# Model for Unbalanced Multiple Description Video Transmission using Path Diversity

Sila Ekmekci and Thomas Sikora

Technical University Berlin, Einsteinufer 17, Berlin, 10587 Germany

## ABSTRACT

Multiple State Video Coding (MSVC) is a Multiple Description Coding Scheme where the video is coded into multiple independently decodable streams, each with its own prediction process and state. The system subject to this work is composed of two subsystems: 1- multiple state encoding/decoding, 2- path diversity transmission system. In [1] we discussed how to optimize the rate allocation of such a system while maximizing the average reconstructed frame PSNR at the decoder and minimizing the PSNR variations between the streams given the total bitrate $R_T$ and the balanced (equal loss probabilities) or unbalanced (unequal) loss probabilities $p_1$ and $p_2$ over the two paths. In our current work we establish a theoretical framework to estimate the rate-(decoder) distortion (R-Dd) function and have taken into account the MSVC-structure, the rate allocation, channel impairments and reconstruction strategies respectively. The video sequence is modeled as an AR(1) source and the distortion associated with each reconstructed frame in both threads in a lossy transmission environment is estimated recursively depending on the system parameters.

**Keywords:** model-based, multiple description, video transmission, path diversity, AR(1) model, recursive distortion estimation, error propagation

## 1. INTRODUCTION

Multiple Description Coding (MDC) is a forward error correction scheme where two or more descriptions of the source are sent to the receiver over different channels. If only one channel is received the signal can be reconstructed with distortion $D_1$ or $D_2$. If both channels are received on the other hand, the combined information is used to achieve a lower distortion $D_0$. Low delay property makes MDC interesting, especially for video streaming applications.

Multiple State Video Coding presented in [2], also an MDC approach, is composed of two subsystems: 1- multiple state encoding/decoding, 2- a path diversity transmission system. The video is coded into multiple independently decodable streams each with its own prediction process and state. The advantages are: 1- if one stream is lost the other stream is still decodable, 2- the correctly received stream can enable state recovery for the corrupted stream using bidirectional information from past and future frames. The path diversity system on the other hand enables the submission of different subsets of packets over different paths. The block diagram of MSVC is given in Figure 1, Figure 2 shows the state recovery method. The assumption is that the probability of simultaneous losses on two paths is smaller than the individual loss probabilities of the paths. We additionally assume that the loss probabilities of the chosen paths are independent. On the other hand the distance between video frames on the same thread increases with increasing number of descriptions causing a decrease in prediction gain as compared to single description coding of the same sequence. Error resilience however increases due to the state recovery ability of the system. To sum up we tradeoff the conflicting requirements of high coding gain and error resilience. In a two-description system we split the video sequence into the sequences of even and odd numbered frames and code both sequences independently. If both streams are received without errors the decoding is performed also independently and interleaved at the end to produce the final stream. However if there are losses on one of the streams the lost frames are reconstructed by interpolation of past and future frames from the other stream.

Further author information: (Send correspondence to S.Ekmekci)
S.Ekmekci: E-mail: ekmekci@nue.tu-berlin.de, Telephone: ++49 30 314-23835

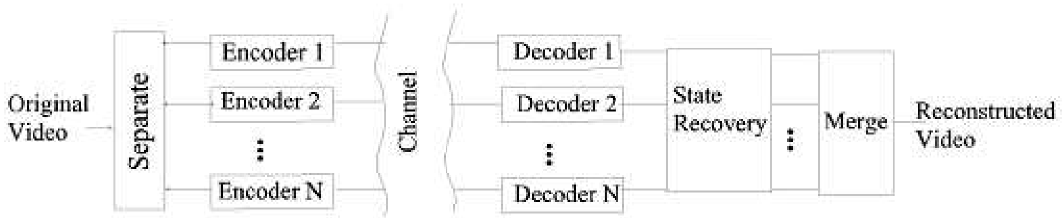


**Figure 1.** Block diagram of the Multiple State Video Coding Approach (after John Apostolopoulos)

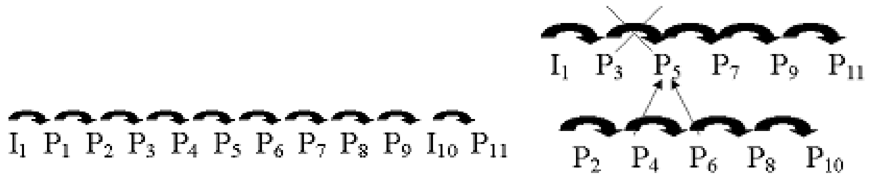


**Figure 2.** a) Prediction in Conventional Coding b) Prediction and State Recovery in Multiple State Video Coding

To unbalance the bitrates of two descriptions frame rate, spatial resolution or quantization of the descriptions can be adapted. A system based on frame rate adaptation is introduced in [3]. The work in [1] on the other hand introduces an unbalanced MD video coding scheme based on quantization adaptation. The goal there is to find the optimal rate allocation $(R_1, R_2)$ given the total bitrate $R_T$ and the loss probabilities $p_1$ and $p_2$ to maximize the average PSNR $PSNR_{avg}$ and the minimize the PSNR variations, flickers $PSNR_{diff}$ between them. Here the video sequence is splitted into two sequences (one for odd and one for even frames) and coded with different combinations of quantization stepsizes $(Q_1, Q_2)$ yielding the same total bitrate $R_T$. To evaluate the system performance at different loss probability pairs the encoded odd and even sequences are decoded and reconstructed at 20 different random channel realizations for each loss probability pair $(p_1, p_2)$. It is assumed that entire frame is lost when a packet is lost.

The goal of the current work is to establish a theoretical framework for the system introduced in [1]. Instead of coding and decoding the video sequences at different bitrates and channel realizations to analyze the system performance, we modeled the video sequence and the reconstruction algorithm to estimate the individual frame distortions as well as the average PSNR $PSNR_{avg}$ over all frames based on the theoretical framework. This helps us to ensure the encoder is working smoothly on the optimal R-Dd function given the loss probabilities $p_1$, $p_2$ as known. We start our work by introducing a recursive distortion estimation model for a one dimensional AR(1) source. Next we extend the model to estimate the frame distortions of a video sequence. Here we model the corresponding blocks in a sequence as samples from a given AR(1) source. The estimated distortion for each block of a frame is calculated recursively.

Section 2 introduces the recursive distortion estimation model for a one-dimensional AR(1) source while section 3 details the extended method for video signals. The comparison of the results produced by the model and the simulations are presented in section 4 whereas section 5 concludes the paper.

## 2. MULTI-STREAM DISTORTION ESTIMATION FOR ONE-DIMENSIONAL AR(1) SOURCE

In this section we analyze the distortion occurring during the multi-stream transmission of a one-dimensional AR(1) source. We assume having two transmission paths with independent loss probabilities $p_1$ and $p_2$. The signal to be transmitted $x(n)$ is splitted into two, where the odd samples are transmitted over the channel with loss probability $p_1$ whereas the even samples are sent over the other path and lost with probability $p_2$. The lost samples are substituted by bilinear interpolation using their previous and next samples. The two streams constituted by the odd and even samples respectively are coded with different bitrates $R_1$ and $R_2$ resulting in quantization error variances $\sigma_{q1}^2$ and $\sigma_{q2}^2$. Our aim is to model the average distortion accumulated on each stream due to losses and associated concealments through frame interpolation/repetition, as well as due quantization errors. Modeled average distortions over all possible bitrate allocations $(R_1, R_2)$ (where $R_T = R_1 + R_2$ for different combinations of $(p_1, p_2)$) will provide insight into optimal bitrate allocation under different transmission conditions.

The AR(1) source is described by $x(n) = a_1 x(n-1) + z(n)$ where $a_1 = \rho_1$ is the normalized autocorrelation coefficient of the source and $z(n)$ is a zero mean uncorrelated source. According to that the odd and even splitted sources can be described by:

$$\begin{aligned} x_1(n_1) &= a_2 x_1(n_1 - 1) + z_1(n_1) \\ x_2(n_1) &= a_2 x_2(n_1 - 1) + z_2(n_1) \end{aligned} \tag{1}$$

where $a_2 = a_1^2$ and $z_1(n_1)$ and $z_2(n_1)$ are zero mean uncorrelated sources. To simplify further calculations without loss of generality we will assume that $\sigma_x^2 = 1$. $x(n)$ and $z(n)$ have zero means and $\sigma_z^2 = 1 - a_1^2$. $z_1(n_1)$ and $z_2(n_1)$ are odd and even splitted parts of $z(n)$ respectively. Differential Pulse Code Modulation (DPCM) with optimal prediction is used to code the samples, i.e.:

$$\begin{aligned} d_1(n_1) &= x_1(n_1) - a_2 x_1(n_1 - 1) \\ d_2(n_1) &= x_2(n_1) - a_2 x_2(n_1 - 1) \end{aligned} \tag{2}$$

Our goal is to estimate the accumulated distortion recursively on each sample given $(R_1, R_2)$ and $(p_1, p_2)$. We verified the model via simulations generating $x(n)$ as an AR(1) source with the predefined $a_1$ and simulating the transmissions with loss probabilities $p_1$ and $p_2$.

**Estimation of Distortion if sample x(n) is lost:**
Suppose the sample $x(n)$ on the second stream is lost and has to be interpolated using $x(n-1)$ and $x(n+1)$. The error on the $n^{th}$ sample $e(n)$ needs to be estimated. $e(n)$ depends on $e(n-1)$ and $e(n+1)$ as well as on $\sigma_{interp}^2$ which is a function of $a_1$. $D(n) = \sigma_{interp}^2$ when $e(n-1) = e(n+1) = 0$. Assuming bilinear interpolation $x(n) = \dfrac{x(n-1) + x(n+1)}{2}$ the error $e(n)$ is given as $e(n) = \dfrac{e(n-1) + e(n+1)}{2} + e_{interp}$. We assume that $e_{interp}$ is independent of the rest of the error term. The samples $x(n+1)$ and $x(n-1)$ are received with a probability of $(1 - p_1)$. The error terms $e(n-1)$ and $e(n+1)$ include not only the quantization errors but also the through error propagation accumulated errors. Besides, $e(n+1)$ can be formulated in terms of $e(n-1)$:

$$e(n+1) = a_2(e(n-1) - q_1(n-1)) + q_1(n+1) \tag{3}$$

where $q_1(n-1)$ and $q_1(n+1)$ are the quantization errors on $x(n-1)$ and $x(n+1)$ respectively (subscript 1 denotes the first stream). The error $q_1(i-1)$ is compensated on the prediction loop so that the sender and the receiver work with the same prediction values if no losses occur. Moreover the through error propagation accumulated residual error is filtered. $e(n)$ is then calculated as:

$$\begin{aligned} e(n) &= \frac{e(n-1) + e(n+1)}{2} + e_{interp} \\ &= \frac{(1 + a_2)e(n-1) - a_2 q_1(n-1) + q_1(n+1)}{2} + e_{interp} \end{aligned} \tag{4}$$

Next we calculate the distortion $D(n)$ on $x(n)$:

$$
\begin{aligned}
D(n) &= E[e^2(n)] \\
&= \frac{(1+a_2)^2 D(n-1) + a_2^2\sigma_{q1}^2 + \sigma_{q1}^2 - 2a_2(1+a_2)(1-p_1)\sigma_{q1}^2}{4} + \sigma_{interp}^2 \\
&= \frac{(1+a_2)^2}{4} D(n-1) + \frac{1+a_2^2}{4}\sigma_{q1}^2 - \frac{a_2(1+a_2)(1-p_1)\sigma_{q1}^2}{2} + \sigma_{interp}^2
\end{aligned} \tag{5}
$$

If $x(n-1)$ is received with a probability of $1-p_1$, $e(n-1)$ has a component equal to $q_1(n-1)$ whereas the residual error $e(n-1) - q_1(n-1)$ is uncorrelated with $q_1(n-1)$, yielding:

$$
E[e(n-1)q(n-1)] = (1-p_1)\sigma_{q1}^2 \tag{6}
$$

Moreover $\sigma_{interp}^2$ can be calculated as:

$$
\begin{aligned}
\hat{x}(n) &= \frac{x(n-1) + x(n+1)}{2} \\
&= \frac{x(n-1) + a_1 x(n) + z(n+1)}{2} \\
e_{interp} &= x(n) - \hat{x}(n) \\
&= x(n) - \frac{a_1}{2}x(n) - \frac{z(n+1)}{2} - \frac{x(n-1)}{2} \\
e_{interp} &= x(n)(1 - \frac{a_1}{2}) - \frac{x(n-1)}{2} - \frac{z(n+1)}{2} \\
\sigma_{interp}^2 &= E[e_{interp}^2] \\
&= (1 - \frac{a_1}{2})^2 + \frac{1}{4} + \frac{1-a_1^2}{4} - a_1(1 - \frac{a_1}{2}) \\
&= (1-a_1)^2 + \frac{1}{2}(1-a_1^2)
\end{aligned} \tag{7}
$$

where $\hat{x}(n)$ is the interpolated sample.

**Estimation of Distortion if samples x(n) and x(n-2) are both received:**

x(n) is reconstructed as $\tilde{x}(n)$:

$$
\begin{aligned}
\tilde{x}(n) &= a_2\tilde{x}(n-2) + d(n) + q_2(n) \\
e(n) &= a_2(e(n-2) - q_2(n-2)) + q_2(n) \\
D(n) &= E[e^2(n)] = a_2^2 D(n-2) + a_2^2\sigma_{q2}^2 + \sigma_{q2}^2 - 2a_2^2\sigma_{q2}^2 \\
&= a_2^2 D(n-2) + (1-a_2^2)\sigma_{q2}^2
\end{aligned} \tag{8}
$$

where $d(n)$ is the difference value sent to the receiver and $\sigma_{q2}^2$ is the quantization error variance of the second stream where $x(n)$ was coded and transmitted.

**Estimation of Distortion if sample x(n) is received but x(n-2) is lost:**

$$
\begin{aligned}
\tilde{x}(n) &= a_2\tilde{x}(n-2) + d(n) + q_2(n) \\
e(n) &= a_2(e(n-2) - q_2(n-2)) + q_2(n) \\
D(n) &= E[e^2(n)] = a_2^2 D(n-2) + a_2^2\sigma_{q2}^2 + \sigma_{q2}^2 - 2a_2^2 E[e(n-2)q_2(n-2)] \\
&= a_2^2 D(n-2) + (1+a_2^2)\sigma_{q2}^2
\end{aligned} \tag{9}
$$

The calculation of $D(n-2)$ is performed as follows:

$$
\begin{aligned}
\tilde{x}(n-2) &= \frac{\tilde{x}(n-1)+\tilde{x}(n-3)}{2} \\
\tilde{x}(n-1) &= x(n-1)+e(n-1) \\
\tilde{x}(n-3) &= x(n-3)+e(n-3) \\
e(n-2) &= x(n-2)-\tilde{x}(n-2) \\
&= x(n-2)-\frac{x(n-1)+e(n-1)+x(n-3)+e(n-3)}{2} \\
D(n-2) &= \frac{3}{2}+\frac{D(n-1)}{4}+\frac{D(n-3)}{4}-2a_1+\frac{a_1^2}{2}+\frac{a_1^2}{2}D(n-3)(1-p_1) \\
&= \frac{3}{2}+\frac{D(n-1)}{4}+\frac{D(n-3)}{4}-a_1-a_1^3+\frac{a_1^2}{2}+(1-p_1)a_1^2D(n-3) \qquad (10)
\end{aligned}
$$

where $E[e(n-1)e(n-3)] = (1-p_1)a_1^2D(n-3)$ and $(1-p_1)$ is the probability that $x(n-3)$ was not lost. $D(n)$ is calculated by combining Equations 9 and 10:

$$
D(n) = \frac{3}{2}a_1^4+\frac{a_1^4}{4}D(n-3)+\frac{a_1^4}{4}D(n-1)-2a_1^5+\frac{a_1^6}{2}[1+(1-p_1)D(n-3)]+(1+a_1^4)\sigma_{q2}^2 \qquad (11)
$$

**Estimation of Distortion if sample x(n) is lost but x(n+1) is received:**
$\tilde{x}(n)$ is constructed by interpolating $\tilde{x}(n-1)$ and $\tilde{x}(n+1)$.

$$
\begin{aligned}
\tilde{x}(n) &= \frac{\tilde{x}(n-1)+\tilde{x}(n+1)}{2} \\
&= \frac{x(n-1)+x(n+1)+e(n-1)+e(n+1)}{2} \\
e(n) &= x(n)-\frac{x(n-1)+x(n+1)+e(n-1)+e(n+1)}{2} \\
&= e_{interp}+\frac{e(n-1)+e(n+1)}{2} \\
&= e_{interp}+\frac{e(n-1)+a_1^2e(n-1)-a_1^2q_1(n-1)+q_1(n+1)}{2} \\
D(n) &= \sigma_{interp}^2+\frac{(1+a_1^2)^2}{4}D(n-1)+\frac{(1+a_1^4)}{4}\sigma_{q1}^2-(1-p_1)\frac{(1+a_1^2)a_1^2}{2}\sigma_{q1}^2 \qquad (12)
\end{aligned}
$$

where $E[e(n-1)q(n-1)] = (1-p_1)\sigma_{q1}^2$ and $(1-p_1)$ is the probability that $x(n-1)$ was not lost.

**Estimation of Distortion if samples x(n) and x(n+1) are both lost:**

The reconstruction method in this case is the repetition of the last sample.

$$
\begin{aligned}
\tilde{x}(n) &= \tilde{x}(n-1) \\
e(n) &= e(n-1)+e_{rep} \\
D(n) &= D(n-1)+\sigma_{rep}^2 \qquad (13)
\end{aligned}
$$

with

$$
\begin{aligned}
e_{rep}(n) &= x(n)-x(n-1) \\
\sigma_{rep}^2 &= E[e_{rep}^2(n)] = 1+1-2a_1 = 2(1-a_1) \qquad (14)
\end{aligned}
$$

Here we assumed that $e(n-1)$ is not correlated to $x(n-1)$ or $x(n)$ which does not hold as $p_1$,$p_2$ and/or $n$ is increasing. The model can be improved by by taking the correlation with the error term into account.

The overall distortion model for the $n^{th}$ sample (on the second stream) can be summarized as:

$$\begin{aligned} D(n) \;=\; & (1-p_2)(1-p_2)(a_2^2 D(n-2) + (1-a_2^2)\sigma_{q2}^2) \\ & +(1-p_2)p_2(\frac{3}{2}a_1^4 + \frac{1}{4}a_1^4 D(n-3) + \frac{1}{4}a_1^4 D(n-1) + (1+a_1^4)\sigma_{q2}^2 - 2a_1^5 + \frac{1}{2}a_1^6[1+(1-p_1)D(n-3)]) \\ & +p_2(1-p_1)(\sigma_{interp}^2 + \frac{(1+a_1^2)^2}{4}D(n-1) + \frac{(1+a_1^4)}{4}\sigma_{q1}^2 - (1-p_1)\frac{(1+a_1^2)a_1^2}{2}\sigma_{q1}^2) \\ & +p_2 p_1(D(n-1) + \sigma_{rep}^2) \end{aligned} \tag{15}$$

For a sample on the first stream we obtain the symmetrical equation when $p_1$ is exchanged with $p_2$ and $\sigma_{q1}^2$ with $\sigma_{q2}^2$. The first part of Equation 15 considers the case that the $n^{th}$ and the $(n-2)^{th}$ samples both are received. The second part belongs to the case that the $n^{th}$ sample is received and the $(n-2)^{th}$ sample is lost whereas, the third part to the case that the $n^{th}$ lost and the $(n+1)^{th}$ is received. The last and fourth line deals with the case that both $n^{th}$ and $(n+1)^{th}$ are lost.

Figures 3 and 4 compares the results obtained via model and simulations. For simulations several one-dimensional AR(1) sources with correlation coefficient $a_1 = 0.9$ and variance $\sigma_x^2 = 1$ are generated and multi-stream transmission is simulated using 1000 different random loss patterns for each loss probability $p_1$ and $p_2$. The average PSNR value $PSNR_{avg}$ (for given bitrates $R_1$ of the first stream) are calculated by averaging the obtained average PSNR values over all samples and over all simulated transmissions. As the loss probabilities $p_1$ and $p_2$ increase the difference between the simulation and model results increases for large $n$. The main reason for this gap is the uncorrelation assumption of error terms from the sample values after a certain point. This however could be taken into account by increasing the depth of recursion.

Figure 3 depicts the case of unbalanced loss probabilities where the first channel is lossless. The total bitrate is 0.8 bits/sample. As $R_1$ increases the average PSNR of the $(0\%, 20\%)$ case exceeds that of the $(0\%, 5\%)$ case. Figure 4 on the other hand, shows the case of balanced loss probabilities. Since $p_1$ is not equal to zero anymore, the last term of Equation 15 becomes important and the gap between simulation and model increases.

## 3. EXTENSION OF THE MODEL FOR VIDEO

In this section we extend our R-Dd model for video sequences. Here he corresponding blocks from each frame along the sequence are considered as samples of an AR(1) source while each block in a frame belongs to a different AR(1) source. Recalling our model in section 2 the prediction coefficient between corresponding blocks is $a_2 = 1$ this time. As in section 2 each frame is classified as odd or even and sent on path 1 or path 2 based on this strategy. According to that each second corresponding block is sent on the same path with the same error probability. But before treating the blocks as one dimensional-samples we recall that the pixels in a block are bilinearly interpolated to generate subpixel accuracy for motion estimation/compensation. We assume that the interpolation filter is the only loop filter and will use quarter pixel accuracy for our analysis. The analysis for half pixel or one eighth pixel accuracy is performed similarly. Assuming that the main source of error (besides quantization error) on each pixel is the due to error concealment through frame interpolation (motion based), we define a constant $\alpha_{interp}$ which gives the attenuation of the propagated distortion due to the subpixel motion compensation (loop filter). In section 3.1 we show why $\alpha_{interp}$ can be approximated as a constant over time for a block in a frame while each block has a different $\alpha_{interp}$ depending on its horizontal and vertical correlation coefficients $\rho_h$ and $\rho_v$ respectively. Below we will discuss the four identical reconstruction scenarious investigated in section 2 where $x$ is the sequence of corresponding blocks and $n$ is the frame number the block belongs to. $D(n)$ gives the distortion of block $x(n)$ in frame $n$ whereas $D(n-2)$ gives the distortion of its corresponding block $x(n-2)$ in frame $n-2$.

**Estimation of Distortion if blocks x(n) and x(n-2) are both received:**

$$D(n) = \alpha_{inter}(D(n-2) - \sigma_{q2}^2) + \sigma_{q2}^2 \tag{16}$$

The residual error $e(n-2) - q_2(n-2)$ is filtered by the loop filter which attenuates its energy by $\alpha_{inter}$.

**Estimation of Distortion if block x(n) is received but x(n-2) is lost:**

$$\begin{aligned} e(n) &= \sqrt{\alpha_{interp}}[\frac{x(n-3)+e(n-3)+x(n-1)+e(n-1)}{2} - x(n-2) - q_2(n-2)] + q_2(n) \\ &= \sqrt{\alpha_{interp}}[e_{interp}(n-2) + \frac{e(n-1)+e(n-3)}{2} - q_2(n-2)] + q_2(n) \\ D(n) &= E[e^2(n)] \\ &= \sigma_{interp}^2 + (1+\alpha_{interp})\sigma_{q2}^2 + \frac{\alpha_{interp}}{4}D(n-3) + \frac{\alpha_{interp}}{4}D(n-1) + \frac{\alpha_{interp}^{\frac{3}{2}}}{2}D(n-3)(1-p_1) \end{aligned} \tag{17}$$

where $e_{interp}(n-2) = \frac{x(n-3)+x(n-1)}{2} - x(n-2)$.

The interpolation error variance $\sigma_{interp}^2$ is different for each block of each frame. For the results presented in section 4 we interpolated (motion controlled) the current frame using the previous and the next frames. The interpolated frame is subtracted from the original one yielding the interpolation error. Using the above formula in each recursion with the same $\sigma_{interp}^2$ assumes that the interpolating for corresponding blocks yields the same interpolation error at every frame. This is in practice not quite correct and depends on the motion vector fields as well as on the temporal correlation coefficient $\rho_t$ of corresponding blocks.

**Estimation of Distortion if block x(n) is lost but x(n+1) is received:**

$$D(n) = \frac{(1+\sqrt{\alpha_{interp}})^2}{4}D(n-1) + \frac{1+\alpha_{interp}}{4}\sigma_{q1}^2 - \frac{(1+\sqrt{\alpha_{interp}})\sqrt{\alpha_{interp}}}{2}\sigma_{q1}^2(1-p_1) + \sigma_{interp}^2 \tag{18}$$

**Estimation of Distortion if blocks x(n) and x(n+1) are both lost:**

$$D(n) = D(n-1) + \sigma_{rep}^2 \tag{19}$$

The repetition error variance $\sigma_{rep}^2$ is different for each block of each frame. For results depicted in section 4 the original frame is subtracted from the preceding one yielding the repetition error. From here the repetition error variance can be calculated for each individual block. Using the formula recursively with the same $\sigma_{rep}^2$ assumes that the repetition for each corresponding block using the corresponding reference blocks will yield the same interpolation error which is not correct. As $p_1$, $p_2$ and $N$ (the length of the video sequence) increases, this last part of the equation produces smaller values than the real values occurring during the real transmission and decoding. To avoid this we should consider a longer history by extending the last term of the formula. This point will be considered in further work. For our current work we scaled the last term by a factor to consider the inaccuracy in $\sigma_{rep}^2$ depending on the loss rates of the streams.

### 3.1. Estimation of Error Propagation in Multi-Stream Video Transmission

We will mainly discuss four different cases and analyze how the interpolation error (the main source of error in multi-state recovery) is affected by the loop filter which performs nothing but subpixel interpolation. Denoting the errors on neighboring pixels (horizontal or vertical) on pixel grid by $e_1(n-1)$ and $e_2(n-1)$ with variance $D(n-1)$ over all pixels in the block, the error on possible four positions on the subpixel grid is calculated as listed below. According to the position $x$ we analyzed the resulting error variance in three classes, where positions 2 and 4 are affected in the same way and therefore analyzed in the same class. The attenuation is analyzed separately in horizontal and vertical directions. To ease the analysis we assume bilinear interpolation in horizontal and vertical directions for all positions. $e_x(n-1)$ will be denoted as $e_x$ in the sequel. $\alpha_{1,interp}$ is the attenuation in one direction whereas $\alpha_{interp}$ is the combined attenuation in both directions (product of one-dimensional attenuations).

*Case 1, x=1*

Only the pixels belonging to the grid of the original image are used at motion compensation, i.e. $\alpha_{1,interp} = 1$. The error variance $D(n-1)$ is not changed for this block.

*Case 2, x=2,4* The subpixel used in motion compensation is one quarter pixel distance away from one of the original pixels and three quarters from the next one.

$$\begin{aligned} e_x &= \frac{3}{4}e_1 + \frac{1}{4}e_2 \\ \alpha_{1,interp} &= \frac{\sigma^2_{e,x}}{\sigma^2_{e,1}} \\ &= \frac{9}{16} + \frac{1}{16} + \frac{6}{16}\rho \\ &= \frac{5}{8} + \frac{3}{8}\rho \end{aligned} \tag{20}$$

where $\rho$ is the horizontal or vertical correlation coefficient between adjacent error terms.

*Case 3, x=3* The subpixel is in the middle of two original pixels.

$$\begin{aligned} e_x &= \frac{1}{2}e_1 + \frac{1}{2}e_2 \\ \alpha_{1,interp} &= \frac{1}{2} + \frac{1}{2}\rho \end{aligned} \tag{21}$$

Which of the above cases will occur depends on the particular motion vector in horizontal and vertical direction. We assume that these all 16 (all combinations of 4 horizontal and 4 vertical positions) occur with the same probability. According to that the expected value of $\alpha_{interp}$ is calculated as follows:

$$\begin{aligned} E[\alpha_{interp}] &= \frac{1}{16}(1 + 2[\frac{5}{8} + \frac{3}{8}\rho_h] + [\frac{1}{2} + \frac{1}{2}\rho_h])(1 + 2[\frac{5}{8} + \frac{3}{8}\rho_v] + [\frac{1}{2} + \frac{1}{2}\rho_v]) \\ &= \frac{1}{256}(121 + 55\rho_h + 55\rho_v + 25\rho_h\rho_v) \end{aligned} \tag{22}$$

Since blocks in a frame have different $\rho_h$ and $\rho_v$ values, $E[\alpha_{interp}]$ should be calculated separetely for each. $E[\alpha_{interp}]$ will be denoted by $\alpha_{interp}$ in the sequel. Since the errors terms on blocks are assumed to be the accumulation of several error terms, the total error energy will be attenuated by the same value. In subsection 3.2 we show that the interpolation error on a block has the same $\rho_h$ and $\rho_v$ values as the block. The error terms constituted just by quantization are attenuated by $E[\alpha_{interp}] = \frac{121}{256} = 0.4727$ by setting $\rho_h = \rho_v = 0$ in Equation 22.

### 3.2. Analysis of the Interpolation Error

In subsection 3.1 we assumed that adjacent interpolation error values $e_1$ and $e_2$ are correlated with the same correlation coefficients $\rho_h$ and $\rho_v$ as the block pixels themselves. Here we will show that this assumption holds. The total distortion accumulated in each block is the sum of distortions due to previous interpolations. In the following $c_x$ is the original sample which is to be reconstructed whereas $a_x$ and $b_x$ are the corresponding pixels from the previous and the next frames. $x$ and $x-1$ denote the neighbouring pixel positions in the block.

$$\sigma^2_{e,x} = R_{e,x}(0)$$

$$
\begin{aligned}
&= (c_x - \frac{a_x}{2} - \frac{b_x}{2})^2 \\
&= 1 + \frac{1}{4} + \frac{1}{4} - \rho_t - \rho_t + \frac{\rho_t^2}{2} \\
&= \frac{(\rho_t - 3)(\rho_t - 1)}{2}
\end{aligned}
\tag{23}
$$

where $\rho_t$ is the temporal correlation coefficient between corresponding blocks in adjacent frames. Similarly we calculate

$$
\begin{aligned}
R_{e,x}(1) &= E[(c_x - \frac{a_x}{2} - \frac{b_x}{2})(c_{x-1} - \frac{a_{x-1}}{2} - \frac{b_{x-1}}{2})] \\
&= \rho - \frac{\rho\rho_t}{2} - \frac{\rho\rho_t}{2} - \frac{\rho\rho_t}{2} + \frac{\rho}{4} + \frac{\rho_t^2\rho}{4} - \frac{\rho\rho_t}{2} + \frac{\rho_t^2\rho}{4} + \frac{\rho}{4} \\
&= \frac{\rho(\rho_t - 3)(\rho_t - 1)}{2}
\end{aligned}
\tag{24}
$$

Combining Equations 23 and 24 we obtain:

$$
\begin{aligned}
\rho_{e,x} &= \frac{R_{e,x}(1)}{R_{e,x}(0)} \\
&= \rho
\end{aligned}
\tag{25}
$$

$\rho$ is the horizontal or the vertical correlation coefficient, the analysis can be performed separately in both directions.

## 4. EXPERIMENTAL RESULTS: MODEL VS. SIMULATION

In this section we compare the R-Dd curves of video sequences obtained via simulation and model. We performed the comparisons for sequences Foreman and Suzie and present here the results for the first 100 frames (50 frames in each stream) of the Foreman sequence. Figures 5 and 6 show the average PSNR $PSNR_{avg}$ over $R_1$, rate allocated to the first stream. Figure 5 depicts the case of balanced loss probabilities $(5\%, 5\%)$, $(10\%, 10\%)$ and $(20\%, 20\%)$ whereas Figure 6 the case of unbalanced loss probabilities $(0\%, 20\%)$, $(5\%, 20\%)$ and $(10\%, 20\%)$. As expected both for simulation and model, balanced loss probabilities favor balanced bitrate allocation which corresponds to $(70kbit/s, 70kbit/s)$ where $R_{total} = 140kbit/s$. As $R_1$ increases unbalance in rate allocation increases. Variations in $PSNR_{avg}$ as unbalance increases are smallest for the $(20\%, 20\%)$ case due to increased interactions of both streams in terms of frame interpolations. As the loss probabilities also become unbalanced we expect that unbalanced bit allocations are preferred compared to balanced bit allocations. This is observed for the $(0\%, 20\%)$ curve both for simulation and model. The drop in $PSNR_{avg}$ after reaching the maximum point is greater for model results compared to simulation results. As loss rate difference between streams decreases maximum point is shifted toward symmetric rate allocations. Model and simulation results are in good accordance in all cases.

## 5. SUMMARY & CONCLUSION

We established a theoretical framework to estimate the rate-(decoder) distortion (R-Dd) function and have taken the MSVC-structure, the rate allocation, channel impairments and reconstruction strategies into account. First a one-dimensional model for one-dimensional AR(1) sources is developed and then the model is extended for video signals. The distortion associated with each reconstructed frame in both threads in a lossy transmission environment is estimated recursively depending on the system parameters. We modelled the corresponding blocks in a sequence as samples of the same AR(1) source and estimated their distortion in each frame. By performing this for every block of each frame and taking the average of distortion over all blocks of a frame and afterwards over all frames, both the frame PSNR and the average PSNR are calculated respectively. The results presented for one dimensional sources as well as for video show accordance of model with simulations. In further work, the model will be improved by increasing the depth of recursion to control the increasing interactions of streams at increasing loss probabilities and length of data sequences.

## REFERENCES


1. S. Ekmekci and T. Sikora, *Unbalanced Quantized Multiple Description Video Transmission using Path Diversity*, SPIE/Electronic Imaging 2003, Santa Clara, CA, USA, January 2003.
2. J. Apostolopoulos, *Reliable video communication over lossy packet networks using multiple state encoding and path diversity*, VCIP, January 2001.
3. J. Apostolopoulos and S. Wee, *Unbalanced Multiple Description Video Communication Using Path Diversity*, ICIP, October 2001.
4. C. Chen, *Motion-Compensated Hybrid Coders in Video Communications*, Ph.D. Thesis, Monash University, Australia, 1992.

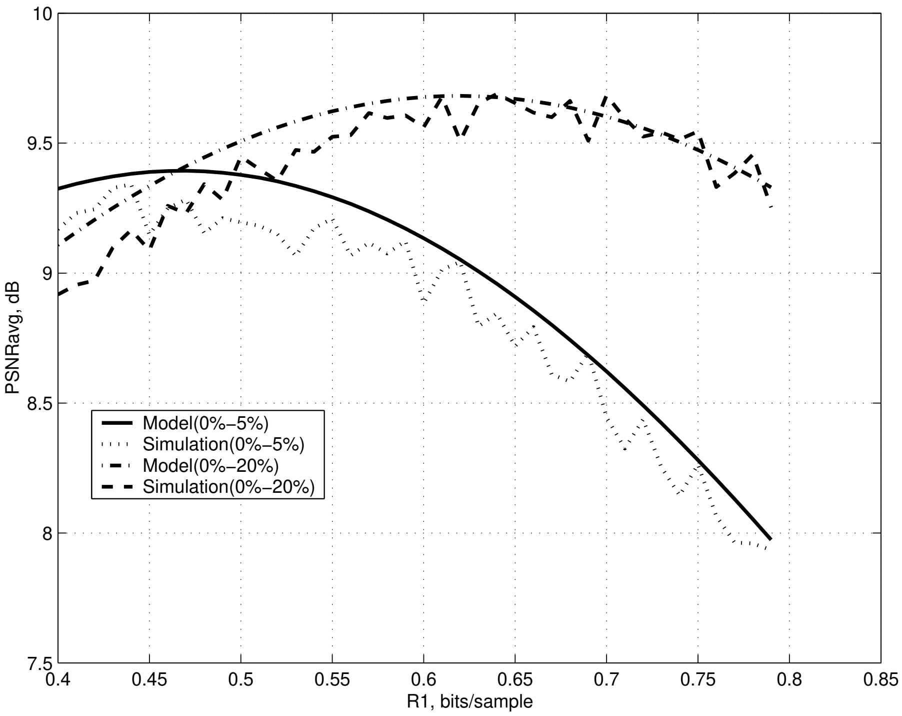


**Figure 3**. Simulation vs. Model, AR(1) source ($a_1 = 0.9$), (0%,5%) & (0%,20%)

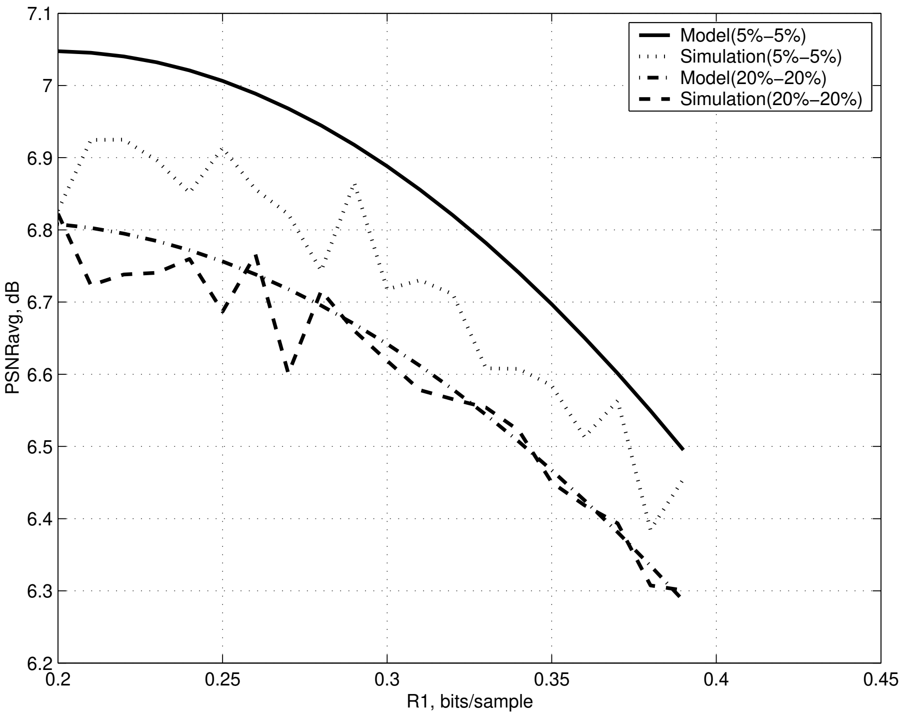


**Figure 4**. Simulation vs. Model, AR(1) source ($a_1 = 0.9$), (5%,5%) & (20%,20%)

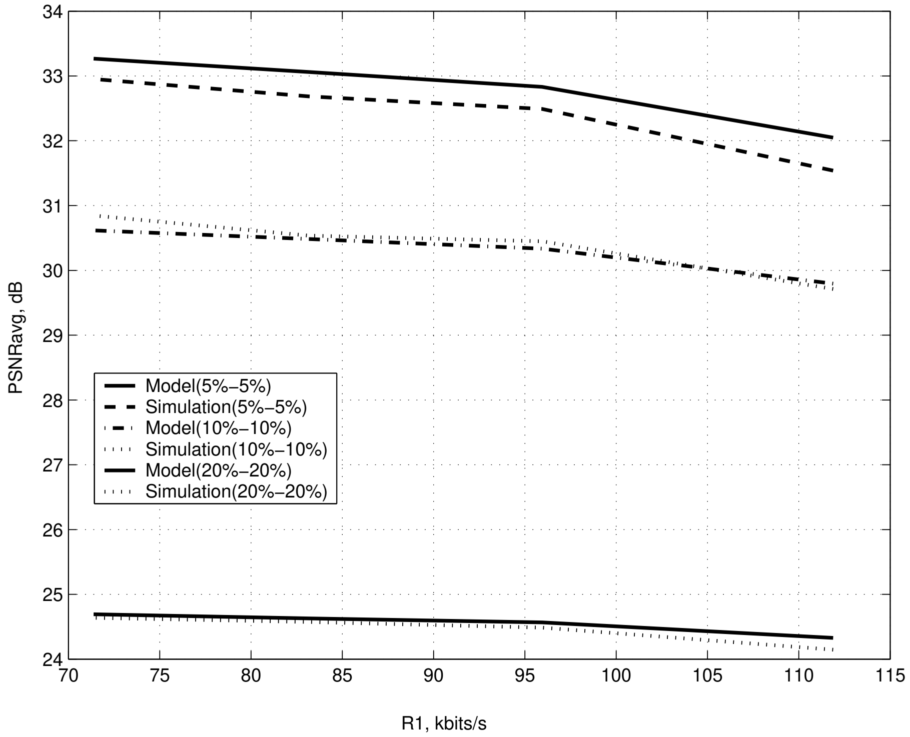


**Figure 5**. Simulation vs. Model, Foreman sequence (100 Frames), balanced loss probabilities

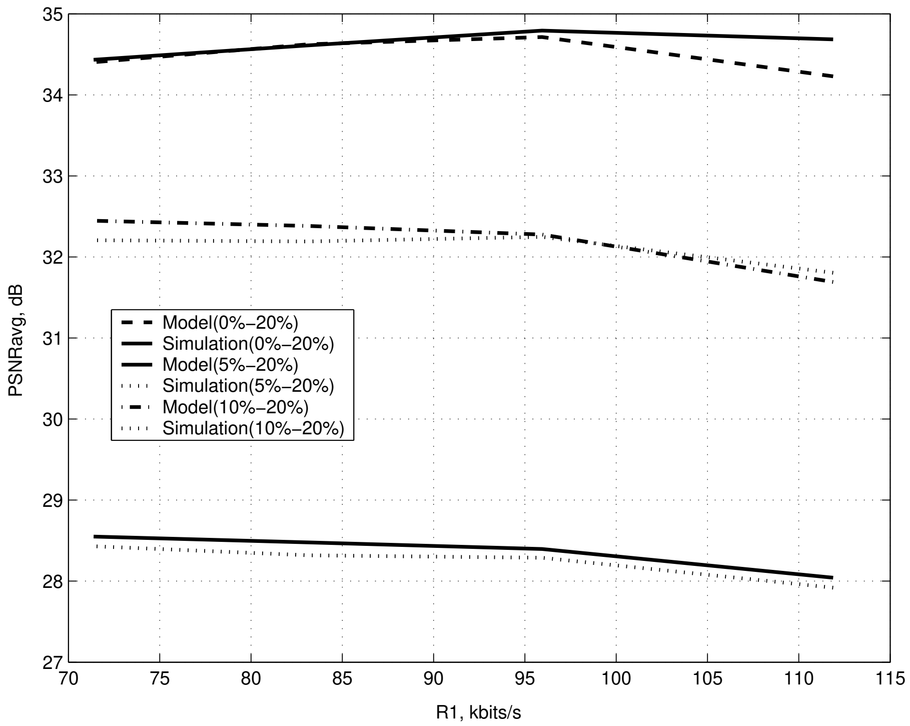


**Figure 6**. Simulation vs. Model, Foreman sequence (100 Frames), unbalanced loss probabilities