$

$P\{x, t\}$ is the probability of x new arrivals occurring in a time interval with duration t. DoP is more fair than d_{ik} as a stream continuity metric, as it also accounts for synchronization losses due to frame overflows.

A basic idea reflected in the definitions of both d_{ik} and DoP_{ik} is that the perceptual cost of an idle time gap between two frames (occurring when the first frame stays on display for more than T) is equal to the perceptual cost of a loss-of-information discontinuity of equal duration. This is based on recent perceptual studies [19] where it is shown that jitter degrades the perceptual quality of video nearly as much as packet loss does [2].

3 Design of Optimal Playout Schedulers

In this section a PVR consisting of a playout buffer and a playout scheduler is studied using Markov decision theory and linear programming. The goal is to derive the optimal playout policy, which by controlling the duration of frames based on the current buffer occupancy, provides a perceptually optimal presentation schedule.

3.1 MDP Problem Formulation

Let $\{I_n\}_{n>0}$ be a stochastic process for i, the number of frames in the playout buffer upon the presentation completion instant[3] of the nth frame. $\{I_n\}$ is a Markov process under the Poisson arrival process which is assumed in this paper for the modeling of frame arrivals. The Poisson process and the associated interarrival-time exponential distribution are much more variable than actual frame arrival processes and thus lead to rather pessimistic performance results. Nevertheless, the memoryless property of the exponential distribution simplifies the model and is thus suitable for a first exposition of the basic ideas that appear in this paper (the implications of the Poisson assumption are further examined in Sect. 5).

To formulate $\{I_n\}$ as a Markov decision process (MDP), we need to define a tuple $\langle \mathcal{S}, \mathcal{A}, P, C \rangle$, where $\mathcal{S}$ is the set of possible states, $\mathcal{A}$ is the set of possible actions, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the state transition function specifying the probability $P\{j|i, k\} \equiv p_{ij}(k)$ of observing a transition to state $j \in \mathcal{S}$ after

[1] See [16] for a detailed derivation of $\bar{l}_{ik}$.

[2] For an example in support of this claim; we may think that an underflow with a duration T, degrades stream continuity, nearly as much, as does a lost frame.

[3] Hereafter called an *observation*, or a *decision* instant.

taking action $k \in \mathcal{A}$ in state $i \in \mathcal{S}$ and, finally, $C : \mathcal{S} \times \mathcal{A} \to \Re$ is a function specifying the cost $C(i, k) \equiv c_{ik}$ of taking action $k \in \mathcal{A}$ at state $i \in \mathcal{S}$. A *policy* $R \equiv \{D_{ik} : i \in \mathcal{S}, k \in \mathcal{A}\}$ is a mapping: $\mathcal{S} \times \mathcal{A} \to [0, 1]$. A policy is completely defined for a given tuple $\langle \mathcal{S}, \mathcal{A}, P, C \rangle$ by the probabilities

$$D_{ik} \triangleq P\{\text{action} = k | \text{state} = i\} \tag{5}$$

The state space $\mathcal{S}$ of $\{I_n\}$ comprises all possible buffer occupancy levels thus takes values in $[0 \dots N]$, N being the buffer size. An *action* is defined to be the choice of an integer value k that explicitly determines B_k, the presentation duration for the next frame. The action space for the problem is $\mathcal{A} = [0 \dots K]$, where K is an integer value that results in the maximum allowable playout duration B_K. For $k = 0$ the playout scheduler discards the next frame.

The action taken at an observation instant affects the evolution of $\{I_n\}$ by affecting the probability law for the next transition. Let P be a $(N+1) \times (N+1)$ matrix containing vectors elements $\boldsymbol{p}_{ij}$, with $K + 1$ elements per vector. The $(k + 1)$th element of vector $\boldsymbol{p}_{ij}$ is the probability of observing a transition from state i to state j when decision $k, k \in \mathcal{A}$, is made. Let $P\{x, t\}$ denote the probability of observing x new frame arrivals in a time interval of duration t. Since $\{I_n\}$ corresponds to the buffer occupancy, the probability of observing a transition to a state j after playing the next frame for time B_k (selecting action k), depends on the number of new frames that will arrive during B_k.

$$p_{ij}(k) = \begin{cases} P\{j, B_k\} & i = 0, 0 \leq j < N \\ 1 - \sum_{m=0}^{N-1} P\{m, B_k\} & i = 0, j = N \\ P\{j - i + 1, B_k\} & 0 < i < N, i \leq j < N \\ 1 - \sum_{m=0}^{N-i} P\{m, B_k\} & 0 < i \leq N, j = N \\ P\{0, B_k\} & 0 < i \leq N, j = i - 1 \\ 0 & \text{elsewhere} \end{cases} \tag{6}$$

In this paper $P\{x, t\}$ follows the Poisson distribution with parameter λ, the frame production rate. Using the transition matrix of (6), we can obtain $\pi_i(R)$, the limiting distribution of $\{I_n\}$ for a particular policy R, by solving the stationary equations $\pi(R) = \pi(R) \cdot P(R)$

Let c_{ik} denote the cost incurred when action k is taken when the occupancy process is in state i. The optimal policy R_{opt} is defined to be the policy that minimizes the expected value of c_{ik} over all $i, k \in \mathcal{S} \times \mathcal{A}$. If π_i denotes the limiting probability that process $\{I_n\}$ is in state i, then

$$R_{opt} = \arg\min_R E\{c\} \quad \text{where} \quad E\{c\} = \sum_{i=0}^{N} \sum_{k=0}^{K} c_{ik} \cdot D_{ik} \cdot \pi_i$$

3.2 Cost Assignment

The cost considered here consists of two components: one that captures the induced lack of continuity, and one that captures the induced buffering delay. The relative weighing between the two cost components, reflects the desired latency-continuity performance compromise that is to be achieved by the optimal policy.

Continuity Cost This cost component punishes the *lack of continuity* that may arise from a certain action. This lack of continuity may be directly experienced as in the case of a frame presentation with a duration smaller or larger than T. In addition, the continuity cost also accounts for the anticipated lack of continuity that may arise over the chosen presentation interval as consequence of the chosen action. To be precise, the anticipated lack of continuity refers to the possibility of losing frames due to buffer overflow over the chosen presentation interval.

An candidate for the continuity cost is DoP_{ik}, as defined in (3). Setting $c_{ik} = DoP_{ik}$ returns an $E\{DoP\}$-optimal policy; a policy that provides a minimal expected value for the Distortion of Playout. The results of Sect. 4 indicate that R_T, the static deterministic policy with constant presentation durations equal to the frame period, is $E\{DoP\}$-optimal.

The minimization of $E\{DoP\}$ calls for the minimization of the accumulated amount of synchronization loss which is due to: underflow discontinuities, slowdown discontinuities, overflow discontinuities and fast-forward discontinuities. The minimization of $E\{DoP\}$ is a legitimate objective but cannot guarantee perceptual optimality, as it only caters for the minimization of the accumulated loss of synchronization, without paying any attention as to how this loss of synchronization spreads in time. It has been realized that the human perceptual system is more sensitive to a small frequency of long-lasting disruptions than to a higher frequency of sort-lived disruptions [15]. This is due to human perceptual inability to notice small deviations of presentation rate. As a result, a better perceptual quality can be expected by replacing large continuity disruptions (underflows and overflows) with shorter ones (slowdowns and fast-forwards), even when the later lead to a higher value for $E\{DoP\}$. Thus, a playout policy should be allowed to sacrifice an increase of $E\{DoP\}$ if this increase provides for a smoother spacing between synchronization-loss occurrences, thus help in concealing them. We pursue this idea by defining the state-action cost to be

$$c_{ik} \triangleq \beta \cdot DoP_{ik} + (1 - \beta) \cdot DoP_{ik}^2$$

The weighing factor β is a user-defined input that controls the relative importance between the two minimization objectives; the minimization of the expected value of DoP and minimization of the variability of DoP. Setting $\beta = 1$ leads to the minimization of $E\{DoP\}$ without any regard for the variance of DoP. Setting $\beta = 0$ results in the cost $c_{ik} = DoP_{ik}^2$, demanding the minimization of the expected square value of DoP. The minimization of $E\{DoP^2\}$ returns a policy that distributes synchronization losses more smoothly than $E\{DoP\}$-optimal policies do. As will be shown later, the reduction in the variance of DoP comes at the cost of an increase in the expected value of DoP. Values of β that fall between the two extremes (0 and 1) provide various levels of compromise between $min\{E\{DoP\}\}$ and $min\{E\{DoP^2\}\}$. Due to the fact that the two quantities have different units (here *seconds* and *seconds*^{2}) small values of β must be used if a meaningful tradeoff is to be achieved, otherwise the policy turns absolutely in favor of $min\{E\{DoP\}\}$.

Latency Cost In addition to the continuity cost a latency cost is jointly considered to allow for the control of the buffering delay.

The buffering delay for an arriving frame depends on the number of frames i found in the buffer upon arrival, the presentation duration associated with these frames, $B_n, n = 1 \ldots i$, and the remaining duration of the frame that is currently being presented, X. The expected buffering delay of an arriving frame that finds i frames in the buffer is:

$$W_i = X + \sum_{n=1}^{i} B_n \tag{7}$$

W_i cannot be expressed explicitly, as B_n's follow a distribution which is not known; in fact, this distribution is is derived in the process of obtaining the optimal policy. Nevertheless, it is expected that the mean value of B_n will not differ significantly from the actual frame duration T (since the desirable mean playout rate induced by the optimal policy is close to and cannot exceed $1/T$). Thus, W_i can be approximated by some value in the interval $[i \cdot T, (i+1) \cdot T]$. $\widetilde{W}_i = i \cdot T$ is chosen as the approximation of W_i. Let $\{A_n\}_{n>0}$ denote the buffer occupancy process on frame arrival instants and let $a_i, i = 0 \ldots N$ denote the distribution of $\{A_n\}$. Then the expected buffering delay is approximated by:

$$\widetilde{W} = \sum_{i=0}^{N} a_i \cdot \widetilde{W}_i = \sum_{i=0}^{N} a_i \cdot i \cdot T \tag{8}$$

Equation (8) approximates the mean buffering delay by a function of a_i, the distribution of $\{A_n\}$. It applies that $\{A_n\}$ has the same distribution with $\{I_n\}$, that is, $a_i \equiv \pi_i$ (see Sec. 5.3 in [20] or Sec. 8.3 in [21] for details); thus $\widetilde{W}$ becomes a function of π_i, the distribution of $\{I_n\}$. By regulating the mean value of $\{I_n\}$ we also regulate the delay approximation $\widetilde{W}$ since the latter only depends on the distribution π_i. The mean value of $\{I_n\}$, and thus the delay $\widetilde{W}$, can be regulated by introducing a latency cost component to every action. We define the latency cost for taking decision $k, k \in \mathcal{A}$, when the number of frames in buffer (the state) is i to be

$$L_i \triangleq \frac{1}{T \cdot N} \cdot \widetilde{W}_i = \frac{i}{N} \tag{9}$$

The latency cost does not explicitly depend on the chosen playout duration (determined by k), but only implicitly, as the evolution of $\{I_n\}$ depends on the presentation duration of the current frame. The buffer size N together with T normalize the latency cost to unity.

We add the latency cost to the continuity cost introduced in the previous section by attaching to it a weighing factor γ that controls its relative importance over the continuity cost. The final expression for the transition cost is given by

$$c_{ik} \triangleq \beta \cdot DoP_{ik} + (1-\beta) \cdot DoP_{ik}^2 + \gamma \cdot L_i \tag{10}$$

3.3 Linear Programming Formulation

In this section, we employ linear programming (LP) to derive the optimal policy for the MDP of Sect. 3.1. The decision variables of the linear program are x_{ik}'s which denote the joint probability of being at state $i \in \mathcal{S}$ and performing action $k \in \mathcal{A}$:

$$x_{ik} = P\{\text{action} = k \text{ and state} = i\}$$

The x_{ik}'s do not directly stipulate a policy, a way to select an action at a given state. However, their are required by the *Simplex Method*, used for the solution of the LP, as they are more convenient as decision variables than the actual policy D_{ik}. Having calculated x_{ik}'s, the optimal policy is readily determined since it holds

$$D_{ik} = \frac{x_{ik}}{\pi_i} \quad \text{where} \quad \pi_i = \sum_{k=0}^{K} x_{ik}$$

The function z we wish to minimize is the expected cost over the state-action space, whereas the constraints of the problem reflect the structure of the MDP. The complete problem is the following:

minimize

$$z = E\{c\} = \sum_{i=0}^{N} \sum_{k=0}^{K} c_{ik} x_{ik} \tag{11}$$

subject to

$$\forall j: \ \sum_{k=0}^{K} x_{jk} = \sum_{i=0}^{N} \sum_{k=0}^{K} x_{ik} p_{ij}(k) \tag{12}$$

$$\sum_{i=0}^{N} \sum_{k=0}^{K} x_{ik} = 1 \tag{13}$$

$$\forall i, k: \ x_{ik} \geq 0 \tag{14}$$

The $N + 1$ constraints in (12) are the steady-state equations, suggesting that the steady state probability of being in state j equals the sum of probabilities of being in any other state i and performing a transition to j. Constraint (13) and the $N + 1 \times K + 1$ constraints in (14) stem from the fact that the x_{ik}'s form a probability distribution over the state-action space and must, therefore, be non-negative and sum to 1.

It can be proved [22, 23] that for each state $i \in \mathcal{S}$ there is *exactly* one $x_{ik} > 0$. This essentially means that the optimal policy is deterministic i.e., D_{ik}'s are 0 or 1.

4 Results and Discussion

In this section we demonstrate the effectiveness of the MDP formulation by deriving several numerical results for various parameter configurations. All the examples will be based on a reference system with the following parameters: a playout buffer with space for $N = 50$ frames; a normal frame rate $\lambda = 30$ frames/second; a basic time quantum c, equal to a tenth of the normal frame period T (i.e., a=10 in (1)). The two weighing factors, β, γ of (10), will shape different optimal policies for applications with different continuity-latency requirements.

4.1 Optimizing Stream Continuity

The following numerical results demonstrate policies that perceive the intrastream synchronization quality, as a combination of $E\{DoP\}$ and $Var\{DoP\}$, following the arguments of Sect. 3.2. The latency weight γ is set to zero, thus does not affect the shape of the optimal policy. We focus on the role of continuity weight β, that regulates the relative importance between the mean value and the variance of DoP.

Figure 1 depicts the effect of β on the structure of the optimal policy. The x,y surface illustrates the optimal policy, for a specific value of β (on the z-axis). We let β take values in $[0, 0.1]$ with an increment of 0.001. We have already noted that only small values of β provide a meaningful compromise between mean value and variance. This is due to the fact that the state dependent cost components DoP_{ik} and DoP^2_{ik} have different units. We save a lot of unnecessary optimization runs by not letting β run up to 1. The optimal policy becomes R_T, statical deterministic with presentation duration T regardless of buffer occupancy, soon after $\beta = 0.1$. For small values of β the policy tries to minimize the variance of DoP. In doing so, it presents frames slower when approaching an underflow and faster when approaching an overflow. For the given numerical example, the maximum frame duration is B_{14} and the minimum B_9. It might seem odd that for an occupancy equal to zero, the optimal policy plays the next frame with a normal duration $B_{10} = T$ and not slower as would be expected. This decision is justified by noting that for the special case of an underflow occurrence, prior to the display of the first arriving frame, an underflow interval exists; the policy does not lengthen the duration of this frame because the associated *slowdown* discontinuity d_{0k}, would become very large as it also contains the underflow interval.

Figures 2,3 illustrate the effect of β on the values of $E\{DoP\}$ and $Var\{DoP\}$. It is evident that small values of β favor the reduction of $Var\{DoP\}$ by increasing $E\{DoP\}$, while the opposite holds for large values of β ($\beta > 0.1$), where the optimal policy returns a small value for $E\{DoP\}$ but with a large variance of DoP. Note that both $E\{DoP\}$ and $Var\{DoP\}$ are not continuous but change in steps at different values of β. This is an expected behavior as different values of β produce policies that may differ at some *specific* state-action pairs. The discrete action space A allows only specific values for $E\{DoP\}$ and $Var\{DoP\}$.

Figure 4 reveals a relationship between the continuity weight β, and $E\{I_n\}$, the expected buffer occupancy at the observation instants. Small values of β, cause a slight increase in the occupancy of the buffer. This is justified by looking carefully at the structure of policies that favor the reduction in the variance of DoP (β taking small values). This reduction of variance is achieved by adjusting the duration of frames at the buffer extremes, thus preventing occurrences of

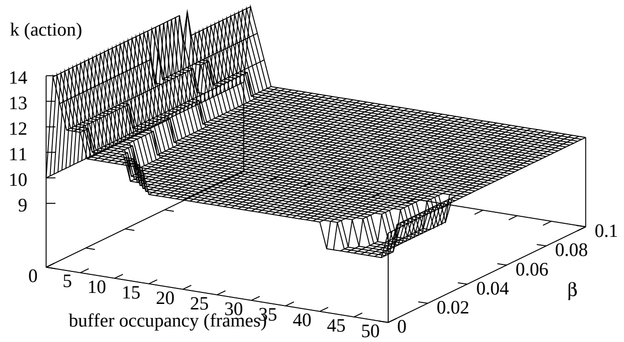

Fig. 1. Optimal policies for various values of β, the weight factor that governs the tradeoff between mean value and variance of DoP. No latency cost is considered here ($\gamma = 0$). The normal frame duration T corresponds to B_{10} i.e., when the decision value k is ten.

variance-increasing events, such as underflows and overflows. Figure 1 shows that presentation slowdowns – occurring when the buffer level drops – are severe (large expansion of the duration of the frames for very low buffer occupancy). In comparison, presentation fast-forwards, are less severe (slight reduction of the duration of frames for very high buffer occupancy). This behavior leads to a slightly reduced mean playout rate which increases the mean buffer occupancy by increasing the limiting probabilities of large occupancy levels.

In the aforementioned analysis, the buffering delay is limited by N, the maximum buffer capacity. It also decreases with β, as shown in Fig. 4. Having neglected γ, the weighing factor of the latency cost, an implicit buffering delay method can be devised, be choosing a limited buffer size N, and use β for more detailed delay adjustments.

4.2 Comparative Analysis with Non-Optimal Schedulers

This section demonstrates the performance gains of the optimal scheduler by comparing its performance with the performance of the non-optimal empirical threshold-based playout scheduler of [16] (briefly discussed in Sec. 1). Figure 8 illustrates the performance of the threshold scheduler. The optimal value for the expected value of DoP is achieved for the threshold parameter $TH = 2$ and is $E\{DoP\}_{TH=2} = 6.8 \cdot 10^{-4}$. The variance of DoP for $TH = 2$ is $V\{DoP\}_{TH=2} = 1.69 \cdot 10^{-5}$. Let $X\{DoP\}'_{\beta}$ denote the expected value or the variance of DoP for some value(s) β of the optimal scheduler. The numerical results show that for the same value of averages, $E\{DoP\}'_{\beta>0.09} = E\{DoP\}_{TH=2}$, the optimal scheduler exhibits twice as better variance of DoP, $V\{DoP\}'_{\beta>0.09} = 0.8 \cdot 10^{-5} < 1.69 \cdot 10^{-5} = V\{DoP\}_{TH=2}$. Fixing the value of the variance at $V\{DoP\}_{TH=14} = V\{DoP\}'_{\beta>0.09} = 0.8 \cdot 10^{-5}$, the optimal policy is again superior by achieving a better expected value of DoP, $E\{DoP\}'_{\beta>0.09} = 6.8 \cdot 10^{-4} < 8.4 \cdot 10^{-4} = E\{DoP\}_{TH=14}$. Note, that for the threshold scheduler, the absolutely smallest

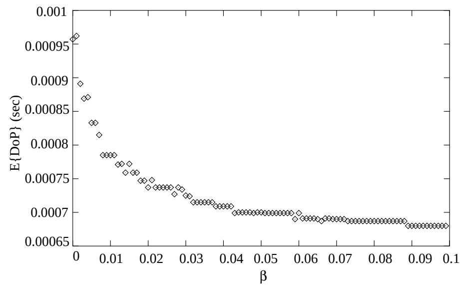

Fig. 2. The effect of the continuity weight β on $E\{DoP\}$. The latency weight γ is set zero.

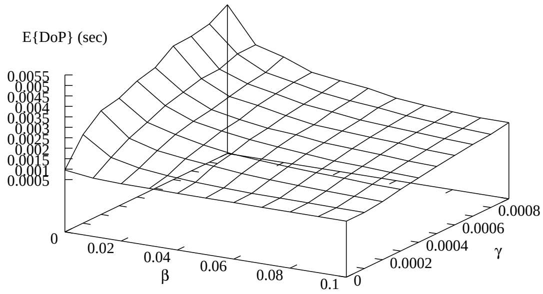

Fig. 5. The effect of β and γ on $E\{DoP\}$.

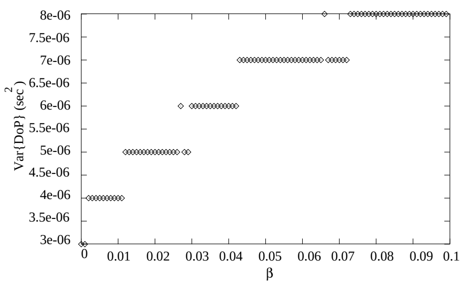

Fig. 3. The effect of the continuity weight β on $Var\{DoP\}$. The latency weight γ is set zero.

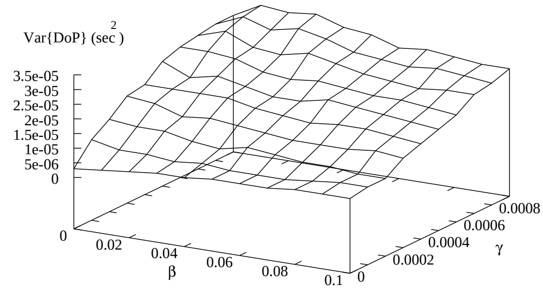

Fig. 6. The effect of β and γ on $Var\{DoP\}$.

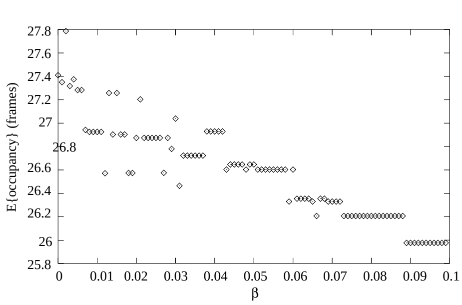

Fig. 4. Expected playout buffer occupancy with β.

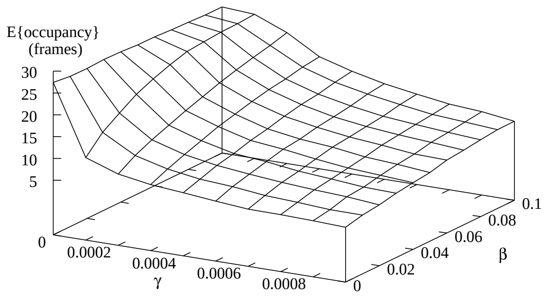

Fig. 7. The effect of β and γ on buffer occupancy.

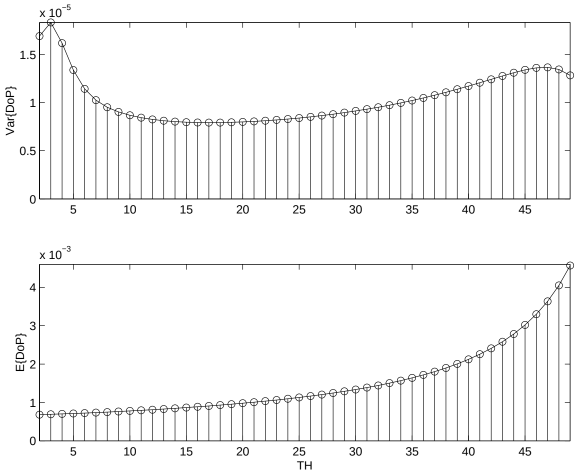

Fig. 8. $E\{DoP\}$ and $V\{DoP\}$ from a threshold-based playout adaptation scheme of [16] where the value of the threshold parameter (TH) regulates the tradeoff between the two metrics. The expected value is minimized for $TH = 2$ and the variance for $TH = 17$. The optimal policy outperforms the threshold-based schedulers in the entire continuum of choices between optimization of $E\{DoP\}$ and optimization of $V\{DoP\}$.

variance of DoP is achieved at a threshold value of $TH = 17$: $V\{DoP\}_{TH=17} = 0.79 \cdot 10^{-5}$. We used for the comparison $TH = 14$ instead of $TH = 17$ because for the latter we cannot find a value of β that provides $V\{DoP\}'_{\beta} = V\{DoP\}_{TH=17}$ and fix $V\{DoP\}$ for a numerical comparison, nevertheless, $TH = 14$ is very close to the absolutely smallest $V\{DoP\}$ achieved by the threshold scheduler at $TH = 17$. Also note that the absolutely smallest $V\{DoP\}$, from both methods, is achieved by the optimal scheduler for $\beta = 0$ and it is over two times superior to the optimal $V\{DoP\}$ performance of the threshold scheduler at $TH = 17$. Thus, it is concluded that the optimal scheduler outperforms the threshold-based scheduler in the entire continuum of choices between optimization of $E\{DoP\}$ and optimization of $V\{DoP\}$.

4.3 The Tradeoff Between Stream Continuity and Buffering Delay

In this section we show how the latency weight γ of (10) can be used to control the playout buffer occupancy and the associated buffering delay. Our results indicate that the latency control function has a twofold degrading effect on stream continuity; both the expected value of DoP and the variance of DoP, increase with γ. The degradation of synchronization quality has also been exhibited with schedulers that employ *harsh* latency control methods, such as deadline discard of late frames. It also applies to our scheduler, despite the *milder* latency control function, which does not discard frames, but plays them faster in order to control the buffering delay.

Figure 7 illustrates the reduction of the expected value of $\{A_n\}$ with increasing values of γ. For a given latency weight γ, smaller values of β lead to smaller occupancy levels. This happens as the DoP_{ik} costs (adding to c_{ik} more with large βs) are higher than the corresponding DoP_{ik}^2 costs (adding to c_{ik} more with small βs); thus larger βs balance more effectively the latency weight γ, which is pushing for greater latency reduction.

By choosing an appropriate latency weight the playout buffer occupancy can be stabilized to a level that adds an acceptable buffering delay component. Figure 5 [4] shows the increase in the expected value of DoP that is paid for the regulation of buffer occupancy. The tradeoff between continuity and buffering delay is apparent.

Figure 6 depicts the effect of the latency weight on the variance of distortion of playout. As we would expect the occupancy control function not only increases $E\{DoP\}$ but also increases $V\{DoP\}$.

5 Implementation Issues and Future Work

It has already been mentioned that the exponential interarrival distribution is much more variable compared to typical interarrival distributions of periodic streams. The optimal policy is expected to approach an optimal performance as long as the variability of a real distribution approaches the variability of the exponential distribution. Such high variability can be realistic in time-windows of extreme network jitter. However, in times of reduced network jitter the proposed scheme can deviate significantly from the expected optimal performance. To overcome this problem, the analytical model is being expanded to allow for the incorporation of arrival processes that are more regular than Poisson. This will allow for the derivation of different optimal policies according to the current level of network jitter. The envisioned implementation will only need to estimate the current level of network jitter and then *load* the appropriate offline-computed optimal policy. The gain of such an architecture is twofold: the performance of the scheduler is optimized, and the complexity of the system is kept low as no online optimization is performed since offline computed policies are being used.

6 Conclusions

This paper has presented a family of playout schedulers for packet video receivers. The proposed schedulers optimize a meaningful expression of stream quality which involves the two major performance quantities: the stream continuity and the induced buffering delay. A fair and compact stream continuity metric – the Distortion of Playout – has been used as the basis for the detailed assessment of the overall intrastream synchronization quality that caters

[4] Note that compared to Fig. 7, we have switched the axes for γ and β in order to make the plots more comprehendible.

for both the accumulated amount of discontinuities and variance of their duration. It has been suggested that the joint consideration of the amount and the pattern of synchronization loss has the potential of improving the perceptual quality, compared to the case where only the amount is considered.

The numerical results reveal an important tradeoff that must be considered when designing a packet video receiver. It is realized that the reduction of the accumulated amount of synchronization loss and the attempt to evenly spread it in time, are two antagonistic objectives. Both continuity components are degraded by the introduction of a delay control function, despite the fact that the proposed delay control scheme tries to be friendly towards stream continuity, by avoiding crude operation such as frame discardings.

The performance gains of the optimal scheduler have been pointed out by means of comparison with an empirical scheduler from the literature. Finally, the applicability of the proposed schemes has been discussed together with directions for the future development of the work.

References

1. Dinesh C. Verma, Hui Zhang, and Domenico Ferrari, "Delay jitter control for realtime communication in a packet switching network," in *Proceedings of Tricomm '91*, Chapel Hill, North Carolina, Apr. 1991, IEEE.
2. Domenico Ferrari, "Distributed delay jitter control in packet-switching internetworks," Technical Report TR-91-056, University of California, Berkeley, California, Oct. 1991, also in *Journal of Internetworking: Practice and Experience.*
3. Reza Rejaie, Mark Handley, and Deborah Estrin, "An end-to-end rate-based congestion control mechanism for realtime streams in the internet," in *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, New York, Mar. 1999.
4. Sally Floyd, Mark Handley, Jitendra Padhye, and Joerg Widmer, "Equation-based congestion control for unicast applications," in *SIGCOMM Symposium on Communications Architectures and Protocols*, Stockholm, Sweden, Aug. 2000, ACM.
5. James D. Salehi, Zhi-Li Zhang, Jim Kurose, and Don Towsley, "Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing," *IEEE/ACM Transactions on Networking*, vol. 6, no. 4, pp. 397–410, Aug. 1998.
6. Z. Jiang and L. Kleinrock, "A general optimal video smoothing algorithm," in *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, San Francisco, California, March/April 1998, p. 676.
7. Zhi-Li Zhang, Yuewei Wang, D. H. C. Du, and Dongli Su, "Video staging: A proxy-server-based approach to end-to-end video delivery over wide-area networks," *IEEE/ACM Transactions on Networking*, vol. 8, no. 4, Aug. 2000.
8. Mario Baldi and Yoram Ofek, "End-to-end delay analysis of videoconferencing over packet switched networks," *IEEE/ACM Transactions on Networking*, vol. 8, no. 4, Aug. 2000.
9. William E. Naylor and Leonard Kleinrock, "Stream traffic communication in packet switched networks: destination buffering constraints," *IEEE Transactions on Communications*, vol. COM-30, no. 12, pp. 2527–2534, Dec. 1982.

10. Giulio Barberis and Daniele Pazzaglia, "Analysis and optimal design of a packet-voice receiver," *IEEE Transactions on Communications*, vol. COM-28, no. 2, pp. 217–227, Feb. 1980.
11. Warren A. Montgomery, "Techniques for packet voice synchronization," *IEEE Journal on Selected Areas in Communications*, vol. SAC-1, no. 6, pp. 1022–1028, Dec. 1983.
12. Felipe Alvarez-Cuevas, Miquel Bertran, Francesc Oller, and Josep M. Selga, "Voice synchronization in packet switching networks," *IEEE Network*, vol. 7, no. 5, pp. 20–25, Sept. 1993.
13. Ramachandran Ramjee, Jim Kurose, Don Towsley, and Henning Schulzrinne, "Adaptive playout mechanisms for packetized audio applications in wide-area networks," in *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, Toronto, Canada, June 1994, pp. 680–688, IEEE Computer Society Press, Los Alamitos, California.
14. Sue B. Moon, Jim Kurose, and Don Towsley, "Packet audio playout delay adjustment: performance bounds and algorithms," *ACM/Springer Multimedia Systems*, vol. 5, no. 1, pp. 17–28, Jan. 1998.
15. Maria C. Yuang, Shih T. Liang, Yu G. Chen, and Chi L. Shen, "Dynamic video playout smoothing method for multimedia applications," in *Proceedings of the IEEE International Conference on Communications (IEEE ICC)*, Dallas, Texas, June 1996, p. S44.
16. Nikolaos Laoutaris and Ioannis Stavrakakis, "Adaptive playout strategies for packet video receivers with finite buffer capacity," in *Proceedings of the IEEE International Conference on Communications (IEEE ICC)*, Helsinki, Finland, June 2001.
17. Maria C. Yuang, Po L. Tien, and Shih T. Liang, "Intelligent video smoother for multimedia communications," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 2, pp. 136–146, Feb. 1997.
18. Hang Liu and Magda El Zarki, "Delay and synchronization control middleware to support real-time multimedia services over wireless PCS networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 9, pp. 1660–1671, Sept. 1999.
19. Mark Claypool and Jonathan Tanner, "The effects of jitter on the perceptual quality of video," in *ACM Multimedia '99*, Orlando, FL, USA, 1999.
20. Leonard Kleinrock, *Queueing Systems Volume I: Theory*, Wiley-Interscience, New York, 1975.
21. Ronald W. Wolff, *Stochastic Modeling and the Theory of Queues*, Prentice Hall, New Jersey, 1988.
22. Frederick S. Hillier and Gerald J. Lieberman, *Introduction to Operations Research*, McGraw-Hill, 2000.
23. O. Berman and K.P. Sapna, "Optimal control of service for facilities holding inventory," *Computers & Operations Research*, vol. 28, no. 5, Apr. 2001.