# Matching Pursuits Multiple Description Coding for Wireless Video

Xiaoyi Tang and Avideh Zakhor


### Abstract

Multiple description coding (MDC) is an error resilient source coding scheme that creates multiple bitstreams of approximately equal importance. The reconstructed signal based on *any* single bitstream has an acceptable quality. However, a higher quality reconstruction can be achieved with larger number of bitstreams. We develop a multiple (2) description video coding scheme based on the 3 loop structure originally proposed in [1]. We modify the discrete cosine transform structure to the matching pursuits framework and evaluate performance gain using maximum likelihood (ML) enhancement when both descriptions are available. We find that ML enhancement works best for low motion sequences and results in gains of up to 1.3 dB in terms of average PSNR. Rate distortion performance is characterized. Performance comparison is made between our MDC scheme and single description coding (SDC) schemes over lossy channels, including two state Markov channels and Rayleigh fading channels. We find that MDC outperforms SDC in bursty slowly varying environments. In the case of Rayleigh fading channels, interleaving helps SDC close the gap and even outperform MDC depending on the amount of interleaving performed, at the expense of additional delay.


## I. Introduction

Multiple description coding (MDC) generates multiple encoded bitstreams of a source and sends them through several independent channels. The source can be reconstructed at a lower yet acceptable quality from any single bitstream, and at a higher quality from more bitstreams. Unlike layered coding which requires correct reception of the base layer for enhancement layers to be useful, MDC can reconstruct the source from any subset of bitstreams. MDC provides a robust source coding scheme for communication over unreliable channels without quality-of-service (QoS) guarantee.

Early work on MDC was focused on theoretical characterization of the rate distortion bound for two description coding [2], [3]. Since then, there has been a great deal of work on designing practical MDC schemes. Among these schemes, a multiple description scalar quantizer was designed by Vaishampayan [4] and a transform based scheme was developed by Wang, Orchard, and Reibman [5].

More recently, practical MDC (M=2) schemes have been developed for video [1], [6] and their performance over lossy channels has been studied [7], [8]. The challenge in video MDC design arises due to the predictive nature of video. The decoder could receive any subset of all descriptions and the encoder must track state of the decoder to prevent drift. In [6], two independent prediction loops are used in the encoder. The quantizers in both loops are designed such that two descriptions can be combined to yield a higher quality reconstruction. Authors in [1] provide plausibility arguments that this type of scheme yields a poorer prediction when both descriptions are available. In [1], a three prediction loop structure is proposed; there is a central loop for prediction based on both descriptions, and two side loops for prediction based on one description to avoid mismatch between encoder and decoder.

In this paper, we migrate the H.263 based 3 prediction loop MDC scheme in [1] to the Matching Pursuits (MP) video coding framework [9]. MP video coding has generally been shown to outperform discrete cosine transform (DCT) based codecs including H.263 and MPEG-4 [10], [11]. In addition, we improve the rate distortion performance of the scheme by applying enhancement techniques to all three reconstructions when both descriptions are available. Finally, we compare the performance of our MDC scheme with traditional single description coding (SDC) schemes over lossy channels through simulations. We use both two state Markov and Rayleigh fading channel models for simulations.

In Section II, we provide an overview of MP video coding framework and the three loop video MDC scheme based on MP, called MP-MDC. In Section III, we discuss improvements to the base scheme through enhancement from both descriptions. In Section IV, we characterize the rate distortion performance of our MP-MDC scheme. Performance comparisons are made between MDC and SDC over lossy channels next in Section V, and conclusions are presented in Section VI.

## II. Matching Pursuits Video MDC

Our video MDC scheme is based on the three prediction loop structure proposed by Reibman *et al.* [1] and the MP framework [9]. In this section, we provide an overview of MP video coding and the three loop structure.

### A. Overview of Matching Pursuits Video Coding

Almost all existing video coding standards use the hybrid motion-compensated discrete cosine transform (DCT). However, block-based DCT coding introduces noticeable distortion and block artifacts at lower bit rate. In matching pursuits video coding, the residual from motion compensation is decomposed onto a larger basis set than the complete set provided by DCT. At each stage of coding the residual, a search for the best basis function is performed by computing the inner product between the residual and a basis function. The residual is subtracted from the best basis function and the iteration repeats. This ensures that most important features are coded first. Interested readers are referred to [9] for detailed description of matching pursuit video coding.

### B. Three Loop Video MDC

The diagram of the three loop video MDC is shown in Figure 1. In all video coding systems, prediction is essential to achieve video coding efficiency. In two description coding, the decoder could receive either one or two descriptions. Thus there are three prediction loops in the encoder so that the decoder could still track the encoder state when a description is lost.

In the central prediction loop, a new frame is first motion compensated from its prediction based on both descriptions. The residue is then coded into two correlated

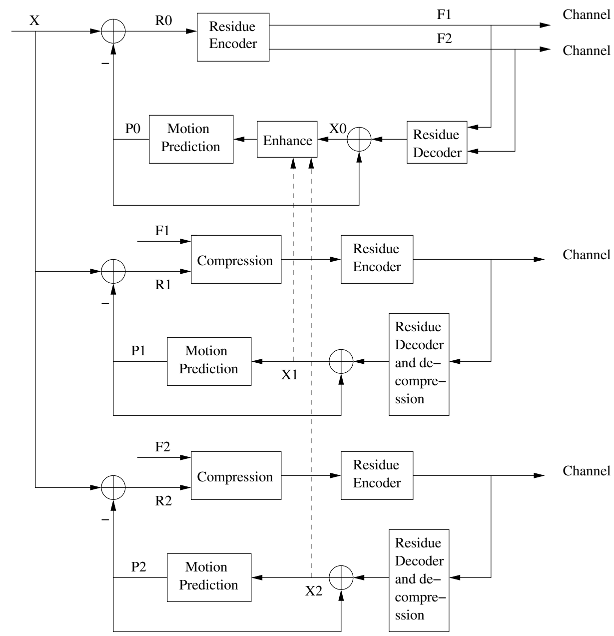



Fig. 1. Diagram of video multiple description coding.

bit streams that are sent to two separate channels. In [1], a DCT transform is first used to transform the residue to the frequency domain, followed by a correlation matrix. In our proposed matching pursuit based scheme, the residue is coded into two sets of atoms[1], F1 and F2, to be sent over two channels. The first L atoms found during MP iterations are shared by both sets and subsequent atoms are alternatively put into the two sets. As a result, F1 and F2 are of approximately equal importance because atoms are found in decreasing order of magnitude in MP iterations. The correlation between these two sets of atoms is controlled by the number of shared atoms L. Motion vectors, frame headers, and intra-coded (I) frames are duplicated and sent through both channels.

In the two symmetric side loops, we also use MP coding. There is no motion estimation in the side loops as the motion vectors are taken from the central loop. The rationale is that motion vectors in the side loops would be very similar to the motion vectors in the central loop. Bits saved in coding new motion vectors in side loops could be used to code the video at better quality. The energy of the residue R1(R2) from compensation based on one description is first reduced by exploiting its correlation with the coded residue from the central loop F1(F2) through pixel-wise subtraction, as in [1].

[1]An atom describes one basis function.

Tradeoff between the quality of reconstruction from two descriptions (PSNR0) and quality of reconstruction from one description (PSNR1) can be controlled in two ways, namely by rate allocation between the central and side loops, as well as the number of shared atoms L between two atom sets F1 and F2; increasing L reduces PSNR0 and increases PSNR1. The tradeoff curves are shown in Section IV, after we examine one improvement to our MDC scheme.

## III. Enhancement Using Three Reconstructions

Besides changing to the MP framework, we also examine an improvement to the original scheme, namely enhancement when both descriptions are available. In [1], the side loops are considered to be pure redundancy and so, when both descriptions are received, the reconstructed video is taken from the central loop. Since each of three prediction loops has its own reconstruction of the source frame, we can combine all three reconstructions to yield a better reconstruction than the one in the central loop[2]. This is a special case of the multi-channel restoration problem [12].

### A. Model

The reconstructed video in each loop can be modeled as

$$y_i(l,m) = x(l,m) + n_i(l,m) \qquad i = 1,2,3 \tag{1}$$

where $i$ is the loop index, $y_i(l,m)$ is the reconstructed pixel value at position $(l,m)$ in loop $i$, $x(l,m)$ is a pixel at $(l,m)$ in a source frame, and $n_i(l,m)$ is the additive noise. In vector notation, let $\overrightarrow{X}$, a $M \times 1$ vector, denote the lexicographical ordering of source image, i.e.

$$\overrightarrow{X} = [x(1,1), x(1,2), x(1,3), \ldots]^T \tag{2}$$

Let

$$\overrightarrow{Y} = [y_1(1,1), y_2(1,1), y_3(1,1), y_1(1,2), y_2(1,2), y_3(1,2), \ldots]^T \tag{3}$$

and

$$\overrightarrow{N} = [n_1(1,1), n_2(1,1), n_3(1,1), n_1(1,2), n_2(1,2), n_3(1,2), \ldots]^T \tag{4}$$

[2]Gain can be expected only if the difference in quality between the central loop reconstruction and side loop reconstructions is not too large.

Rewriting Eq (1) in vector form, we get

$$\overrightarrow{Y} = A\overrightarrow{X} + \overrightarrow{N} \tag{5}$$

where $A$ is a $3M \times M$ matrix defined by

$$A = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & a & 0 & 0 \\ 0 & 0 & .. & 0 \\ 0 & 0 & 0 & a \end{bmatrix} \tag{6}$$

with $a = [1,1,1]^T$.

### B. Performance of ML Enhancement

There are several methods to estimate $\overrightarrow{X}$ given $\overrightarrow{Y}$ in Eq (5), namely least squares (LS), regularized least squares (RLS), Maximum Likelihood (ML), Maximum A-Posteriori (MAP), and Linear Minimum Mean Square Error (LMMSE). We use ML estimation because it outperforms LS and RLS, and is simpler than MAP and LMMSE. Assuming the noise is zero-mean Gaussian[3], ML estimation is given by

$$\hat{\overrightarrow{X}}_{ml} = (A^T R^{-1} A)^{-1} A^T R^{-1} \overrightarrow{Y} \tag{7}$$

where $R = E\{\overrightarrow{N}\overrightarrow{N}^T\}$ is the covariance matrix of $\overrightarrow{N}$.

We use pixel-wise estimation to keep overhead of coding estimated coefficients low. Specifically, we approximate the noise correlation matrix $R$ as a block diagonal matrix

$$R = \begin{bmatrix} R_n & 0 & 0 & 0 \\ 0 & R_n & 0 & 0 \\ 0 & 0 & .. & 0 \\ 0 & 0 & 0 & R_n \end{bmatrix} \tag{8}$$

where $R_n$ is a $3 \times 3$ correlation matrix of noise values in the same pixel location of three reconstructions. Therefore, Eq (7) reduces to

$$\hat{x}_{ml}(l,m) = (a^T R_n^{-1} a)^{-1} a^T R_n^{-1} [y_1(l,m), y_2(l,m), y_3(l,m)]^T \tag{9}$$

The estimation error using ML enhancement is given by

$$\begin{aligned} \epsilon(l,m) &= \hat{x}_{ml}(l,m) - x(l,m) \\ &= (a^T R_n^{-1} a)^{-1} a^T R_n^{-1} [n_1(l,m), n_2(l,m), n_3(l,m)]^T \end{aligned}$$

[3]Strictly speaking, the noise is not Gaussian, though it has a bell-shaped distribution. However, Gaussian model is still used to facilitate analysis. We have experimented with LMMSE as well as ML with Gaussian noise model, and have found the performance difference between the two methods to be very small. Given that the complexity of the algorithm should be kept as simple as possible, and small performance difference between the two algorithms, we believe the use of Gaussian noise model is justified.

Therefore, the squared error is

$$E_{ml} = E\{\epsilon(l,m) * \epsilon(l,m)^T\} = (a^T R_n^{-1} a)^{-1} \tag{10}$$

If the reconstruction in central loop is taken instead, the squared error is

$$E_o = \min_{i=1,2,3} R_n(i,i) = R_n(1,1) \tag{11}$$

assuming index value 1 represents the central loop. Thus, the gain from ML enhancement is

$$G = \frac{E_o}{E_{ml}} = R_n(1,1) a^T R_n^{-1} a \tag{12}$$

Since the decoder does not have information about noise, the encoder must encode the 3 ML coefficients given by the elements of the $1 \times 3$ vector $(a^T R_n^{-1} a)^{-1} a^T R_n^{-1}$. In practice, $R_n$ is computed first in the encoder by averaging samples over an entire frame. Since the sum of three ML coefficients in Eq (9) is 1, only two coefficients are quantized and encoded at 6 bit resolution. It creates a nominal 12 bit overhead for each frame, even with fixed length coding.

To evaluate the performance of ML enhancement, we code a number of QCIF sequences as shown in Table I. For all sequences except Mobile, 50 atoms/frame/description are coded in the central loop, with the first 15 atoms shared between the two descriptions and remaining 70 atoms alternatively split between the two descriptions. Additional 30 atoms/frame/description are coded in the side loops. For Mobile, the numbers are 290, 30, 520, and 160. The numbers are chosen to yield desired bitrates. All sequences, except Larry, are 10 seconds in duration. The first frame of each sequence is intra-coded with quantization step size equal to 8. For all sequences except Larry, all other frames are inter-coded unless a scene change is detected, in which case the frame is intra-coded. For the Larry sequence, we have 1689 frames at 15 frames per second (fps). To avoid excessive error propagation, an I frame is encoded every 100 frames for Larry, in addition to I frames due to scene change. Total bit rates of two descriptions and average PSNR without ML enhancement of the sequences are listed in Table I. PSNR0 is the average PSNR of reconstruction from both descriptions and PSNR1 is the average PSNR of reconstruction from one description. For comparison, average PSNR of a single description MP video codec at the same bit rate is listed in the SDC column, and the difference in PSNR0 between SDC and MDC without ML estimation is listed in the $\Delta_e$ column. MDC has a lower PSNR0 than SDC due to the redundancy.

Figure 2 shows the performance gain by using ML enhancement without feedback for a large number of video sequences at different frame rates. By this we mean the reconstruction from the central loop, not the enhanced reconstruction, is used in the encoder central prediction loop and the enhanced reconstruction is used in the decoder. The baseline scheme for comparison has no ML enhancement and uses the reconstruction in the central loop when both descriptions are received. The average gain in PSNR0 from ML enhancement without feedback is listed in $\Delta_a$ column in Table I. It can be seen that the gain achieved by ML enhancement varies from sequence to sequence. ML enhancement works better for sequences with low motion, such as Hall and Akiyo, than high motion sequences.

Additional gain can also be achieved by feeding back the enhanced reconstruction to the central loop. More specifically, the enhanced reconstruction of the previous frame is used for motion prediction and compensation of the current frame in the central loop. Table I lists the average gain in PSNR0 from the feedback in column $\Delta_b$, as well as the gain in PSNR1 in column $\Delta_c$, over MDC without ML estimation. As seen, there are gains in PSNR1, the PSNR of reconstruction from one description. This is because the use of enhanced reconstruction from two descriptions in the central prediction loop creates better motion vectors and two sets of atoms, F1 and F2, that reduce the residue energies in the side loops. The variation in PSNR1 gain from feedback among different sequences, i.e. $\Delta_c$, seems to be correlated with the variation in PSNR0 gain from ML enhancement, i.e. $\Delta_a$. There is little additional gain in PSNR0 due to feedback, i.e. the difference between $\Delta_b$ and $\Delta_a$ is small, except for a few low motion sequences. The combined average gains in both PSNR0 and PSNR1 are quite significant for low motion sequences. Column $\Delta_d$ in Table I lists the difference in PSNR0 between SDC and MDC using ML estimation with feedback. The difference is smaller for low motion sequences due to better gain from ML enhancement and less redundancy in MDC from motion vector duplications. Comparing $\Delta_d$ and $\Delta_e$ columns, we see that the gap between PSNR0 of MDC and SDC has been reduced after ML estimation.

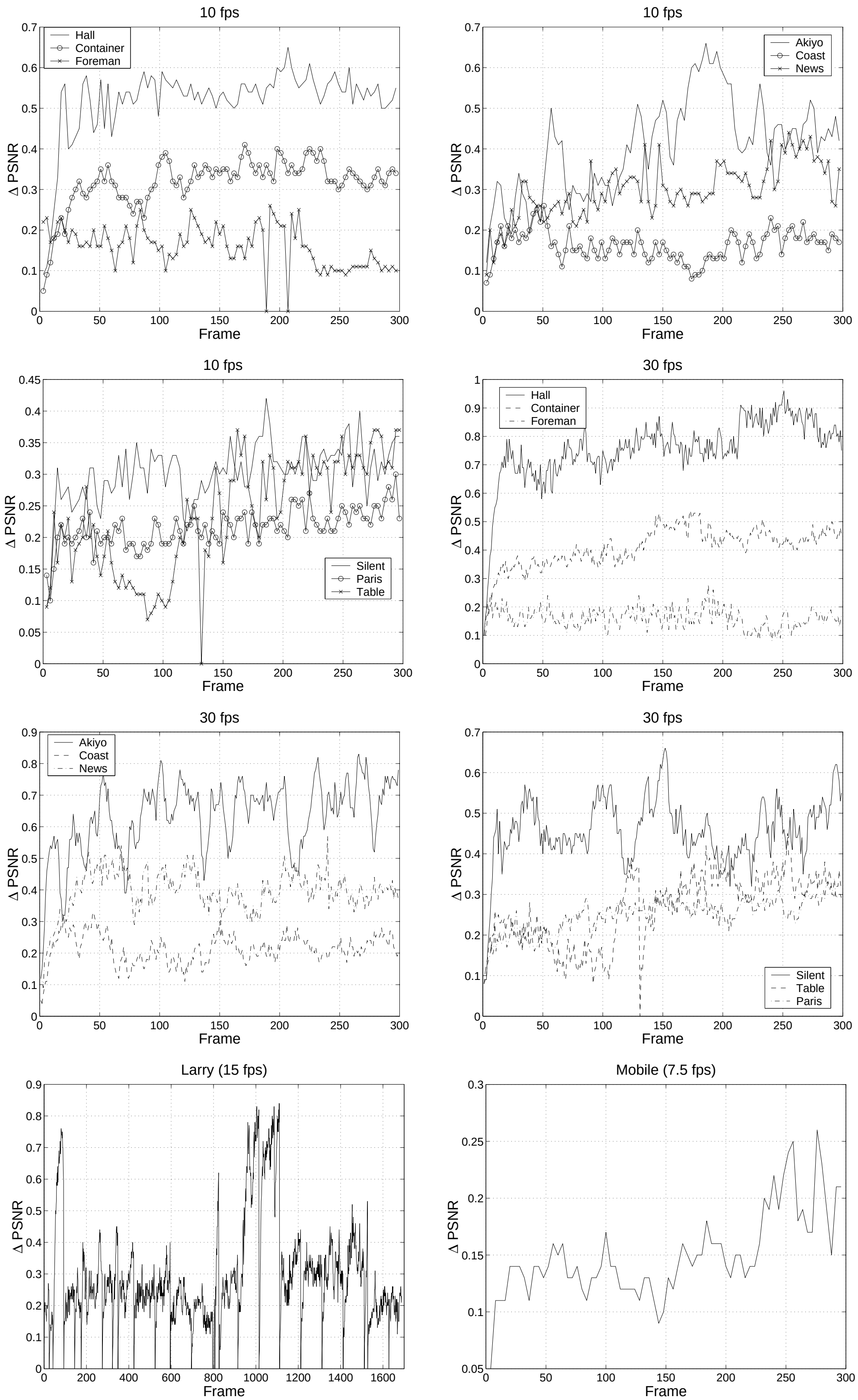



Fig. 2. Gain in PSNR0 from ML estimation.

| Sequence | bitrate (kbps) | MDC w/o ML est. PSNR0 | MDC w/o ML est. PSNR1 | SDC PSNR0 | SDC $\Delta_e$ | ML w/o fb. PSNR0 | ML w/o fb. $\Delta_a$ | ML estimation with feedback PSNR0 | ML estimation with feedback $\Delta_b$ | ML estimation with feedback PSNR1 | ML estimation with feedback $\Delta_c$ | $\Delta_d$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 fps | | | | | | | | | | | | |
| Hall | 45.2 | 35.70 | 34.66 | 37.01 | 1.31 | 36.22 | 0.52 | 36.36 | 0.66 | 35.13 | 0.47 | 0.65 |
| Container | 42.4 | 33.91 | 33.01 | 35.78 | 1.87 | 34.23 | 0.32 | 34.18 | 0.27 | 33.35 | 0.34 | 1.60 |
| Foreman | 85.6 | 30.92 | 30.28 | 34.32 | 3.40 | 31.08 | 0.16 | 31.07 | 0.15 | 30.47 | 0.19 | 3.25 |
| Akiyo | 42.6 | 38.27 | 37.01 | 40.52 | 2.25 | 38.67 | 0.40 | 38.92 | 0.65 | 37.58 | 0.57 | 1.60 |
| Coast | 64.6 | 28.50 | 27.76 | 30.89 | 2.39 | 28.66 | 0.16 | 28.67 | 0.17 | 28.03 | 0.27 | 2.22 |
| News | 54.2 | 33.02 | 32.15 | 36.11 | 3.09 | 33.32 | 0.30 | 33.38 | 0.36 | 32.53 | 0.38 | 2.73 |
| Paris | 60.4 | 27.92 | 27.17 | 30.85 | 3.93 | 28.13 | 0.21 | 28.18 | 0.26 | 27.45 | 0.28 | 2.67 |
| Silent | 60.0 | 33.43 | 32.59 | 36.20 | 2.77 | 33.73 | 0.30 | 33.75 | 0.32 | 32.89 | 0.30 | 2.45 |
| Table | 66.8 | 32.06 | 31.46 | 34.98 | 2.92 | 32.29 | 0.23 | 32.33 | 0.27 | 31.69 | 0.23 | 2.65 |
| 30 fps | | | | | | | | | | | | |
| Hall | 123.0 | 36.73 | 35.24 | 38.29 | 1.56 | 37.48 | 0.75 | 38.03 | 1.30 | 36.13 | 0.89 | 0.26 |
| Container | 115.6 | 35.03 | 33.38 | 36.76 | 1.73 | 35.44 | 0.41 | 35.46 | 0.43 | 33.94 | 0.56 | 1.30 |
| Foreman | 181.4 | 32.21 | 31.27 | 35.11 | 2.90 | 32.37 | 0.16 | 32.43 | 0.22 | 31.69 | 0.42 | 2.68 |
| Akiyo | 114.4 | 39.68 | 37.44 | 42.64 | 2.96 | 40.31 | 0.63 | 40.89 | 1.21 | 38.61 | 1.17 | 1.75 |
| Coast | 148.0 | 29.76 | 28.72 | 31.99 | 2.23 | 29.97 | 0.21 | 29.98 | 0.22 | 29.07 | 0.35 | 2.01 |
| News | 131.6 | 35.06 | 33.65 | 38.03 | 2.97 | 35.45 | 0.39 | 35.61 | 0.55 | 34.29 | 0.64 | 2.42 |
| Paris | 137.8 | 29.49 | 28.33 | 32.69 | 3.20 | 29.74 | 0.25 | 29.70 | 0.21 | 28.72 | 0.39 | 2.99 |
| Silent | 144.8 | 35.22 | 33.80 | 38.05 | 2.83 | 35.68 | 0.46 | 35.86 | 0.64 | 34.46 | 0.66 | 2.19 |
| Table | 155.6 | 33.27 | 32.35 | 36.05 | 2.78 | 33.53 | 0.26 | 33.56 | 0.29 | 32.72 | 0.37 | 2.49 |
| 15 fps | | | | | | | | | | | | |
| Larry | 74.4 | 34.19 | 33.24 | 36.18 | 1.99 | 34.48 | 0.29 | 34.63 | 0.44 | 33.65 | 0.41 | 1.55 |
| 7.5 fps | | | | | | | | | | | | |
| Mobile | 180.0 | 27.35 | 26.11 | 29.97 | 2.62 | 27.49 | 0.14 | 27.39 | 0.04 | 26.40 | 0.29 | 2.58 |

TABLE I

MDC/SDC bit rate and average PSNR. $\Delta_a$: the difference between MDC PSNR0 using ML estimation without feedback (ML w/o fb.) and MDC PSNR0 without ML estimation. $\Delta_b$: the difference between MDC PSNR0 using ML estimation with feedback and MDC PSNR0 without ML estimation. $\Delta_c$: the difference between MDC PSNR1 using ML estimation with feedback and MDC PSNR1 without ML estimation. $\Delta_d$: the difference between SDC PSNR0 and MDC PSNR0 using ML estimation with feedback. $\Delta_e$: the difference between SDC PSNR0 and MDC PSNR0 without ML estimation.

## IV. Rate Distortion Performance

In this section, we characterize the rate distortion performance of our MDC scheme. There are three parameters related to rate distortion in the two description case, namely rate, average two description PSNR (PSNR0), and average one description PSNR (PSNR1).

In the first test, we keep the bit rate of both descriptions constant and study the tradeoff between PSNR0 and PSNR1 by varying the number of atoms per description per frame in the central loop (C) and side loops (S), as well as number of shared of atoms in the central loop (L). To have an approximately constant bit rate for each description, we have fixed total number of atoms per frame per description, $C + S = 80$. Other test parameters are the same as these in Section III-B. To obtain the PSNR0 and PSNR1 tradeoff curve, we examine a number of combinations of L and S, and take the concave hull of the measured data points (PSNR0, PSNR1). More specifically, L and S take values from sets $\{1\ 5\ 15\ 25\}$ and $\{5\ 10\ 15\ 20\ 30\ 40\}$, respectively. We find that changing S is a more effective way of allocating redundancy than changing L since most data points on the concave hull correspond to small values of L, i.e. $L = 1$ and 5. The tradeoff with and without ML enhancement is shown in Figure 3. As can be seen, ML enhancement improves the performance compared to the baseline case without ML enhancement. Gains are smaller at large PSNR0 since fewer bits are spent in the side loops.

In the second test, we compare the Redundancy Rate Distortion (RRD) performance of our MP based MDC (MP-MDC) scheme with the H.263 based MD coder

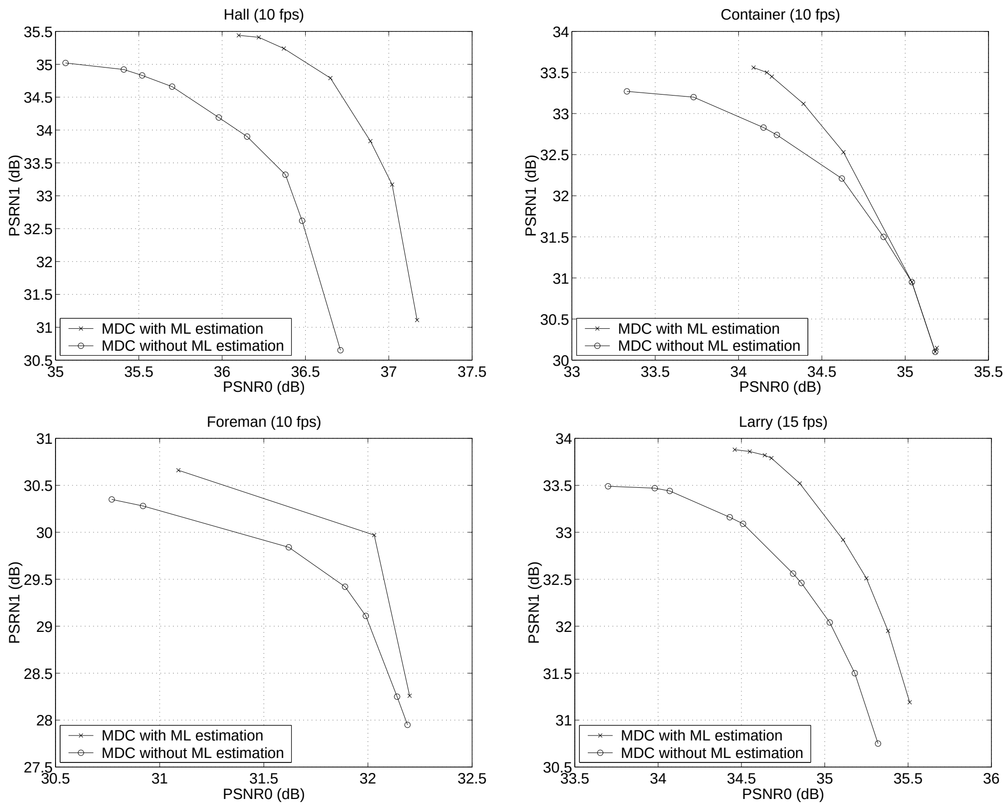



Fig. 3. Tradeoff between PSNR0 and PSNR1 with and without ML enhancement.

(MDTC) in [1]. Redundancy rate is defined as the additional bit rate of a MDC scheme achieving the same two description distortion (PSNR0) of a SDC scheme [13]. We have 2 reference SDC schemes, an H.263 codec for MDTC and a MP codec for MP-MDC. I-frame rate of 1 of every 15 frames is used. Quantization stepsize in I frames is fixed at 12. Group-of-blocks (GOB) headers are used in this test. Table II lists the QCIF sequences used for the test, frame rates, PSNR0 of 2 reference SDC schemes at same bit rate (notice that PSNR0 for MP is slightly higher due to the efficiency of MP coding). PSNR0 of each MDC scheme is fixed at PSNR0 of its corresponding reference SDC scheme in the test. Figure 4 compares the RRD performance of the two coders by plotting PSNR1 versus bit rate. Redundancy rate is expressed as a percentage of the reference rate. Due to frequent intra coding and duplication of I frames in MP-MDC, the redundancy starts at a larger value than MDTC, which uses MD transform coding for I frames as well. We can see that MP-MDC outperforms MDTC in PSNR1 quite significantly.

## V. Performance Over Lossy Channels

In this section, we study the performance of our video MDC scheme over lossy channels through simulation, and compare it with single description coding (SDC) schemes. There are two independent channels with each description generated by MDC sent over one channel. For SDC, the bitstream is split and sent through two channels. Intuitively, we would expect MDC to outperform SDC if one channel introduces large bit error rate (BER) beyond correction capability of forward error correction (FEC) schemes employed, i.e. one channel is off, while the other channel has low BER. Under these conditions, MDC for video can reconstruct the source from only one description without losing synchronization with the encoder while SDC is likely to lose synchronization and cause error propagation. The more slowly varying the two channels are, the longer the error propagation for the SDC case, and hence the more advantageous MDC becomes over SDC. Since in video communication variation of video quality over time generally needs to be minimized, MDC could potentially offer advantages over SDC in bursty slowly varying channels. In what follows, we will demonstrate this more quantitatively through simulations with two state Markov and Rayleigh fading channels.

We choose two MP based SDC schemes for comparison.

**SDC1** : We code at the same source rate as two descriptions in MDC and use the same FEC for atoms so that total rate is the same.

**SDC2** : We code at such source rate that PSNR is same as two description MDC with no channel error, and choose FEC to match the overall rate with MDC.

Comparisons in the literature are typically between MDC and SDC1 over lossy channels, showing that MDC outperforms SDC1 at large loss rates and vice versa at small loss rates due to redundancy in MDC. We believe a more interesting comparison would be between MDC and SDC2, both having same error free performance. SDC2 allocates all available redundancy to FEC while MDC uses weaker FEC and allocates remaining redundancy to source coding. Comparison between MDC and SDC2 would test the efficiency of a MDC design.

For fair comparison, we assume the FEC protection methods for I frames, motion vectors and headers are the same in both MDC and SDC schemes. Due to the importance of I frames, we send two copies of I frames[4], one to each channel with BCH(127,64,10) protection for both MDC and SDC schemes. If there are remaining errors in an I frame, reconstructed GOB(s)[5] that are affected by the errors are replaced by the corresponding GOB(s) in the previous frame. Motion vectors are also duplicated, protected by BCH(127,64,10) code and sent over two channels. If there are errors in motion vectors, default value of zero is assumed. The parameters of BCH code are chosen to achieve desired range of performance across channel conditions. We have not considered the use of adaptive error correction codes because we are primarily interested in low delay applications in which channel condition is not fed back to the sender, and hence adaptivity is not an option.

Atoms in SDC are equally divided among two channels per frame by allocating atoms in the even GOBs to one channel and atoms in the odd GOBs to the other channel. The FEC used for atoms in MDC and SDC1 is BCH(127,78,7). With source rate fixed, we test the efficiency of MDC with large redundancy (PSNR0 = 34.63 dB, PSNR1 = 33.65 dB) and MDC with small redundancy (PSNR0 = 35.33 dB, PSNR1 = 32.11 dB). For atoms in SDC2, the corresponding FECs used are BCH(127,50,13) and BCH(127,71,9) for large and small redundancies, respectively. The overall bit rate after FEC is 140 kbps for all schemes. When atoms are received incorrectly, all remaining atoms in the same GOB are lost due to loss of synchronization in entropy coding. For MDC, we apply a simple error concealment method: When one channel introduces lots of errors, a large number of atoms received from the channel are deleted, creating a large mismatch between encoder and decoder. The mismatch can be reduced, if the other channel has small errors and atoms in the description received through the good channel are copied to the other description because of the redundancy in the two sets of atoms of two descriptions. The decision when to apply this method is heuristic; in the following simulations, it is applied when one channel receives 30 or more atoms than the other.

### *A. Two State Markov Channels*

In this model, a channel can be in either "good" state with bit error rate BER $P_g$, or "bad" state with BER $P_b$. The state transition probabilities are $p_1$, from "good" state to "bad" state", and $p_2$, from "bad" state to "good" state". In the simulations, we assume channels can change state only at the start of a frame transmission to achieve relatively slowly varying channels[6].

Figure 5 plots the average PSNR of MDC and two SDC schemes versus BER in the bad state $P_b$ for the Larry sequence[7] at 15 fps for the case with large and small redundancy. For both cases in the plots, SDC1 outperforms MDC and SDC2 at low BER because all errors can be corrected by the FEC schemes used and SDC1 has the least redundancy. As error rate increases from approximately 3% to 9%, MDC outperforms SDC1 because of redundancy in MDC, but SDC2 is best because it has the strongest FEC

[4]Macro block based intra refresh is interesting to test. However, we are not able to test it due to time constraint.

[5]GOB headers are used to regain synchronization after an error in the bitstream. For QCIF, GOB size is 11.

[6]A very fast varying two state Markov channel(e.g. possible state change every few bits) behaves similar to a memoryless uniform random bit error channel, in which case SDC2 always outperforms MDC because of better error protection.

[7]Due to time constraints, we limited our experiments in the section to Larry sequence only.

protection. As BER increases further, MDC outperforms SDC2 since most errors in the bad state can no longer be corrected. This verifies our intuition that MDC outperforms SDC in slowly varying bursty channels.

### B. Rayleigh Fading Channels

We measure the performance of MDC and SDC schemes over Rayleigh fading channels simulated using Jakes' method [14]. Two channels are assumed to correspond to two carrier frequencies at 900 MHz and 1.8 GHz, respectively, the motivation being cell phones which operate at both frequencies depending on whether they are being used in USA or Europe. A 200 kHz channel bandwidth and binary phase shift keying (BPSK) modulation are assumed. We run tests under both slowly varying channels and fast varying channels. The Doppler frequency $f_D$ equals $vf_c/c$, where $v$ is the mobile speed, $f_c$ is the carrier frequency, and $c$ is the light speed. The product $f_D T$, where $T$ is the bit interval, corresponds to the varying speed of a Rayleigh channel. Since the mobile moves at a given speed, and one carrier frequency is twice the other one, this results in $f_{D1}T$ for channel 1 to be twice as large as $f_{D2}T$ for channel 2. A block interleaver is used to study the effect of interleaving delay.

First, we examine the performance with large redundancy in MDC in Figures 6 to 8. Figure 6 shows the performance of SDC and MDC schemes in Rayleigh fading channels with different channel variation speed, i.e. $f_D T$. The interleaving delay is fixed at 100 ms. The following observations can be made about Figures 6(a) and 6(b):

*1.* In Figure 6(a), corresponding to slowly varying channels, MDC performs best for channel PSNR below 18 dB and SDC1 performs best beyond 18 dB.

*2.* In Figure 6(b), corresponding to fast varying channels, SDC2 performs best for channel PSNR below 15 dB and SDC1 performs best for above 15 dB.

*3.* Combining the above two observations, we see that using MDC is more advantageous in slowly varying channels than faster varying ones. Other simulations not shown here show that this is particularly true when interleaving is large, e.g. 100 ms as in Figures 6(a) and 6(b). The reason is that interleaving makes bit error occurrence more uniform, thereby making FEC more effective, and hence SDC more attractive than MDC.

*4.* Comparing Figures 6(a) and 6(b), we find that at low values of channel SNR, both SDC and MDC techniques do worse in faster varying channels than in slowly varying channels; this is because under these circumstances, in fast varying channels FEC cannot protect the bitstream with large uniform error corruption, whereas in slowy varying channels, at least some information can be recovered by FEC since errors occur in bursts.

Figures 7 and 8 show the effect of interleaving on the performance of SDC and MDC schemes in slowly and fast varying Rayleigh channels. The following observations can be made about Figures 7 and 8:

*1.* In Figure 7(a) corresponding to no interleaving, MDC does best for channel SNR values below 18 dB and SDC1 for above 18 dB. Thus, MDC is quite effective in slowly varying channels and low delay applications where interleaving is not an option.

*2.* In Figure 7(b) corresponding to interleaving of 150 ms, MDC still does best at small SNR, however the gap between MDC and SDC has decreased as compared to Figure 7(a). This is to be expected since large interleaving and FEC make a powerful combination in combatting noise.

*3.* As the redundancy level in MDC decreases, for example, by traversing along the curves in Figure 3, the advantage of MDC over SDC becomes smaller for all interleaving delays and channel speeds. The corresponding plots for this case are not shown here due to space limitations.

*4.* Figure 8 correponds to fast varying channels. In Figure 8(a) corresponding to no interleaving, SDC and MDC show more or less identical performance; In Figure 8(b) corresponding to large interleaving SDC does better than MDC.

Combining the above three observations and comparing Figures 7 and 8, we conclude that MDC is most useful in situations where (a) channel is varying slowly and (b) low delay requirments of the application preclude use of interleaving; As the variation rate of channel increases and delay requirements are relaxed, use of FEC together with SDC becomes more attractive.

## VI. Conclusion

In this paper, we study a video MDC scheme based on a 3 loop structure and MP coding. We examine performance gain through ML enhancement from both descriptions and find that it works best for low motion sequences. Gain up to 1.3 dB can be achieved in our tests. Performance comparison is made between MDC and SDC schemes over lossy channels, including two state Markov channels and Rayleigh fading channels. We find that MDC with large redundancy outperforms SDC in bursty slowly varying environments. For Rayleigh fading channels, bit interleaving helps SDC close the gap and even outperform MDC depending on the amount of interleaving and speed of channel variation. It would be interesting to compare the performance of MP-MDC with MDTC and study the effect of macro block based intra refresh in lossy channels.

## VII. Acknowledgement

The authors would like to thank Amy Reibman for providing MDTC RRD data and anonymous reviewers for their comments which greatly enhanced the clarity of the paper.

## References

[1] A. Reibman, Y. Wang, M. Orchard, and R. Puri, “Multiple description coding for video using motion compensated prediction,” in *Proc. ICIP 99*, October 1999, pp. 837–841.

[2] L. Ozarow, “On a source coding problem with two channels and three receivers,” *Bell System Technical Journal*, vol. 59, pp. 1909–1921, December 1980.

[3] A. EL Gamal and T. Cover, “Achievable rates for multiple descriptions,” *IEEE Trans. on Information Theory*, vol. IT-28, no. 6, pp. 851–857, November 1982.

[4] V. Vaishampayan, “Design of multiple description scalar quantizers,” *IEEE Trans. on Information Theory*, vol. 39, no. 3, pp. 821–834, May 1993.

[5] Y. Wang, M. Orchard, and A. Reibman, “Optimal pairwise correlating transforms for multiple description coding,” in *Proc. ICIP 98*, 1998, pp. 679–683.

[6] V. Vaishampayan and S. John, “Interframe balanced multiple description video compression,” in *Proc. Packet Video 1999*, April 1999.

[7] N. Gogate and S. Panwar, “Supporting video/image applications in a mobile multihop radio environment using route diversity,” in *Proc. ICC 99*, 1999, pp. 1701–1706.

[8] A. Reibman, Y. Wang, X. Qiu, Z. Jiang, and K. Chawla, “Transmission of multiple description and layered video over an EGPRS wireless network,” in *Proc. ICIP 2000*, 2000.

[9] R. Neff and A. Zakhor, “Very low bit-rate video coding based on matching pursuits,” *IEEE Trans. CSVT*, vol. 7, no. 1, pp. 158–171, February 1997.

[10] O.K. Al-Shaykh, E. Miloslavsky, T. Nomura, R. Neff, and A. Zakhor, “Video compression using matching pursuits,” *IEEE Trans. CSVT*, vol. 9, no. 1, pp. 123–143, February 1999.

[11] S. Cheung, R. Neff, and A. Zakhor, “Changes regarding matching pursuits in video vm v.11 (mpeg 98/m3832),” in *ISO/IEC JTC1/SC29/WG11*, July 1998.

[12] A. Katsaggelos, “A multiple input image restoration approach,” *Journal of Visual Communication and Image Representation*, vol. 1, no. 1, pp. 93–103, September 1990.

[13] M.T. Orchard, Y. Wang, V. Vaishampayan, and A. Reibman, “Redundancy rate-distortion analysis of multiple description coding using pairwise correlating transforms,” in *Proc. ICIP 1997*, 1997.

[14] G. Stuber, *Principles of Mobile Communication*, Kluwer Academic Publishers, 1996.

| Sequence | Frame rate (fps) | PSNR0 (dB) H.263, MDTC | PSNR0 (dB) MP, MP-MDC | Reference bit rate (kbps) |
|---|---|---|---|---|
| Container | 10 fps | 31.62 | 32.66 | 31.9 |
| Foreman | 7.5 fps | 31.38 | 31.38 | 49.4 |
| Hall | 10 fps | 32.40 | 33.16 | 31.7 |
| Silent | 15 fps | 31.49 | 31.89 | 50.4 |

TABLE II

Test sequences and parameters for RRD performance.

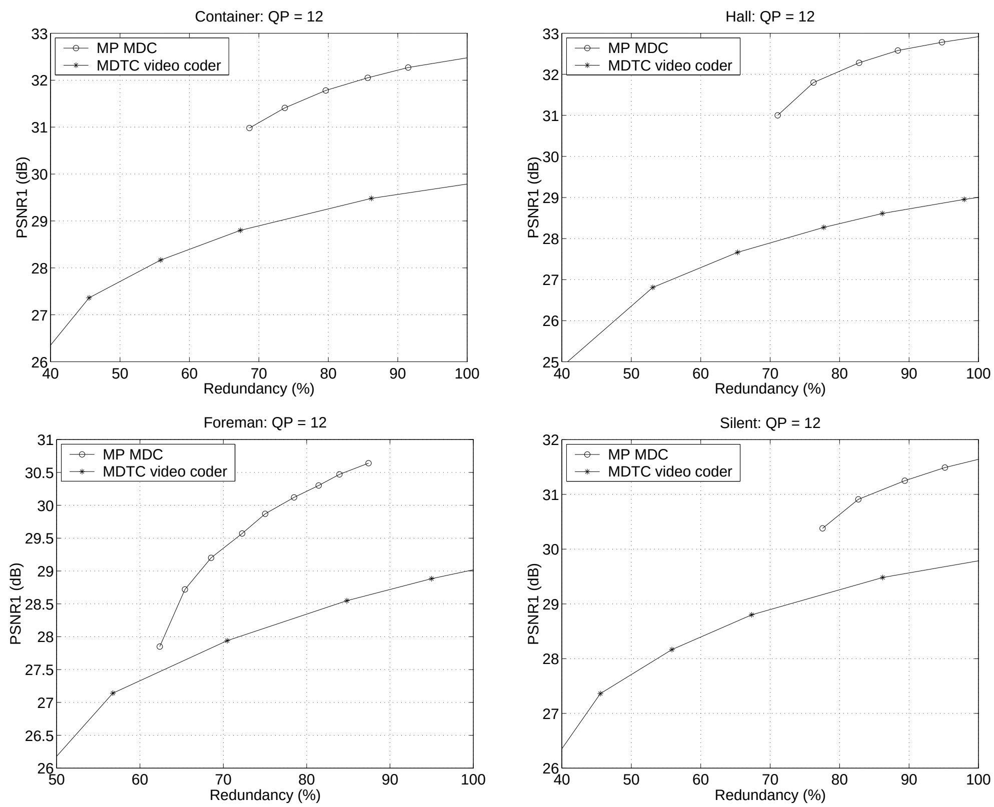
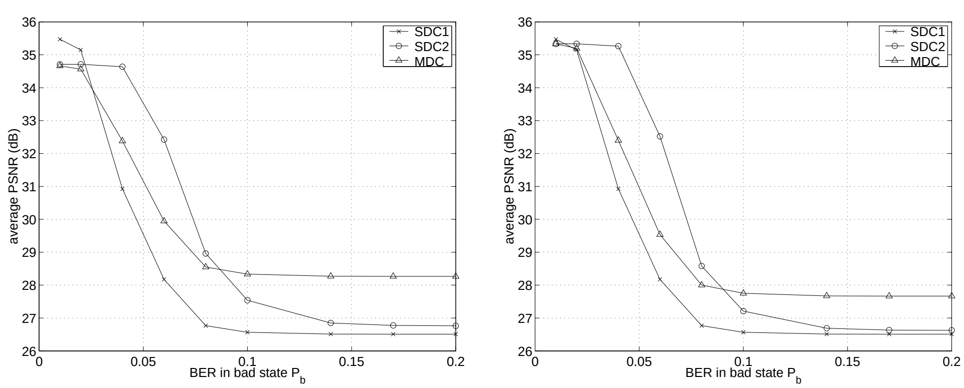
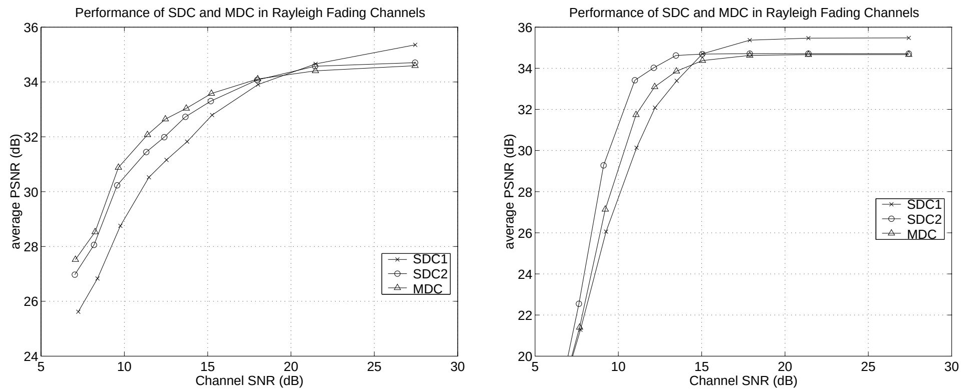



Fig. 4. Redundancy rate distortion performance of MDTC coder and MP MDC.



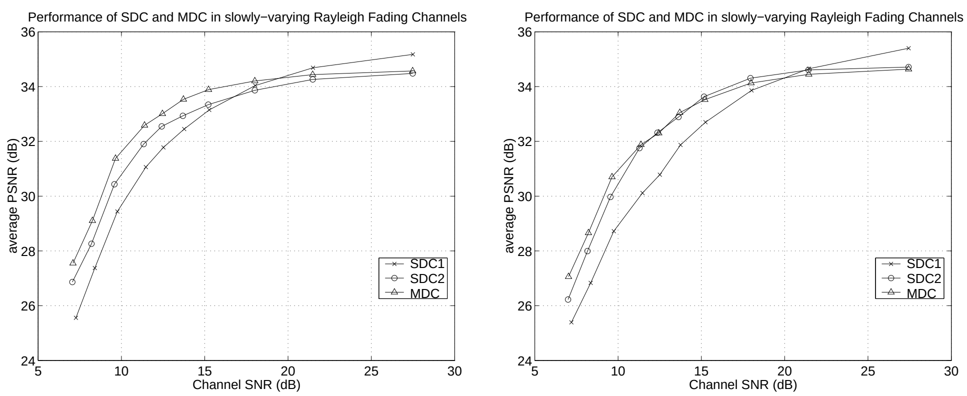

(a) (b)

Fig. 5. Performance of SDC and MDC schemes in slowly varying two-state Markov channels with different redundancy in MDC. $P_g = 0.001$, $p_1 = 0.8$, $p_2 = 0.8$. (a) large redundancy (b) small redundancy



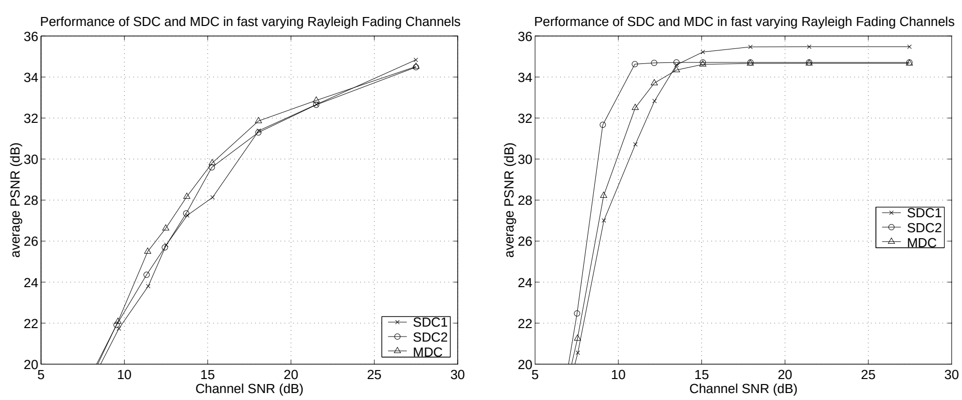

(a) (b)

Fig. 6. Performance of SDC and MDC large redundancy schemes in Rayleigh fading channels with different varying speed and at interleaving delay of 100 ms. Values of $f_dT$ for two Rayleigh channels are (a) $5 \times 10^{-6}$, $10^{-5}$ (b) $1.5 \times 10^{-4}$, $3 \times 10^{-4}$ The ratio is 2 because of two carrier frequencies.




(a) (b)

Fig. 7. Performance of SDC and MDC large redundancy schemes in slowly varying Rayleigh fading channels with different interleaving delays. Values of $f_dT$ for two Rayleigh channels are $5 \times 10^{-6}$ and $10^{-5}$. Interleaving delays are (a) 0 ms (b) 150 ms




(a) (b)

Fig. 8. Performance of SDC and MDC large redundancy schemes in fast varying Rayleigh fading channels with different interleaving delays. Values of $f_dT$ for two Rayleigh channels are $3 \times 10^{-4}$ and $6 \times 10^{-4}$. Interleaving delays are (a) 0 ms (b) 150 ms