

# Rate-distortion optimized mode selection method for multiple description video coding

Yu-Chen Sun · Wen-Jiin Tsai

Published online: 19 April 2013

© Springer Science+Business Media New York 2013

**Abstract** Multiple description coding (MDC) is a potential solution for video transmission over error-prone networks because it shows promising enhancement of error resilient capability. The MDC systems encode a single video stream into two or more equally important independent sub-streams, called descriptions. Therefore, if some of the descriptors get lost, remaining descriptors can be used to recover the video. Much research has proposed different information distribution methods. Since each method has different characteristic, we proposed a general rate-distortion optimization framework for MDC systems in this paper. By sophisticated rate-distortion analysis and optimization, the framework enables MDC systems to adaptively encode video considering contents and channel variation. Experimental results showed that, by comparing with the work in Tsai and You (IEEE Trans Circ Syst Video Technol 22(2):309–320, 2012), the proposed technique improves the R-D performance significantly. The improvement can be up to 2.4 dB for the channels with 0%~20% packet loss rates, and it can be even more if the loss rate increases. The proposed framework is not restricted to specific MDC tools. Ones can easily integrate their proposed coding tools into the framework and achieve better performance as long as the macroblock's bitrate and distortion information can be measured.

**Keywords** Multiple description video coding · Rate-distortion optimization · Unequal error protection · Multimedia transmission

## 1 Introduction

Transmission of video signals over wireless channels or over IP-based networks is a challenging problem, because, during data transmission, packets may be dropped or damaged, due to channel errors, congestion, and buffer limitation. For real-time applications, since retransmission is often not acceptable, error resilience (ER) and error concealment (EC) techniques are required for displaying a pleasant video signal despite the errors and for reducing distortion introduced by error propagation.

---

Y.-C. Sun · W.-J. Tsai (✉)

Department of Computer Science and Information Engineering, National Chiao-Tung University,  
Hsinchu, Taiwan, Republic of China  
e-mail: wjtsai@cs.nctu.edu.tw

Multiple description coding (MDC) [22] has received much attention for the past decades because MDC is a promising approach to enhance error resilient capability. This property makes MDC a potential solution for emerging applications that require video transmission over error-prone networks [13].

Multiple description coding is a technique that encodes a single video stream into two or more equally important sub-streams, called descriptions, each of which can be decoded independently. Different from the traditional single description coding (SDC) where the entire video stream (single description) is sent in one channel, in MDC, these multiple descriptions are sent to the destination through different channels, resulting in much less probability of losing the entire video stream (all the descriptions), where the packet losses of all the channels are assumed to be independently and identically distributed. The first MD video coder, called multiple description scalar quantizer (MDSQ) [21], has been realized in 1993 by Vaishampayan who proposed an index assignment table that maps a quantized coefficient into two indices each could be coded with fewer bits. Due to effectiveness in providing error resilience, a variety of research on different MDC approaches had been proposed afterwards. These approaches can be intuitively classified through the stage where it split the signal, such as, frequency domain [4, 21], spatial domain [2, 12], and temporal domain [1, 10]. In our previous works [11], a hybrid MDC method has been proposed, which applies MDC first in spatial domain to split motion compensated residual data, and then in frequency domain to split quantized coefficients. A hybrid MDC method with spatial and temporal splitting was proposed in [19] and a hierarchical B-picture based hybrid MDC method was proposed in [20]. The results in [9, 19, 20] show that, by properly utilizing more than one splitting technique, the hybrid MDC method can improve error-resilient performance.

To improve coding performance, some researchers proposed to optimize the encoding coefficient for rate-distortion performance. In [5, 6], a R-D optimization technique is proposed for the MDC with one descriptor containing all DCT coefficients and the second one containing only few low frequency coefficients. The R-D technique aims at optimizing the number of pruning coefficients, given the target description bitrate. In [18], the method to find out optimized quantization parameters was proposed for the MDC based on H.264/AVC redundant slices [24]. Then, Lin et. al [14] extended the method from the slice level to the macroblock level.

There are two major benefits of the rate-distortion optimization concept. First, video contents vary spatially and temporally, so it would be inefficient to use a fixed encoding method to encode whole contents. In addition, the importance of different parts of video contents may be different, so adopting an unequal error protection can achieve better rate-distortion performance. Second, the channel condition also varies over time, so a mechanism to dynamically adjust protection level is necessary. With rate-distortion optimization, the encoder can change coding strategy according to video contents and channel conditions, and therefore improve the performance. However, the previous optimization frameworks were based on the specific MDC systems. Since a variety of new MDC coding tools are being proposed and each tool has different characteristics. To enable the rate-distortion optimization concept on these MDC tools, a general framework is desirable. Therefore, this paper aims at proposing a general optimization framework. The proposed framework analyzes the bitrate and distortion information of macroblocks and optimizes the performance. As long as a MDC tool can provide bitrate and distortion measurements, it can utilize the proposed framework to further improve coding performance. This property makes the proposed

framework suitable for most macroblock based MDC tools and not restricted to specific coding structures, such as IPPP or hierarchical B-picture structure. Ones can easily integrate their coding tools into the proposed optimization framework and achieve better performance. In this paper, we applied the proposed optimization framework on the MDC system in [20] to explain the proposed framework. The major differences between the proposed method and the MDC system in [20] include:

- The MDC tool selection is adaptive in the proposed method, while it is fixed in [20].
- The protection level is determined at macroblock level in the proposed method, while it is at frame level in [20]
- Video content characteristics and channel conditions are taken into considerations in the proposed method, while they are not considered in [20].

The remainder of this paper is organized as follows. First, the proposed MDC method which is an improved version of the MDC system in [20] is presented in section 2. Section 3 introduces the proposed framework, and section 4 verifies it with simulation data. Section 5 concludes the paper by summarizing the main results, and discussing possible future work.

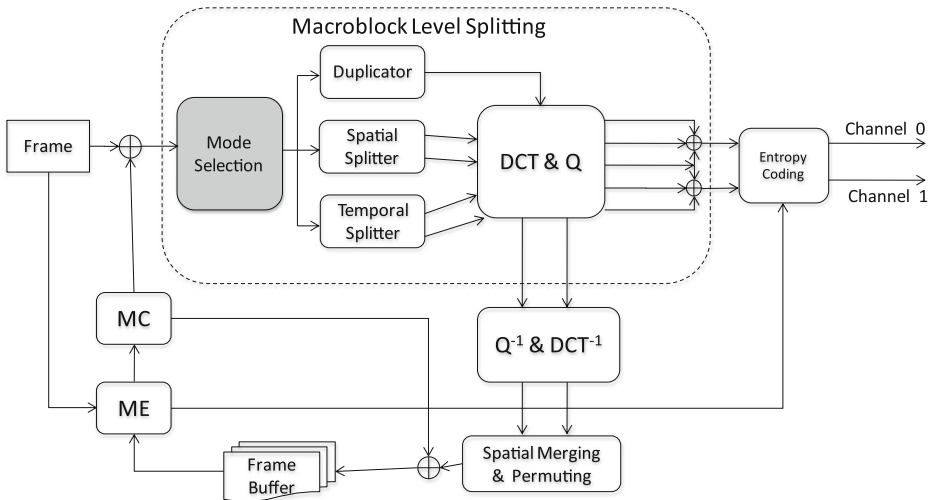
## 2 Proposed MDC based on a hierarchical B-picture structure

This paper proposes a general R-D optimization framework for MDC systems. To illustrate and evaluate the proposed framework, the MDC system in [20] is adopted, although our optimization approach is not restricted to this specific MDC method. The adopted MDC is a complex system with a wide choice of splitters on a hierarchical B-picture coding structure. With the illustration of applying our approach to this complex MDC system, one can easily apply it to relatively simple MDC systems. The details of the improved MDC system are described in the following, and the proposed R-D optimization framework is described in Section 3.

### 2.1 The encoder architecture

Figure 1 shows the encoder architecture of the proposed MDC system which is an improved version based upon the MDC method in [20]. The architecture contains three MDC coding tools: duplicator, spatial splitter, and temporal splitter. The three tools divide a SDC bitstream into two MDC descriptors with different amount of redundancy on each. This architecture is similar to the one in [20] except that a mode selection module is added. To encode a frame, the mode selection module analyzes the importance of a macroblock in the frame and the channel condition and then chooses a suitable splitter for the macroblock, thereby optimizing R-D performance. After determining the coding tool, each macroblock is split and encoded into two individual descriptors.

The system contains three MDC coding tools: duplicator, temporal splitter, and spatial splitter. The duplicator generates two descriptors by directly duplicating the SDC data into each descriptor. Because each descriptor contains complete SDC data, the decoder can perfectly reconstruct the image as long as any one descriptor is received.



**Fig. 1** The encoder architecture of the proposed MDC system. The major difference between it and the one in [20] is that it includes a Mode Selection module

The temporal splitter splits the SDC bitstream in temporal domain, which assigns input macroblocks, in turn, to the two output paths such that successive macroblocks will go to different descriptors. Namely, when a macroblock is assigned to one description, it will be encoded as a skipped MB with no information in the other description. As a consequence, if any one descriptor is lost, those temporally split macroblocks belonging to the lost descriptor will get lost completely and can only be estimated by the macroblocks in spatial or temporal neighborhoods.

Spatial splitter splits each input macroblock into two parts which are then separately transformed, quantized, and entropy encoded before going to their respective descriptors. The spatial splitter performs splitting on an  $8 \times 8$  block basis in residual domain. For each  $8 \times 8$  residual block, it is first polyphase permuted inside the block and then is split to two, as shown in Fig. 2. The permuting mechanism is that, for every  $2 \times 2$  pixels inside the  $8 \times 8$  residual block, the top-left pixel (labeled 0) is re-arranged to the top-left  $4 \times 4$  block, the top-right pixel (labeled 1) to the top-right  $4 \times 4$  block, the bottom-left pixel (labeled 2) to the bottom-left  $4 \times 4$  block, and the bottom-right pixel (labeled 3) to the bottom-right  $4 \times 4$  block, as illustrated in the middle of Fig. 2. After polyphase permutation, the  $8 \times 8$  block is split into two  $8 \times 8$  blocks, each carries two  $4 \times 4$  blocks chosen in diagonal and the remaining two  $4 \times 4$  blocks are given all-zero residuals (labeled as ‘x’ in Fig. 2). Note that there are four  $8 \times 8$  residual blocks in each macroblock, all of them are permuted and split in the same way. Since these split frames need to be merged to serve as reference frames, a Spatial Merger is applied after de-quantization ( $Q^{-1}$ ) and inverse transform ( $DCT^{-1}$ ) as shown in Fig. 1. The Spatial Merger first discards the all-zero  $4 \times 4$  blocks and then adopts Polyphase Inverse Permuting (the reversed process of Fig. 1) to reconstruct the original  $8 \times 8$  blocks.

The proposed improved MDC system is also based on a non-dyadic hierarchical B-picture coding structure with 4 levels as depicted in Fig. 3. For the same structure, the MDC in [20] applies duplicator on the I/P frames at the lowest hierarchical level for providing the highest error resilience, spatial-splitter (S) on the reference B frames at

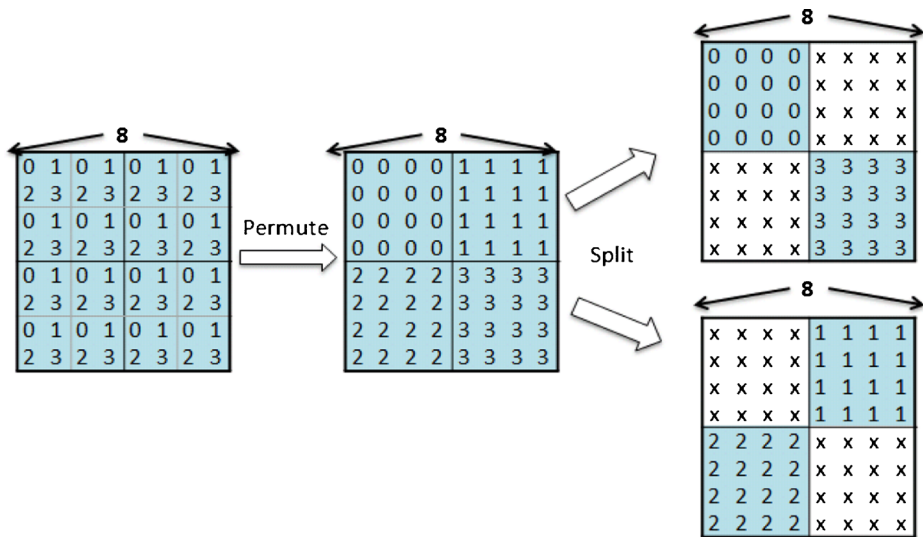


Fig. 2 Spatial splitting

intermediate levels for modest error resilience, and temporal-splitter (T) on the non-reference B frames at the highest level for the lowest error resilience. The idea behind the assignment in [20] is that the frames at the lower hierarchical level are more important and thus should be protected with more redundancy. In this paper, we extend the idea from frame level to macroblock level. In other word, we adaptively choose the splitters macroblock by macroblock according to its importance. A macroblock in the non-reference B frames at the highest level could be split by the temporal splitter or the duplicator; while a macroblock in other frames could be split by the spatial splitter or the duplicator. The proposed mode selection module plays a role to find out a splitter assignment that has better R-D performance. By splitter assignment at macroblock level, there may have three types of macroblocks distributed throughout the video sequence: duplicated macroblocks, spatially split macroblocks, and temporally split macroblocks. Figure 4 shows an example to illustrate the description generation. For temporally split macroblocks, they are encoded normally in one description and as skipped macroblocks in the other. For duplicated macroblocks, all the transformed coefficients belonging to one macroblock are encoded and duplicated in two descriptions. For spatially split macroblocks, only half of coefficients will be encoded in one description. Its counterpart will be in the other description. In these macroblocks, the punched coefficients will be set to zero. It

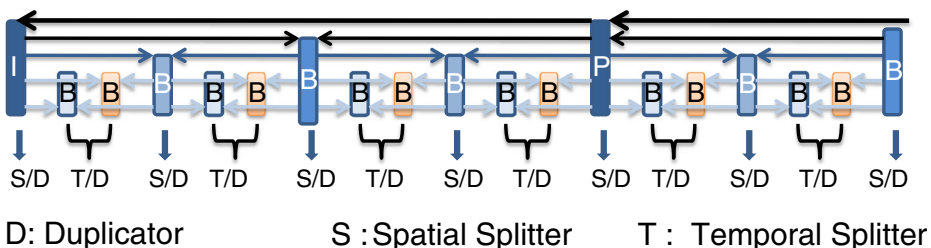
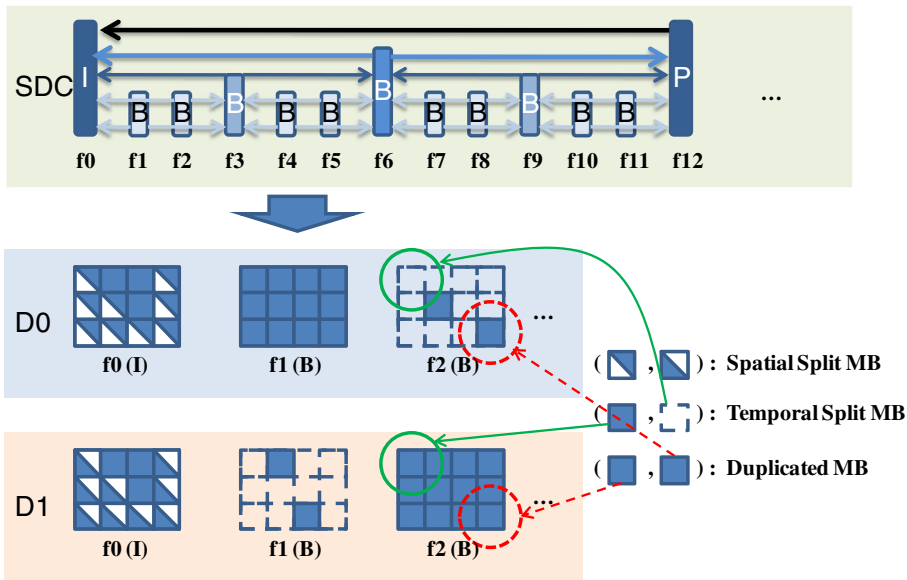


Fig. 3 Proposed MDC based on hierarchical B-picture prediction



**Fig. 4** An example of description generation. Each frame combines more than one types of MBs. For example, f2 combines temporal split MBs, indicted by green solid lines, and duplicated MBs, indicted by red dotted lines

is worth mentioning that since the decoder can decode macroblocks and identify their splitter types, no additional signaling bits are required in the bitstreams.

### 2.2 The decoder estimation methods

With the proposed MDC, assume the generated two descriptors are denoted by D0 and D1, respectively. Assuming one description, D0, is lost, the macroblocks split by duplicator can be easily reconstructed at decoder by using the same macroblocks in the other description, D1. For the macroblocks split by the spatial splitter, one descriptor loss will cause partially loss of the macroblocks, which can be estimated by using the information of their counterparts in D1. As Fig. 5 shown, the black blocks are lost pixels which will be estimated by bilinear interpolation from their counterparts.

As for the macroblocks split by the temporal splitter, one descriptor loss will cause loss of all the macroblocks in a frame, which can only be estimated by using other frames. In case of two-description loss, D0 and D1, it will result in whole-data loss regardless of splitter types. For whole-data loss, each macroblock is recovered based on temporal correlation. Figure 6 shows an example to illustrate temporal estimation. To recover macroblocks on the frame 8, the motion fields of frame 9, denoted as MF, is used to interpolate the motion fields of frame 8. With the interpolated motions, the lost pixels are recovered by the pixels in frames 6 and 9, denoted as DF. For other loss cases, the estimation method is similar, except that the choices of MF and DF may be different. The detail algorithm can be found in [20]. Since the estimation methods are not the focus of this paper, we simply adopt the estimation methods in [20] for our experiments in the later section.

Table 1 summarizes the cases for different estimation methods to be applied, where S denotes the spatial method, T the temporal method, and D the duplication method. The columns describe the two loss cases; while the rows describe three types of splitters. Note

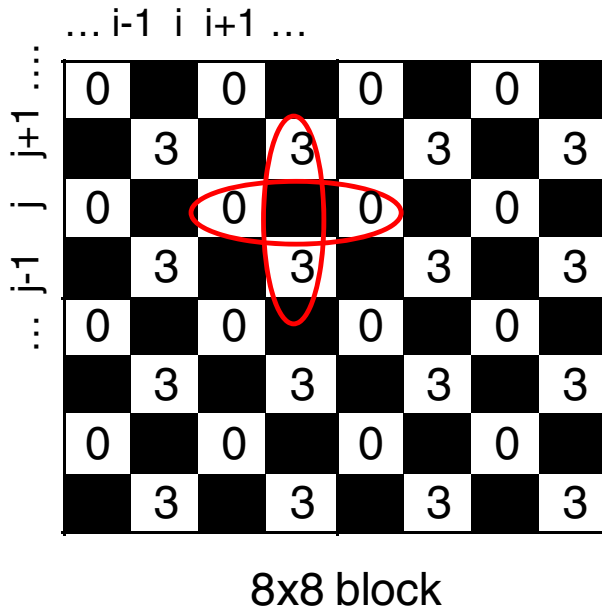


Fig. 5 An example of spatial estimation

that, when descriptions are lost, the frame is recovered by the depicted estimation methods and then it is stored into frame buffer for motion compensation of following frames. Therefore, the distortion due to estimation will be propagated to the following frames. This is a common issue in error recovering for video coding. In the next section, a method is proposed to analyze error propagation effect.

### 3 Rate-distortion mode selection method

A MDC system might contain lots of coding tools and have a complex coding structure. How to find out the mode assignment which has good R-D performance is a challenging problem. This paper proposes a R-D optimization framework. With the framework, encoder

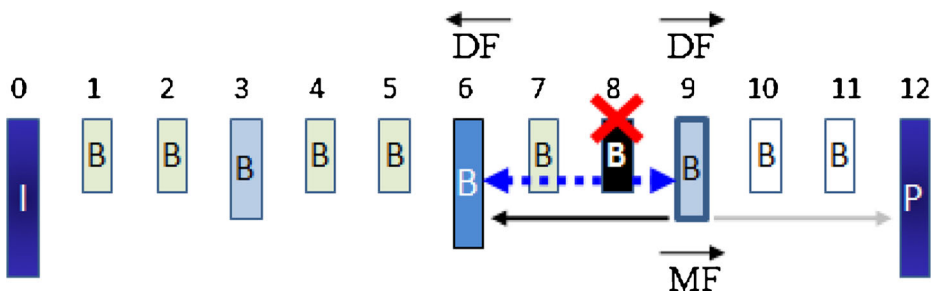


Fig. 6 An example of temporal estimation

**Table 1** Summary of the cases for different estimation methods

Estimation methods		Descriptor status	
		One-descriptor loss	Two-descriptor loss
MB Type	Duplicated MB	D	T
	Spatial split MB	S	T
	Temporal split MB	T	T

can decide a suitable splitter mode for each macroblock, thereby optimize the R-D performance. In following, we first explain the proposed framework on an ideal MDC channel. Then, the framework is extended to a packet loss channel. Finally, we summarize the proposed framework.

### 3.1 Rate-distortion optimization on an ideal MDC channel

An ideal MDC channel assumes that some descriptors are received without losing any information while the others are totally lost. Such a situation is referred to as side reconstruction. In the MDC system with two descriptors, e.g. the system introduced in section 2, there are two cases of side reconstruction.

Assume a video is encoded by traditional close-loop codec, and the resulting coding rate and distortion are  $R_{SDC}$  and  $D_{SDC}$ , respectively. A MDC system tries to divide the SDC data into two MDC descriptors. First, consider a naive design as a baseline design: the system that directly duplicates the whole SDC data into two descriptors, which is denoted as duplicator-only-MDC (DO-MDC). In this system, the bit-rate of each descriptor, say  $R_1$  and  $R_2$ , is equal to  $R_{SDC}$ . And, the distortion of side decoders are equal to  $D_{SDC}$ .

For DO-MDC which has two cases of side reconstruction, the average distortion of the two side decoders and the total bit-rate of the two descriptors are calculated as

$$\begin{aligned} D_{Side,DO-MDC} &= (D_1 + D_2)/2 = D_{SDC} \\ R_{Side,DO-MDC} &= R_1 + R_2 = 2 \times R_{SDC}. \end{aligned} \tag{1}$$

When multiple MDC splitters are available, the encoder can choose a different splitter, instead of the duplicator, to split macroblocks. If the encoder well chooses the splitters for each macroblock, the overall R-D performance would be improved. A challenging R-D optimization problem is that how to find out a good splitter assignment. Assume the encoder choose a mode assignment (say  $\mathbf{M}$ ) for all macroblocks in the sequence and the changes of the resulting distortion and bitrate, compared with DO-MDC, are denoted as  $(\Delta D_{Side,\mathbf{M}}, \Delta R_{Side,\mathbf{M}})$  Then, the new distortion and bitrate are:

$$\begin{aligned} D_{Side,RDO} &= D_{Side,DO-MDC} + \Delta D_{Side,\mathbf{M}}, \\ R_{Side,RDO} &= R_{Side,DO-MDC} + \Delta R_{Side,\mathbf{M}}. \end{aligned} \tag{2}$$

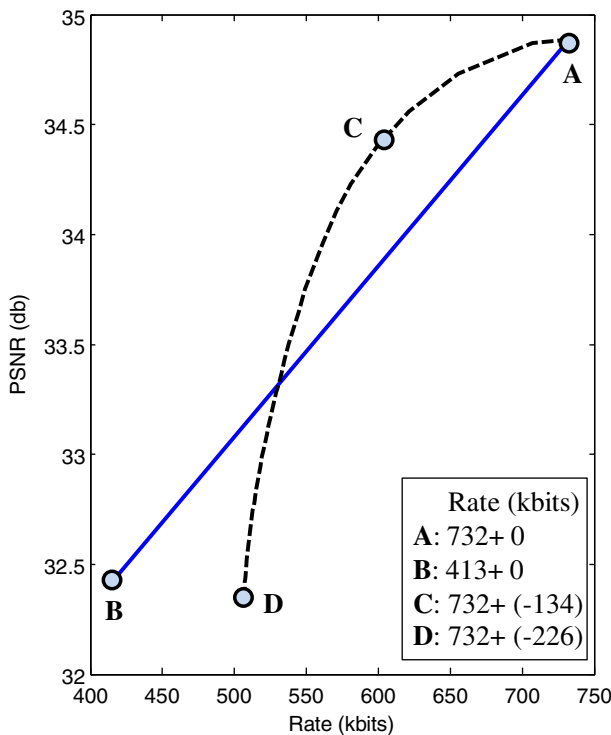
The R-D optimization problem is to find out the  $\mathbf{M}$  for better  $(D_{Side,RDO}, R_{Side,RDO})$ . To solve the problem, we propose a strategy that makes  $(\Delta D_{Side,\mathbf{M}}, \Delta R_{Side,\mathbf{M}})$  satisfy the equation:

$$\frac{d(\Delta D_{Side,\mathbf{M}})}{d(\Delta R_{Side,\mathbf{M}})} = \frac{d(D_{Side, DO-MDC})}{d(R_{Side, DO-MDC})} = \frac{1}{2} \times \frac{d(D_{SDC})}{d(R_{SDC})}. \tag{3}$$



In Eq. (3), the first two terms represent the slope of the R-D curve, which means the ratio of distortion improvement over bitrate consumption. Larger ratio indicates that increasing little bitrate can improve distortion greatly. If we try to divide bitrate resource into two targets as Eq. (2), the best strategy is to keep the slopes of two targets the same. Otherwise, we can easily move rates from the target with the small slope to the target with the large one, and the overall R-D performance will thereby improved. Eq. (3) expresses this concept.

To better understand the proposed method's characteristic on R-D performance, we take an example in Fig. 7 to illustrate the concept of Eq. (3). The Foreman CIF sequence is encoded by a MDC system and its R-D curve is shown in Fig. 7, where there are four R-D points, **A**, **B**, **C**, and **D**. In the right-down legend, the bitrates of four R-D points are shown in the form of  $R_{Side,DO-MDC} + \Delta R_{Side,M}$ . Points **A** and **B** are the R-D points of DO-MDC, where only the duplicator is adopted, so  $\Delta R_{Side,M}$  equals to zero. With other splitters adopted to replace the duplicator for some macroblocks, the R-D points move along the dashed curve from point **A** to **C**. Keeping adopting the splitters for more macroblocks, the R-D curve will go to point **D**. For the R-D curve in Fig. 7, it is observed that point **C** has the best R-D performance and that the bitrate allocated to  $\Delta R_{Side,M}$  is too small for point **A** and too large for point **D**. Since different splitting-mode assignments will result in different R-D performances, Eq. (3) provides a guide to select a good splitting-mode assignment.



**Fig. 7** An example of R-D optimization

According to the concept in Eq. (3), a splitting-mode selection method is proposed. For macroblock  $i$ , the encoder firstly encodes it by DO-MDC and then try each splitter candidate. For each splitter, calculate the bitrate and distortion changes from using DO-MDC and then choose the one closest to Eq. (3).

In the proposed mode selection method, the encoder should calculate the R-D impact for each splitter candidate. However, accurate R-D impact is hard to calculate, because the distortion will propagate among frames according to traditional predictive coding scheme. For each splitter candidate applied on a macroblock, all frames that directly or indirectly reference to this macroblock should be re-encoded to calculate the distortion propagation and then the exact R-D change can be obtained. However, the computation is too complex and is not realistic. In following, we proposed a realistic method to estimate the R-D impact of each splitter candidate.

### 3.2 Rate-distortion estimation

Compared with DO-MDC, if a macroblock  $i$  is encoded by a splitter mode  $j$ , rather than the duplicator, the bitrate change due to this macroblock is denoted by  $\Delta R_{\text{Side,mode } j}^{\text{MB } i}$  and the distortion change is by  $\Delta D_{\text{Side,mode } j}^{\text{MB } i}$ . The bitrate change can be calculated as

$$\Delta R_{\text{Side,mode } j}^{\text{MB } i} = R_{\text{Side,DO-MDC}}^{\text{MB } i} - R_{\text{Side,mode } j}^{\text{MB } i} \tag{4}$$

The distortion change,  $\Delta D_{\text{Side,mode } j}^{\text{MB } i}$ , however, is hard to be calculated because it needs to take into account all the affected macroblocks caused by motion prediction which results in distortion propagation. To reduce the complexity of distortion calculation, an estimation method is proposed as Eq. (5), where each pixel has a *distortion weight*,  $w$ , to approximate the distortion from the pixel itself and the propagation effect.

$$\Delta D_{\text{Side,mode } j}^{\text{MB } i} = \sum_{k \in \text{MB } i} w_k \times \left( d_{\text{Side,DO-MDC}}^{\text{pxl } k} - d_{\text{Side,mode } j}^{\text{pxl } k} \right), \tag{5}$$

where  $\left( d_{\text{Side,DO-MDC}}^{\text{pxl } k} - d_{\text{Side,mode } j}^{\text{pxl } k} \right)$  is the distortion change of pixel  $k$  by replacing the duplicator with a splitter mode  $j$  on macroblock  $i$ . Note that uncapitalized "d" represents distortion of pixel  $k$  itself. In contrast, capitalized "D" represents distortion superimposed on the entire sequence, including the distortion on macroblock  $i$  itself and the distortion propagating to other macroblocks. In Eq. (5), if there is no propagation effect, distortion weight of each pixel will be equal to one. With propagation effects, the distortion weight is approximated by a linear model which sequentially estimates the propagated distortion of each pixel from the trajectory of motion prediction. Since distortion propagation is caused by motion prediction, the amount of propagated error should be larger if a pixel is referred by more pixels, namely, its distortion weight  $w$  should be set larger. According to this concept, we calculate  $w$  from the motion prediction trajectory. Although similar idea has been proposed in [14], there are two major differences between their approach and ours. First, we adopt pixel-level instead of macroblock-level estimation. Second, we consider that the propagated distortion will decay over time [9, 23] and thus adopt a linear model for this effect.

Take an example in Fig. 5 to illustrate how to calculate distortion weights. Figure 8(a) shows successive frames in a hierarchical B coding architecture, where the arrow signs indict the directions of motion prediction. We enlarge the first four frames in Fig. 5(b) and highlight four pixels,  $P_1$ ,  $P_2$ ,  $P_3$  and  $P_4$ , to explain the method of calculation. Since  $P_1$  and  $P_2$  are in non-reference frames, their distortion will not propagate to

other frames and thus the corresponding weights,  $w_1$  and  $w_2$ , both equal to 1. Assuming that  $P_3$  is referred by  $P_1$  and  $P_2$ , since the distortion of  $P_3$  will propagate to  $P_1$  and  $P_2$ , we add some distortion to  $P_3$  to elevate its impact on the overall distortion. In the case of Fig. 5, since  $P_1$  and  $P_2$  are non-reference pixels, the distortion propagated from  $P_3$  will stop on these two pixels. The distortion weight of  $P_3$  can be thereby calculated as  $1 + \alpha_1 + \alpha_2$ , where 1 represents the distortion of  $P_3$  itself, and  $\alpha_1$  and  $\alpha_2$  represent the distortion propagated to  $P_1$  and  $P_2$ , respectively. The values of  $\alpha$  depends on motion prediction schemes of  $P_1$  and  $P_2$ . In this example,  $P_1$  is bi-predicted by  $P_3$  and  $P_4$  ( $0.5 * P_1 + 0.5 * P_4$ );  $P_2$  is uni-predicted by  $P_3$ . Many distortion estimation methods [7, 14] assume that the distortion will propagate to other pixels without any decay. By this assumption,  $\alpha_1$  and  $\alpha_2$  are 0.5 and 1, respectively. However, some coding tools will mitigate the error propagation effects, e.g., de-blocking filter, sub-pixel interpolation filter, quantizer, and so on. Therefore, we adopt a factor,  $\alpha_{PD}$ , representing propagation decays and then  $\alpha_1$  and  $\alpha_2$  become  $0.5 \times \alpha_{PD}$  and  $\alpha_{PD}$ , respectively. Some studies [23] have proposed theoretical derivation of propagation decays. In our approach, the decay factor  $\alpha_{PD}$  is statistically determined by experiments. In the experiments, we introduced little error in a frame and observed the propagated errors in those frames that refer to this frame. The factor,  $\alpha_{PD}$ , can be thereby calculated. To conduct the experiments, four CIF sequences, *Coastguard*, *Hall*, *Harbour*, and *Soccer* were adopted and encoded by hierarchical B picture structures with QPs equal to 16, 22, 28, and 34, respectively. We introduced errors into frames on each hierarchical layer and observed the propagated error. The experimental results are shown in Fig. 9, where the vertical axis is the observed decay factors and the horizontal axis is QP settings. It can be seen that the results of four sequences can be approached by Eq. (6), a linear function of decay factor and QP, using least square method.

$$\alpha_{PD} = 0.0032 \times QP + 0.7466. \tag{6}$$

In the example of Fig. 8, if  $P_1$  and  $P_2$  are also referred by other pixels, then the  $w_1$  and  $w_2$  will not equal to 1. The distortion of  $P_3$  will propagate not only to  $P_1$  and  $P_2$  but also to the pixels referring to them. The distortion weight of  $P_3$  will be the sum of the distortion weights of  $P_1$  and  $P_2$ , i.e.  $w_3 = 1 + 0.5 \times \alpha_{PD} \times w_1 + 1 \times \alpha_{PD} \times w_2$ .

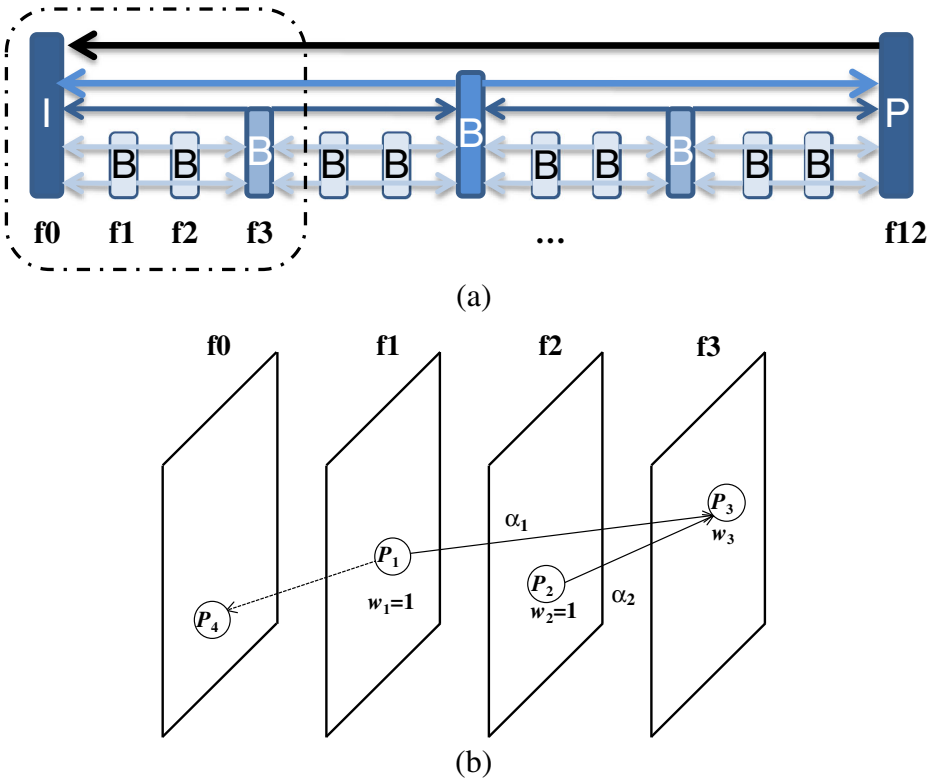
To summarize, the distortion weight of pixel  $k$  is

$$w_k = \begin{cases} 1, & \text{if } k \text{ is a non-reference pixel} \\ 1 + \sum_{l \in \Omega_k} \alpha_l w_l & \text{if } k \text{ is a reference pixel} \end{cases}, \tag{7}$$

where  $\Omega_k$  is the set of the pixels referring to pixel  $k$  and  $\alpha_l$  represents the distortion propagation factor which can be calculated as

$$\alpha_l = \begin{cases} \alpha_{PD}, & \text{if } l \text{ is an uni-predicted pixel} \\ 0.5 \times \alpha_{PD}, & \text{if } l \text{ is a bi-predicted pixel} \end{cases}, \tag{8}$$

where  $\alpha_{PD}$  is calculated by Eq. (6). To determine the best mode assignment, we start from non-reference frames to all the reference frames in the same GOP, so the distortion weights of all pixels in the GOP can be derived from Eq. (7). And then the bit-rate and distortion impact of each mode on each individual macroblock can be calculated by Eq. (4) and Eq. (5), respectively. Finally, the best mode assignment for each macroblock can be found by Eq. (3). The proposed mode selection method is summarized in section 3.D.



**Fig. 8** Illustration of Error Weight

### 3.3 Rate-distortion optimization on a packet loss channel

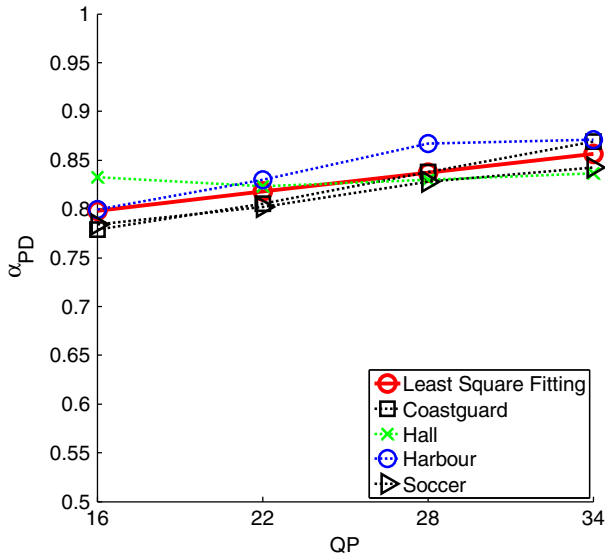
In section 3.A, the proposed mode selection method is discussed in an ideal MDC channel. In following, we will extend it to a general packet loss channel.

Assume a frame is divided into two descriptors. Each descriptor forms a packet and is transmitted through a packet loss network. In the decoder side, the frame can be perfectly reconstructed if two descriptors are received. If any description loss, the data will be recovered by the estimation method proposed in section 2. For a macroblock  $MB_i$ , let  $D_{2D}^{MB_i}$  denote the distortion superimposed on the whole sequence when two descriptions are received, and  $D_{1D}^{MB_i} = D_{0D}^{MB_i}$  when one and no descriptor is received, respectively. Note that, for a macroblock, the distortion superimposed on the sequence includes the distortion caused by itself and the distortion propagated to other macroblocks in the sequence.

Assuming that the distortion caused by the loss of a number of macroblocks is mutually un-correlated [14]. Given packet loss rate,  $P_l$ , the expectation of the distortion is derived as

$$D_{Pl} = (1 - P_l)^2 \cdot \left( \sum_i D_{2D}^{MB_i} \right) + 2(1 - P_l)P_l \cdot \left( \sum_i D_{1D}^{MB_i} \right) + P_l^2 \cdot \left( \sum_i D_{0D}^{MB_i} \right) \quad (9)$$

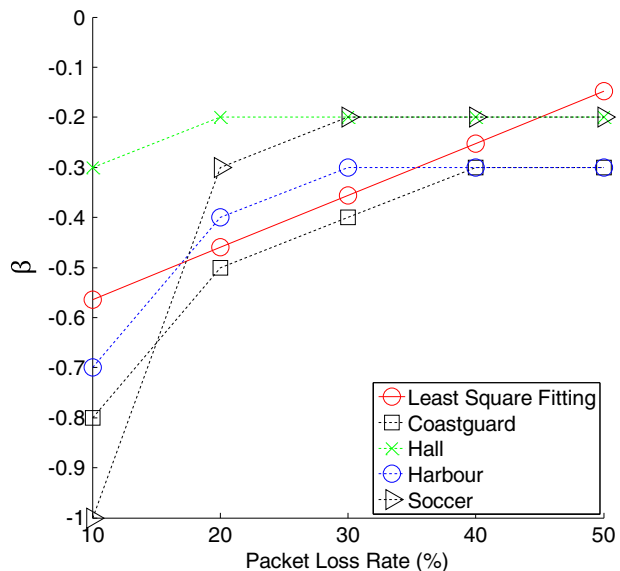
The last part of Eq. (9) can be neglected for low  $P_l$ .



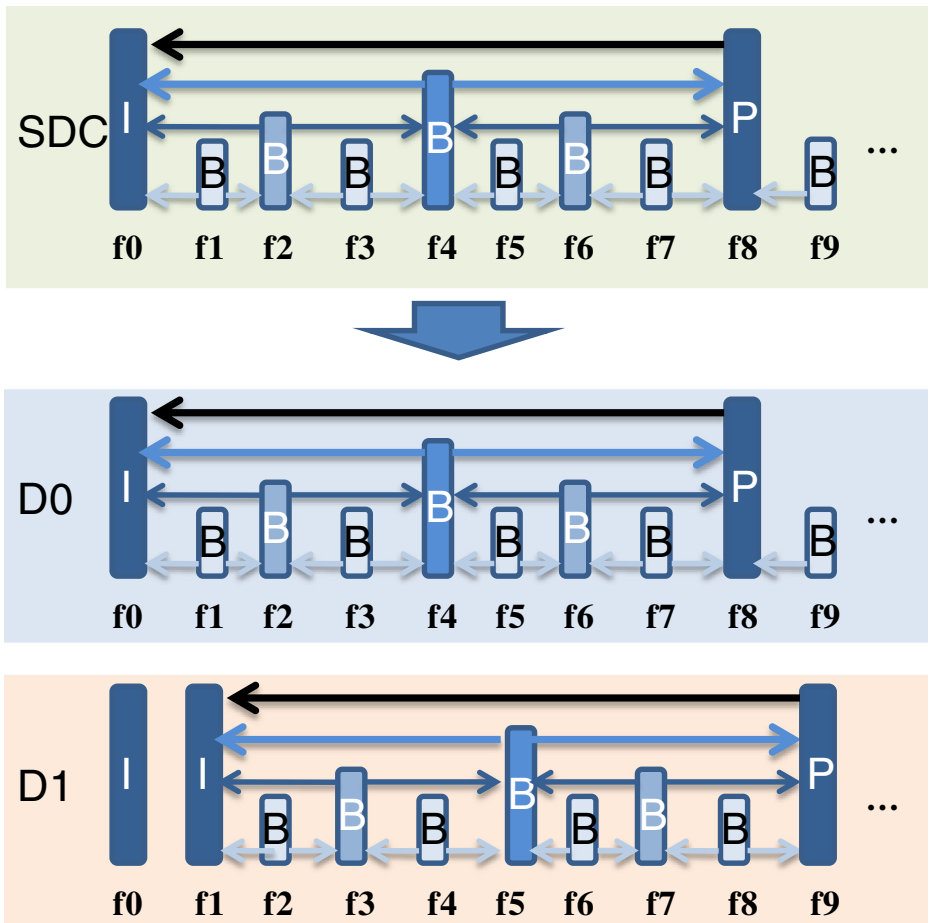
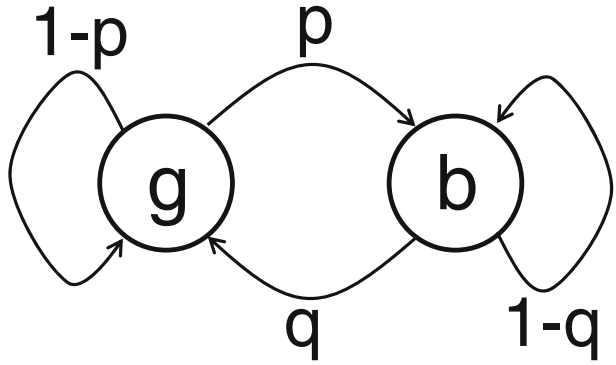
**Fig. 9** Fitting result of propagation decays factors,  $\alpha_{PD}$

To see how Eq. (9) is affected by mode assignment, we firstly consider RO-MDC where the distortion when one or two descriptors are received is equal to the distortion of SDC, namely,  $\sum_i D_{2D,DO-MDC}^{MB i} \sum_i D_{1D,DO-MDC}^{MB i} = D_{SDC}$ . When two descriptors are received, since all information distributed into descriptors are collected on the decoder side without any loss, we assume  $D_{2D}^{MB i}$  would not change. The mode assignment will result in distortion change only when there is any description loss. Let

**Fig. 10** Fitting result of  $\beta$  in Eq. (13)



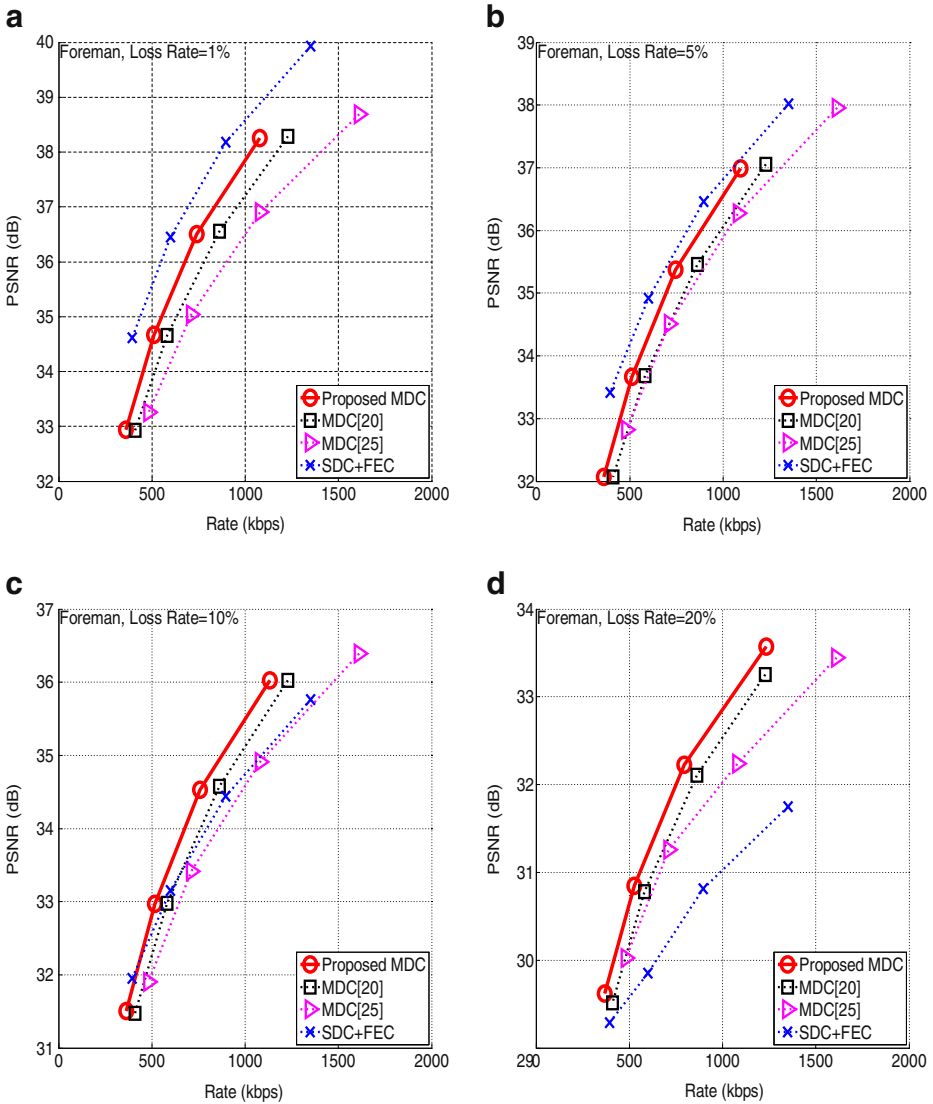
**Fig. 11** Two-state discrete-time Markov chain channel model



**Fig. 12** Illustration of Zhu et al.'s method [25]

$\Delta D_{P_i, \mathbf{M}}$  denote the distortion change when assignment  $\mathbf{M}$  is applied and one description is lost. With mode assignment, Eq. (9) will be re-written as

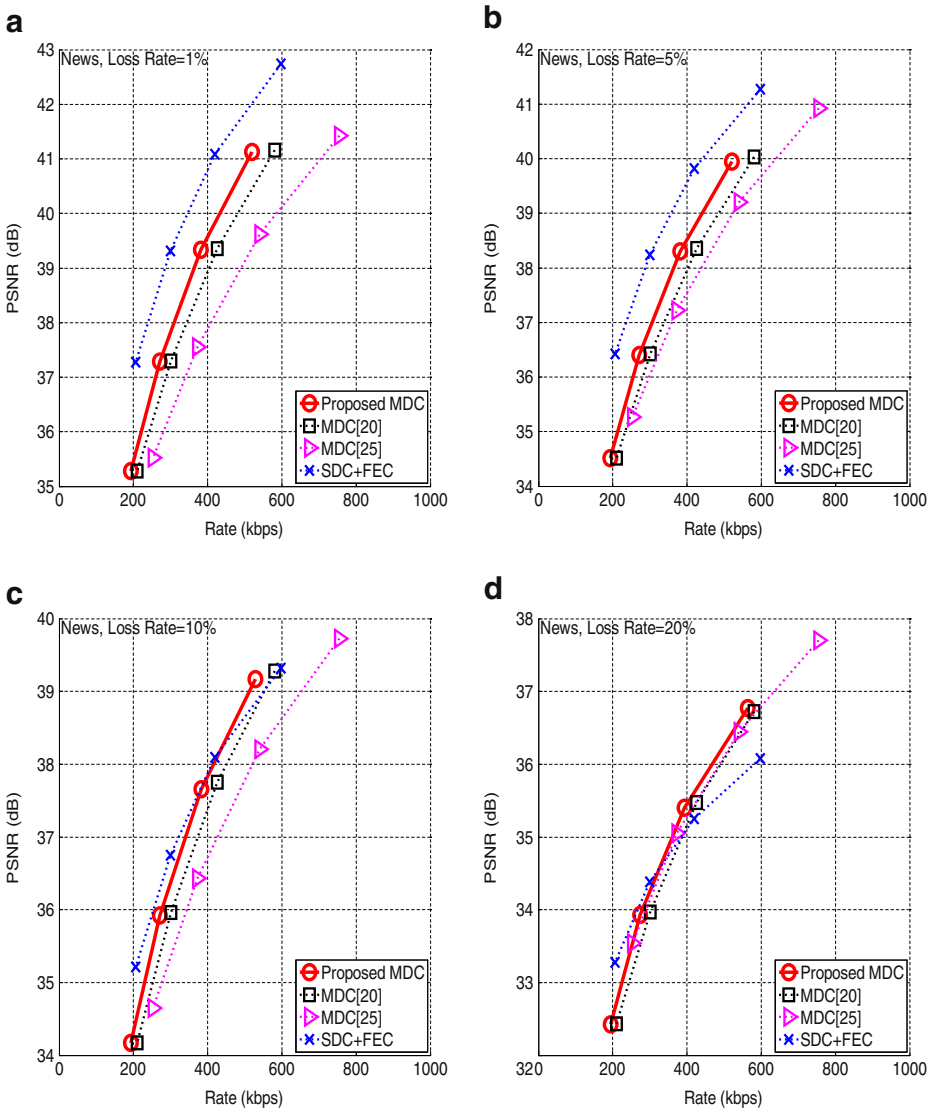
$$\begin{aligned}
 D_{P_i, \text{RDO}} &= (1 - P_l)^2 \times D_{\text{SDC}} + 2(1 - P_l)P_l \times (D_{\text{SDC}} + \Delta D_{P_i, \mathbf{M}}) \\
 &= \left\{ (1 - P_l)^2 + 2(1 - P_l)P_l \right\} \times D_{\text{SDC}}^{\text{Seq}} + 2(1 - P_l)P_l \times \Delta D_{P_i, \mathbf{M}}.
 \end{aligned}
 \tag{10}$$



**Fig. 13** R-D performance of the Forman Sequence. **a** Packet loss rate=1 %. **b** Packet loss rate=5 %. **c** Packet loss rate=10 %. **d** Packet loss rate=20 %

On the other hand, the bit-rate change taking account for mode assignment  $\mathbf{M}$  is denoted as  $\Delta R_{P_l, \mathbf{M}}$ . According to Eq. (3), the assignment  $\mathbf{M}$  should satisfy

$$\frac{2(1 - P_l)P_l}{1} \times \frac{d(\Delta D_{P_l, \mathbf{M}})}{d(\Delta R_{P_l, \mathbf{M}})} = \frac{\{(1 - P_l)^2 + 2(1 - P_l)P_l\}}{2} \times \frac{dD_{SDC}}{dR_{SDC}} \quad (11)$$



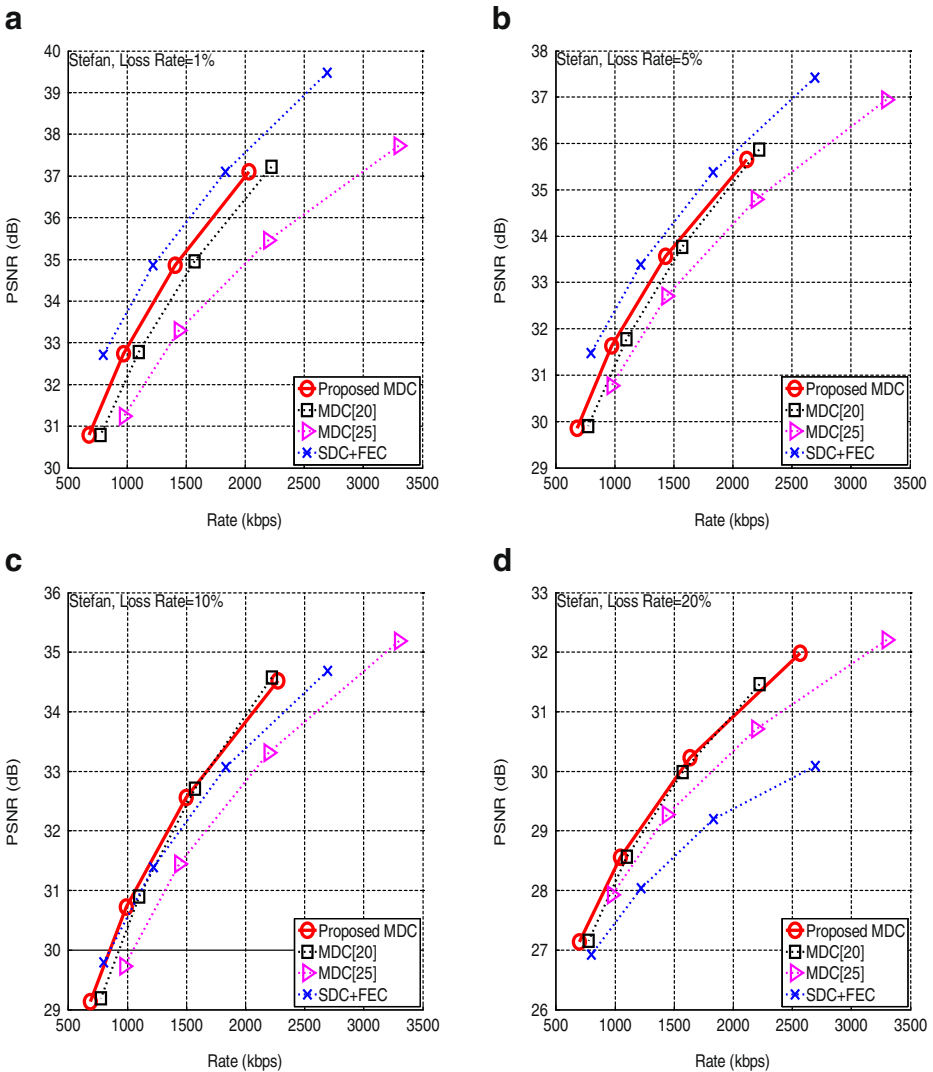
**Fig. 14** R-D performance of the News Sequence. **a** Packet loss rate=1 %. **b** Packet loss rate=5 %. **c** Packet loss rate=10 %. **d** Packet loss rate=20 %



which can be rewritten as

$$\frac{d(\Delta D_{P_l, \mathbf{M}})}{d(\Delta R_{P_l, \mathbf{M}})} = \frac{\left\{ (1 - P_l)^2 + 2(1 - P_l)P_l \right\}}{2 \times 2(1 - P_l)P_l} \times \frac{dD_{SDC}}{dR_{SDC}} \tag{12}$$

Using Eq. (12) instead of Eq. (3), the best assignment  $\mathbf{M}$  under packet loss network can be found using the method proposed in section 3.A.



**Fig. 15** R-D performance of the Stefan Sequence. **a** Packet loss rate=1 %. **b** Packet loss rate=5 %. **c** Packet loss rate=10 %. **d** Packet loss rate=20 %

### 3.4 Summary of proposed rate-distortion mode selection method

Let  $N$ ,  $I$ , and  $P$  respectively denote GOP length, the number of macroblocks in one frame, and the number of pixels in one frame.  $\Lambda()$  is a function, which indicates the frame encoding order. The proposed mode selection method is shown in the following:

---

*/\*Step1: Record R-D performance and motion prediction trajectory \*/*

**For frame  $n = \Lambda(1)$  to  $\Lambda(N)$  in a GOP**

**For macroblock  $i = 1$  to  $I$  in the frame  $n$**

Encode macroblock  $i$  by SDC codec.

Record  $R_{\text{SDC}}^{\text{MB } i}$  and  $D_{\text{SDC}}^{\text{MB } i}$ .

Record the motion vectors.

**end**

**end**

*/\* Step2: Calculate distortion weights \*/*

**For frame  $n = \Lambda(N)$  to  $\Lambda(1)$  in a GOP.**

**For pixel  $p = 1$  to  $P$  in the frame  $n$**

Calculate distortion weights of pixel  $p$  by Eq.(7).

**end**

**end**

*/\* Step3: optimize R-D performance\*/*

**For frame  $n = \Lambda(1)$  to  $\Lambda(N)$  in a GOP**

**For macroblock  $i = 1$  to  $I$  in the frame  $n$**

Calculate  $\Delta D_{\text{mode } j}^{\text{MB } i}$  and  $\Delta R_{\text{mode } j}^{\text{MB } i}$  by Eq.(4) and Eq.(5) for each splitting mode  $j$ .

Select the best mode by Eq.(3) or Eq.(12).

**end**

**end**

---

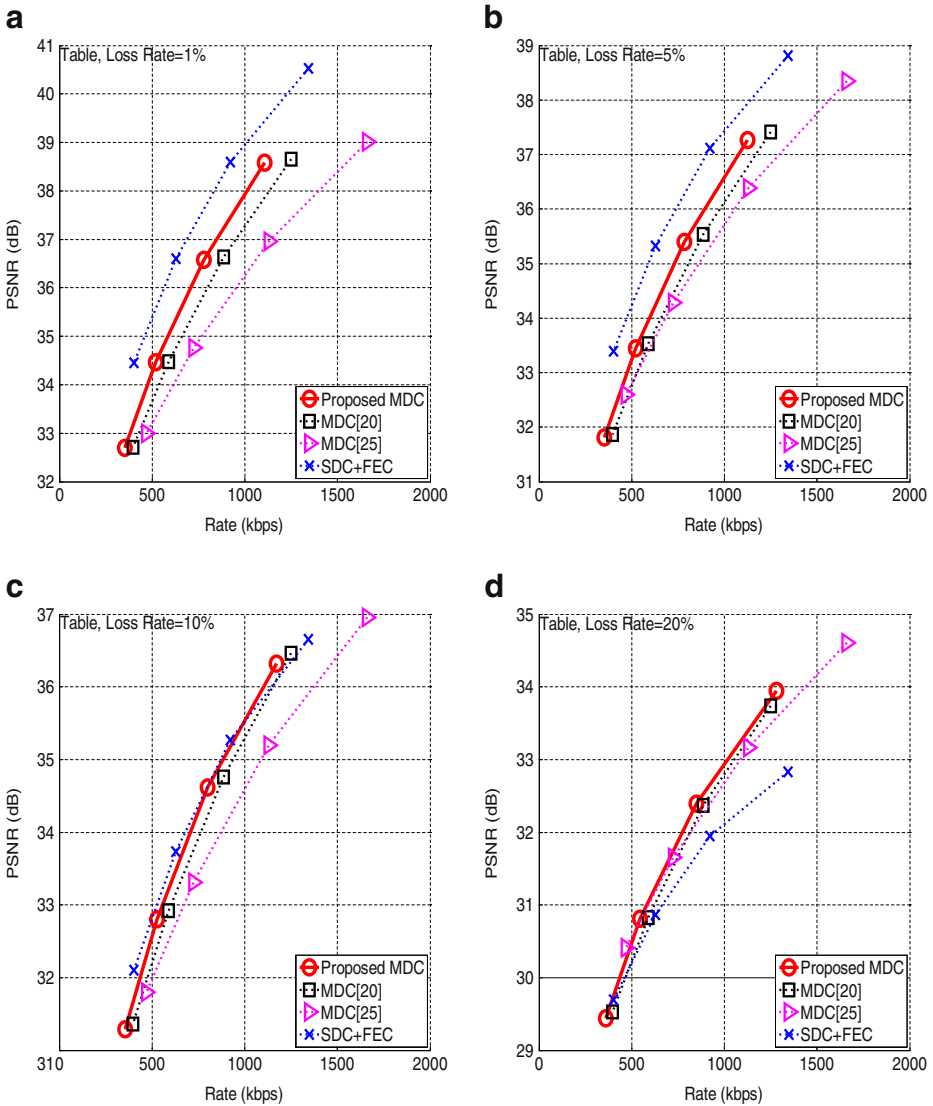
In Eq. (3) and Eq. (12), the R-D slope of SDC,  $d(D_{\text{SDC}})/d(R_{\text{SDC}})$ , is related to adopted SDC codec. For H.264/AVC codec, the slope can be approximated by

$$\frac{dD_{\text{SDC}}}{dR_{\text{SDC}}} = \beta \times 2^{\left(\frac{QP-12}{3}\right)}, \quad (13)$$

where  $\beta$  is empirically fitted as  $-0.85$  in [16, 17]. However, this value is not good enough for the proposed system. To clarify this, experiments have been conducted to find a better  $\beta$  for our framework. We choose four CIF versions of sequences, Coastguard, Hall, Harbour, and Soccer and encode them with different combinations of QPs (22, 25, 28, and 31) and packet loss rates (10 %, 20 %, 30 %, 40 %, and 50 %). For each packet loss rate, we calculate mode assignments

by using ten values of  $\beta$ , equally distributed from 0 to 1. Among these ten values, the one with best R-D performance by B-D method is selected. The best  $\beta$  value selected for each packet loss rate is shown in Fig. 10. It can be found that when packet loss rate increases, the optimal  $\beta$  value increases. We adopt a linear model to fit the relation between  $\beta$  and packet loss rates. The least square fitting result is:

$$\beta = 1.04P_l - 0.67 \tag{14}$$



**Fig. 16** R-D performance of the Table Tennis Sequence. **a** Packet loss rate=1 %. **b** Packet loss rate= 5 %. **c** Packet loss rate=10 %. **d** Packet loss rate=20 %

**Table 2** BD results of the proposed framework on packet loss channels. The column of "Comparing with the MDC system in [20]" shows the BD difference between the proposed method and the MDC system in [20]; The column of "Comparing with the MDC system in [25]" shows the difference between the proposed method and the MDC system in [25]; The column of "Comparing with SDC system with FEC " shows the difference between the proposed method and SDC system with FEC

Sequence	$P_l$	Comparing with the MDC system in [20]		Comparing with the MDC system in [25]		Comparing with the SDC system with FEC	
		BD-PSNR (dB)	BD-Rate (%)	BD-PSNR (dB)	BD-Rate (%)	BD-PSNR (dB)	BD-Rate (%)
Foreman (CIF)	1 %	0.661	-12.632	1.269	-23.733	-0.940	22.315
	5 %	0.551	-11.486	0.639	-13.625	-0.503	12.162
	10 %	0.467	-10.846	0.871	-20.364	0.476	-13.205
	20 %	0.391	-11.175	0.698	-21.059	1.498	-40.785
News (CIF)	1 %	0.602	-9.729	1.597	-24.462	-1.340	26.812
	5 %	0.526	-9.120	0.905	-15.441	-1.192	25.972
	10 %	0.437	-8.304	1.040	-19.271	-0.230	4.080
	20 %	0.288	-6.729	0.119	-3.011	0.111	-3.058
Stefan (CIF)	1 %	0.615	-9.914	1.716	-27.271	-0.812	15.421
	5 %	0.369	-6.848	0.909	-16.571	-0.621	13.074
	10 %	0.168	-3.735	0.950	-19.248	0.282	-6.733
	20 %	0.135	-3.638	0.521	-13.661	1.261	-29.664
Table Tennis (CIF)	1 %	0.612	-11.119	1.391	-24.639	-1.136	24.652
	5 %	0.466	-9.234	0.693	-13.979	-0.970	23.005
	10 %	0.319	-7.081	0.827	-17.950	-0.139	3.105
	20 %	0.207	-5.609	0.160	-4.822	0.576	-15.033

Even though the data do not exactly distributed linearly, we found that the fitting error is not sensitive. Since simple linear model can provide acceptable performance, we adopt linear fitting results to conduct the following experiments.

### 4 Experimental result

In this section, the performance of the proposed mode selection method was evaluated under both packet loss channels and ideal MDC channels. We also evaluate the computational complexity of the proposed method.

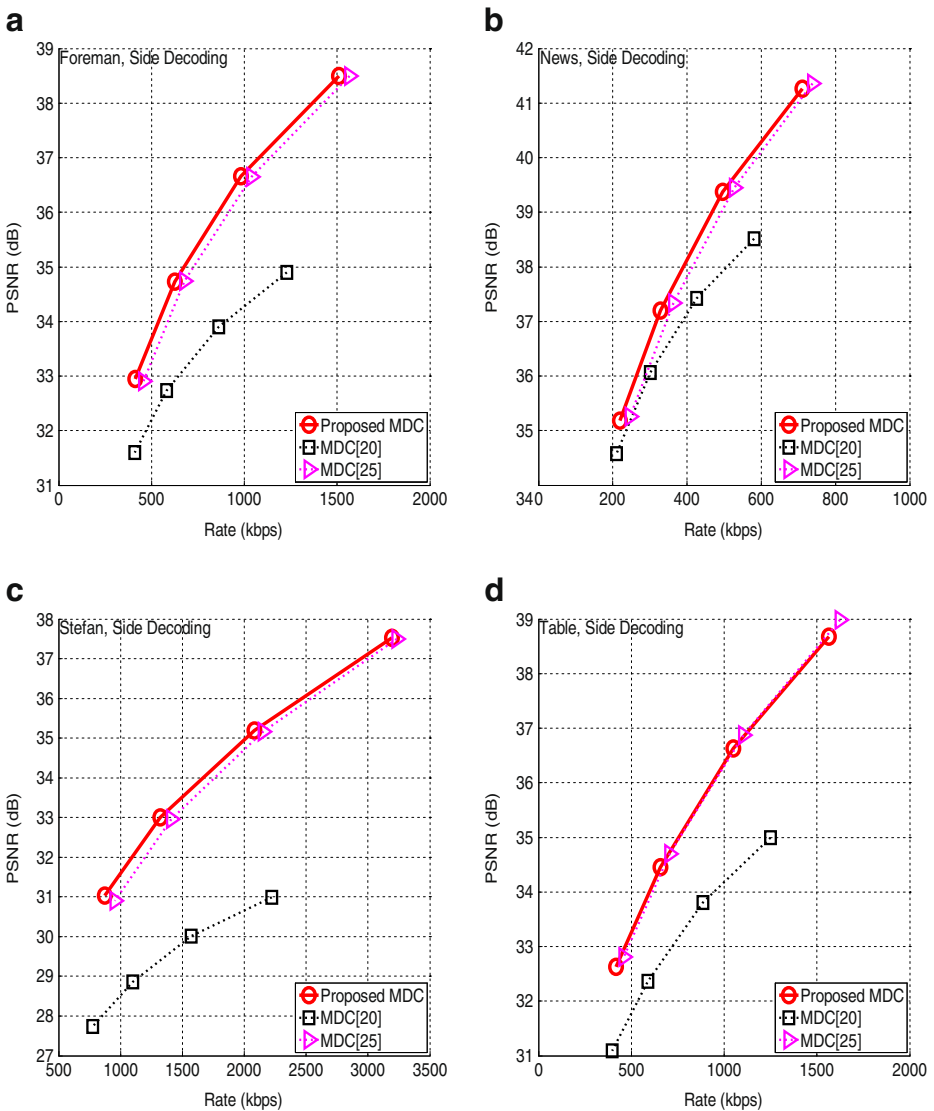
**Table 3** Side decoding BD results of the proposed framework. The column of "Comparing with the MDC system in [20]" and "Comparing with the MDC system in [25]" are defined as Table 2

Sequence	Comparing with the MDC system in [20]		Comparing with the MDC system in [25]	
	BD-PSNR (dB)	BD-Rate (%)	BD-PSNR (dB)	BD-Rate (%)
Foreman(CIF)	1.964	-40.372	0.159	-3.603
News(CIF)	0.941	-17.169	0.174	-3.263
Stefan(CIF)	3.651	-61.215	0.089	-1.758
Table Tennis(CIF)	1.847	-36.846	0.043	-0.915

### 4.1 Packet loss performance

For packet loss channels, a two-state discrete-time Markov chain was adopted as our channel model, which is shown in Fig. 11 where there are two chain states,  $\{g(ood), b(ad)\}$ . A packet transmitted at slot  $n$  is successfully received if the corresponding state is good (i.e.,  $X_n=g$ ); otherwise, it is lost. The transition probabilities from good to bad and vice versa are  $p$  and  $q$ , respectively. The stationary packet loss probability is  $p/(p+q)$  and average burst error length is  $1/q$ .

For conducting the experiments, four CIF sequences, Foreman, News, Stefan, and Table Tennis, were chosen. We select these sequences because they contain different types of



**Fig. 17** Side decoding R-D performance. **a** Foreman. **b** News. **c** Stefan. **d** Table Tennis

contents. Note that, for fair comparison, these sequences are different from those sequences used for the coefficient fitting described in section 3. All sequences were encoded using a dyadic hierarchical structure with 4 levels. Each slice is about 1 k bytes and transmitted in a packet. Four packet loss rates, 1 %, 5 %, 10 %, and 20 %, were chosen for evaluation, and average burst length was 10. For the optimized encoding, it is better to set smaller QPs for the frames that are referenced by other frames. In the Joint Scalable Video Model 11 (JSVM11) [15], QPs of the B frames at level-1 equal to the QPs of the I/P frames plus 4, and the QPs at level- $i$  increase by 1 from level- $(i-1)$ , with  $i \geq 2$ .

The proposed method was compared with three video delivering approaches. First one is the MDC system in [20], where key frames are duplicated, reference B frames are spatially split, and non-reference B frames are temporally split. Second one is the MDC system proposed by Zhu et al. [25] in which each test sequence is duplicated into two and then encoded by hierarchical B structure with staggered key frames in the two sequences. For example, if one sequence is encoded with the structure shown in Fig. 12 where frames  $f_0, f_8, f_{16}, \dots$  are I frames, then the other one will have frames  $f_1, f_9, f_{17}, \dots$  encoded as I frames. This approach is characterized by that each frame at levels 0, 1, or 2 of one sequence will be at level 3 of the other sequence and vice versa, resulting in two fidelities of each frame. Finally, we also compare the performance of the proposed method with single description video coding with forward error correction. The experimental settings in [8] were adopted, where an (100, 90) Reed-Solomon code is adopted to protect video packets.

The resulting R-D curves were shown in Figs. 13, 14, 15 and 16. Bjontegarrd bit rate savings (BD-rate) and PSNR gains (BD-PSNR) [3] are calculated using the methodology presented in [25] and shown in Table 2. It is observed that, compared with other MDC approaches, the proposed method has the best performance.

For CIF sequences, compared with the MDC system in [20], the proposed method has significant improvement when packet loss rate is low (0 %~10 %). As the packet loss rate increases (10 %~20 %), the proposed method still performs better, although the improvement becomes moderate. However, if packet loss rate further increases, resulting in one descriptor is totally lost, the performance gap between the proposed method and the MDC in [20] will be turned to increase again, which is presented in the next subsection. Since the proposed method can adjust error resilience ability according to channel conditions, the R-D performance can be optimized for various packet loss rates, resulting in better performance than the MDC in [20] for every loss rate. Compared with the MDC system in [25], the proposed method also achieves superior performance. The performance gap is even larger. This is due to that the MDC system in [25] allocated too much redundancy for the channel with low error rates. Although the performance gap decreases as the packet loss rate

**Table 4** Center decoding BD results of the proposed framework. The column of "Comparing with the MDC system in [20]", "Comparing with the MDC system in [25]", and "Comparing with the SDC system with FEC" are defined as Table 2

Sequence	Comparing with the MDC system in [20]		Comparing with the MDC system in [25]	
	BD-PSNR (dB)	BD-Rate (%)	BD-PSNR (dB)	BD-Rate (%)
Foreman(CIF)	0.685	-12.759	1.670	-29.590
News(CIF)	0.458	-7.334	1.745	-25.943
Stefan(CIF)	0.646	-10.359	2.227	-34.174
Table Tennis(CIF)	0.576	-10.303	1.669	-28.567

increases, especially when one descriptor is totally lost which is presented in the next subsection, the overall results still show the superiority of the proposed method over the MDC system in [25]. Compared with the SDC system with FEC, the proposed method outperformed the FEC based approach under the channel with high loss rates. When loss rate decreases (about <10 %), the proposed method has worse performance than the FEC based approach. This result matches the conclusion in [8] that multiple description schemes seem to be a valid alternative to the SDC system with FEC for channels with high packet loss rates (about 10 % in the experimental result in [8]).

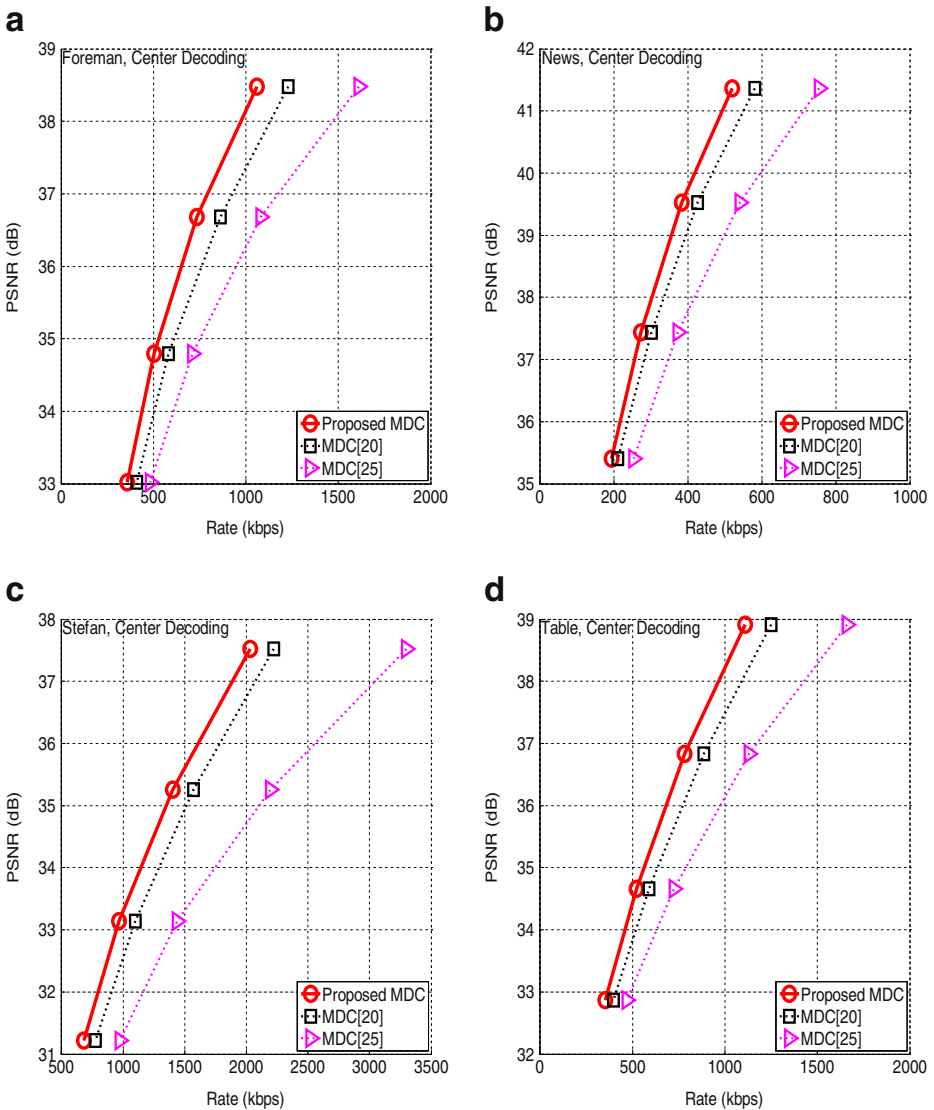


Fig. 18 Center decoding R-D performance. a Foreman. b News. c Stefan. d Table Tennis.

## 4.2 Side reconstruction performance

In following, we evaluated the performance of the proposed method and other two MDC approaches on ideal MDC channels which means that one descriptor is received without losing any information while the other is totally lost. Such performance is called side reconstruction performance and the results were shown in Table 3 and Fig. 17. It can be found that the proposed method has the best performance. Comparing with the MDC system in [20], the performance improvement can be up to 3.7 dB. This is due to that the MDC in [20] adopted fixed redundancy assignment and hence is only suitable for a certain range of packet loss rates. When the loss rate comes to 50 % (one descriptor is lost), it is obviously that the redundancy is insufficient to reconstruct well. The proposed method, however, determines the mode assignment by taking into account for channel conditions, and thus performs better. Compared with the MDC system in [25], the proposed method still has better performance even though the improvement is moderate. The reason might be that the splitting methods adopt in this paper are not good enough. If some advanced MDC tools could be adopted in the system, the performance improvement might increase.

We also showed the performance of center decoding in Table 4 and Fig. 18. When there is error free, the value of Eq. (12) goes to negative infinity. Therefore, the optimization framework would remove redundancy as much as possible, resulting in the best R-D performance.

## 4.3 Impact of high definition video content

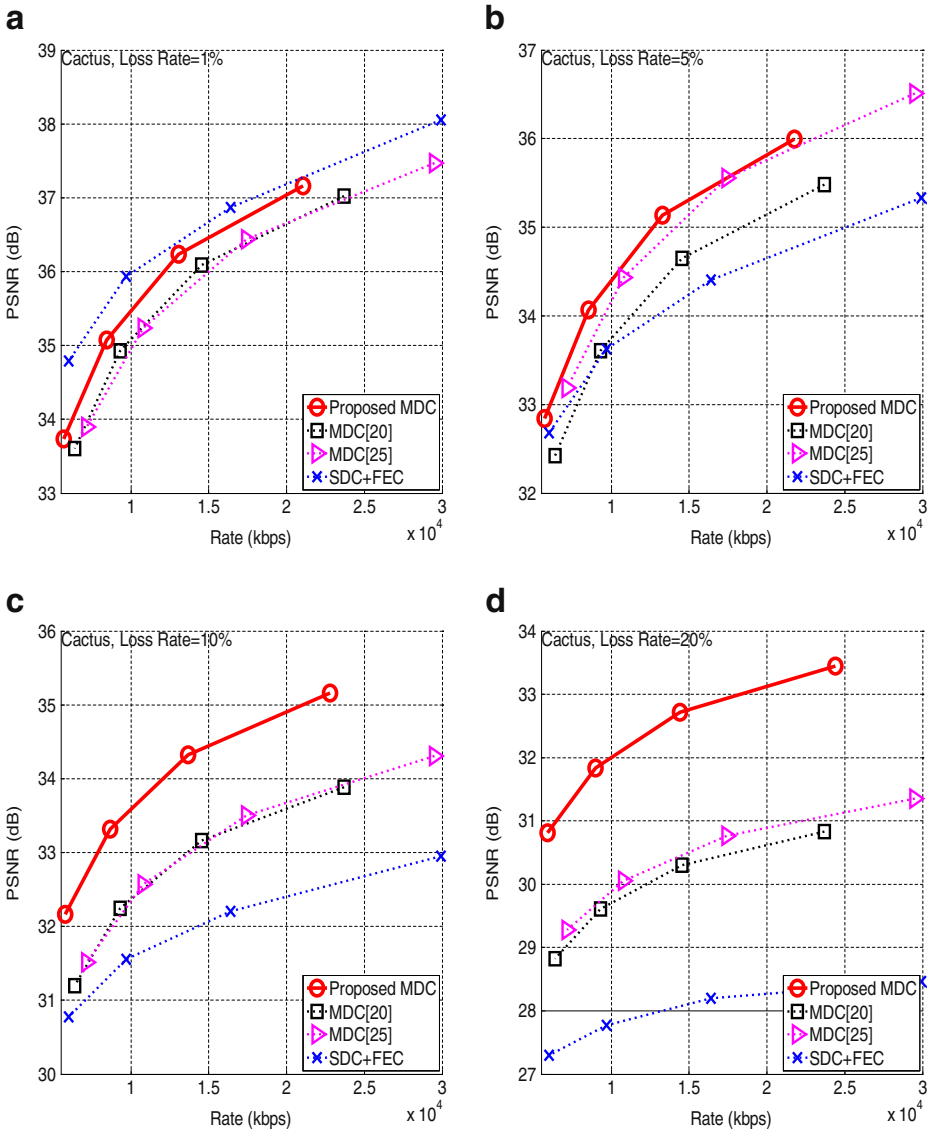
To evaluate the impact of high definition video content on the proposed method, two HD version of sequences, Cactus and Park Scene, were chosen. The results of packet loss channel are shown in Table 5, Figs. 19 and 20.

For high-definition sequences, the performance gap between the proposed method and other methods becomes larger. Compared with [20], the performance gains of the proposed

**Table 5** High definition video's BD results of the proposed framework on packet loss channels. The column of "Comparing with the MDC system in [20]", "Comparing with the MDC system in [25]", and "Comparing with the SDC system with FEC" are defined as Table 2

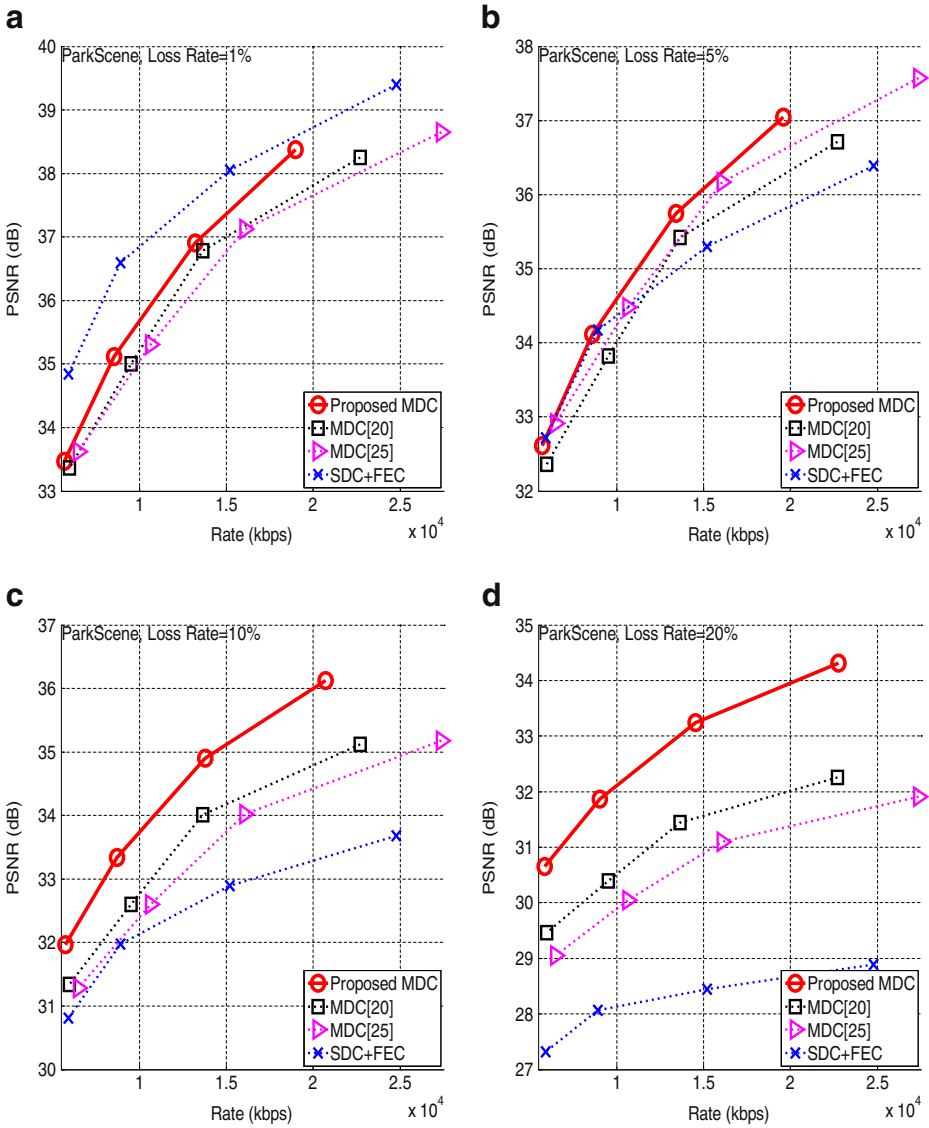
Sequence	$P_l$	Comparing with the MDC system in [20]		Comparing with the MDC system in [25]		Comparing with the SDC system with FEC	
		BD-PSNR (dB)	BD-Rate (%)	BD-PSNR(dB)	BD-Rate (%)	BD-PSNR(dB)	BD-Rate (%)
Cactus (1080p)	1 %	0.412	-14.536	0.489	-17.047	-0.403	18.436
	5 %	0.686	-24.878	0.192	-7.941	0.858	-34.249
	10 %	1.271	-43.219	1.260	-44.579	2.149	-69.775
	20 %	2.381	-74.590	2.179	-71.244	4.395	-100.000
Park scene (1080p)	1 %	0.425	-10.604	0.634	-14.876	-0.989	31.953
	5 %	0.527	-14.309	0.323	-8.457	0.423	-12.351
	10 %	0.937	-25.447	1.357	-35.339	1.840	-48.776
	20 %	1.631	-44.917	2.298	-59.145	4.357	8.124





**Fig. 19** R-D performance of the Cactus Sequence. **a** Packet loss rate=1 %. **b** Packet loss rate=5 %. **c** Packet loss rate=10 %. **d** Packet loss rate=20 %

method is up to 2.38 dB. Compared with [25], similar results can be observed. Compared with the SDC system with FEC, as long as loss rate is larger than 5 %, the proposed MDC system outperform the FEC based approach. The result implies that the improvement of the proposed method increases when the video resolution increases. This might be because, for larger resolution, the rate-distortion optimization can be operated in finer granularity. This property makes the proposed method as a potential approach in next generation video delivering applications.



**Fig. 20** R-D performance of the Park Scene Sequence. **a** Packet loss rate=1 %. **b** Packet loss rate=5 %. **c** Packet loss rate=10 %. **d** Packet loss rate=20 %

#### 4.4 Complexity analysis of the proposed MDC codec

The proposed MDC framework has to perform extra computations in both the encoder and the decoder for rate-distortion analysis and error concealment. To quantify the computational complexity of the encoder and the decoder, we have tested the

**Table 6** Encoding time comparison. Step1, 2, and 3 of proposed MDC are described in section 3.D

H.264/AVC (JM16.0)			
Encoding time (ms)	1.21*10 <sup>5</sup>		
Proposed MDC			
Encoding time (ms)	1.26*10 <sup>5</sup>		
	Step 1	Step 2	Step3
Encoding time (ms)	1.22*10 <sup>5</sup>	1409	2730

proposed MDC codec and the H.264/AVC codec (JM16.0) on an Intel i5 3.1GHz CPU with 8GB RAM. The video sequence used is CIF Foreman sequence at 30 frames per second (a total of 300 frames). Table 4 shows the encoding time comparison, while Table 5 shows the decoding time comparison.

As one can see from Tables 6 and 7, compared with H.264/AVC, on the encoder side, the complexity does increase slightly (about 4 % higher on average), and it is negligible compared to the baseline implementation. On the decoder side, since the error concealment process (spatial/temporal) is involved, the complexity is much larger than H.264/AVC (about 80 % higher on average). However, this complexity overhead also appears in most of other MDC codecs. And, the proposed decoder still can meet real time decoding requirement easily.

## 5 Conclusion and future work

In this paper, we propose a rate-distortion optimization framework for MDC systems. With the proposed framework, the encoder can dynamically adjust coding strategy according to both video contents and channel conditions. Experimental results show that the proposed optimization framework improves coding efficiency significantly.

Although the proposed technique can optimize coding strategy for different channel conditions, the improvement is moderated in the channels with large error rates. This might be due to the MDC tools adopted in this paper are not good enough to deal with these channels well. If more MDC tools can be adopted in the proposed framework, it is possible to further improve R-D performance in the channels with large errors. Based on the proposed results, more detail analysis on designing splitters capable of handling the channels with large errors will be conducted in the future for the design of a more efficient MDC tool.

**Table 7** Decoding time comparison

H.264/AVC (JM16.0)			
Decoding time (ms)	1702		
Proposed MDC			
Decoding time (ms)	3061		
	H.264/AVC Decoding	Concealment	
Decoding time (ms)	1751	1310	

## References

1. Apostolopoulos JG (2000) Error-Resilient Video Compression Through the Use of Multiple States. Proc IEEE Intel Conf Image Process (ICIP)
2. Bernardini R, Durigon M, Rinaldo R, Celetto L, Vitali A (2004) Polyphase Spatial Subsampling Multiple Description Coding of Video Streams with H.264. Proc IEEE Intel Conf Image Process (ICIP)
3. Bjontegaard G (2008) Improvement of the BD-PSNR model. VCEG document VCEG-A111, ITU-T SG16/Q6, 35th VCEG Meeting
4. Campana O, Contiero R (2006) An H.264/AVC Video Coder Based on Multiple Description Scalar Quantizer. Proc IEEE Asilomar Conf Signals Syst Comput (ACSSC)
5. Comas D, Singh R, Ortega A (2001) Rate-distortion optimization in a robust video transmission based on unbalanced multiple description coding. Proc IEEE Int Work Multimed Signal Process, pp 581–586
6. Comas D, Singh R, Ortega A, Marques F (2003) Unbalanced multiple-description video coding with rate-distortion optimization. EURASIP J Appl Sig Process 2003:81–90
7. Correia P, Assuncao P, Silva V (to appear) Multiple Description of Coded Video for Path Diversity Streaming Adaptation. IEEE Trans Multimed
8. Durigon M, Rinaldo R, Vitali A (2005) Comparison Between Multiple Description and Single Description Video Coding With Forward Error Correction. Proc IEEE Work Multimed Signal Process
9. Farber N, Stuhlmuller K, Girod B (1999) Analysis of error propagation in hybrid video coding with application to error resilience. Proc IEEE Intel Conf Image Process (ICIP)
10. Gao S, Gharavi H (2006) Multiple Description Video Coding over Multiple Path Routing Networks. Proc Intl Con Digit Commun Process (ICDP)
11. Hsiao C-W, Tsai W-J (2010) Hybrid multiple description coding based on H.264. IEEE Trans Circ Syst Video Technol 20(1):76–87
12. Jia J, Kim HK (2006) Polyphase downsampling based multiple description coding applied to H.264 Video coding. IEICE Trans Fundam Electron Commun Comput Sci E89-A(6):1601–1606
13. Lin C-S, Syu W-T (2010) A fine-grained balancing scheme for improved scalability in P2P streaming. Multimed Tool Appl 46(1):71–91
14. Lin C, Tillo T, Zhao Y, Jeon B (2011) Multiple description coding for H.264/AVC with redundancy allocation at macro block level. IEEE Trans Circ Syst Video Technol 21(5):559–600
15. Reichel J, Schwarz H, Wien M (2007) Joint Scalable Video Model 11 (JSVM 11), Joint Video Team, Doc. JVT-X202
16. Siwei M, Gao W, Lu Y (2005) Rate-distortion analysis for H.264/AVC video coding and its application to rate control. IEEE Trans Circ Syst Video Technol 15(12):1533–1544
17. Sullivan GJ, Wiegand T (1998) Rate-distortion optimization for video compression. IEEE Sig Process Mag 15(6):76–90
18. Tillo T, Grangetto M, Olmo M (2008) Redundant slice optimal allocation for H.264 Multiple description coding. IEEE Trans Circ Syst Video Technol 18(1):59–70
19. Tsai WJ, Chen J-Y (2010) Joint temporal and spatial error concealment for multiple description video coding. IEEE Trans Circ Syst Video Technol 20(12):1822–1833
20. Tsai W-J, You H-Y (2012) Multiple description video coding based on hierarchical B pictures using unequal redundancy. IEEE Trans Circ Syst Video Technol 22(2):309–320
21. Vaishampayan VA (1993) Design of Multiple Description Scalar Quantizers. IEEE Trans Inf Theory 39
22. Wang Y, Reibman A, Lin S (2005) Multiple description coding for video delivery. Proc IEEE 93:57–70
23. Wang Y, Wu Z, Boyce JM (2006) Modeling of transmission-loss-induced distortion in decoded video. IEEE Trans Circ Syst Video Technol 16(6):716–732
24. Wiegand T, Sullivan GJ, Bjontegaard G, Luthra A (2003) Overview of the H.264/AVC video coding standard. IEEE Trans Circ Syst Video Technol 13(7):560–576
25. Zhu C, Liu M (2009) Multiple description video coding based on hierarchical B pictures. IEEE Trans Circ Syst Video Technol 19(4):511–521



**Yu-Chen Sun** received the B.S. and M.S. degrees in electronics engineering from National Chiao-Tung University (NCTU), Taiwan, in 2004 and 2006, respectively. Currently, he is pursuing the Ph.D. degree in computer science from National Chiao-Tung University. His current research interests include video/image compression, computer vision, and video signal processing.



**Wen-Jiin Tsai** received the Ph.D. degree in 1997 in computer science from National Chiao-Tung University (NCTU), Taiwan, R.O.C. She is an Assistant Professor at the Department of Computer Science of NCTU, Taiwan, R.O.C. Before joining NCTU in 2004, she was with Zinwell Corporation as a Senior R&D Manager for 6 years. Her research interests include video coding, video streaming, error-concealment, and error resilience techniques.