Redundancy Allocation Based on Weighted Mismatch-Rate Slope for Multiple Description Video Coding

Mohammad Kazemi, Razib Iqbal, Shervin Shirmohammadi

***Abstract* –Multiple Description Coding (MDC) is a robust coding technique for video transmission over error prone networks, whereby the video is encoded into multiple descriptions with some redundancy between the descriptions. This redundancy leads to error resiliency in the case of packet loss during the network transport. However, the amount of this redundancy has a critical role in MDC performance. Therefore, a crucial problem in MDC is to find what the optimum amount of redundancy budget is, and then how this redundancy budget can be optimally allocated to the frames. To solve this problem, we propose a scheme in which the redundancy budget is allocated to the frames based on the weighted mismatch-rate slopes so that this additional bitrate can attain maximum distortion reduction. The redundancy is added gradually so that fine tuning of the utilized bitrate is achievable. We have verified our proposed scheme by implementing it in H.264/AVC reference software JM16.0, and running experiments against two representative reference methods. Our experiments show that our scheme not only minimizes the end-to-end distortion with a rate-distortion performance that is better than the reference methods, especially for high PLRs, but also entirely uses the available bandwidth, unlike the reference methods.**

***Index Terms* – Multiple description coding (MDC), redundancy allocation, rate-distortion optimization, video coding, video transmission**

I. INTRODUCTION

Digital video transmission over lossy packet networks is a challenging problem due to the possible loss of synchronization and error propagation in the video signal. Multiple description coding (MDC) is a viable solution for graceful degradation of the reproduced content, especially at higher loss rate [1][2]. In MDC, the video signal is coded into multiple correlated descriptions, each transmitted separately. Each of these descriptions can be decoded independently and also, the descriptions can be merged when more than one description is available which is leading to higher video quality. If all descriptions are available, then they are decoded by the central decoder and the incurred distortion is called central distortion; on the other hand, if one or more descriptions are not received, then the side decoder is used and the resulting distortion is called side distortion [24].

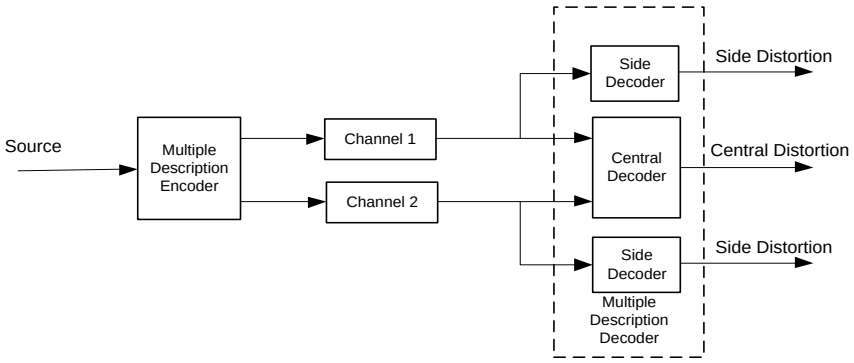


Fig. 1. Multiple description coding technique with two descriptions

In Fig. 1, the MDC technique with two descriptions (as an example) is illustrated. In this figure, both descriptions represent the same video content; however, they have different MDC parameters. The encoder can adjust these MDC parameters to make the descriptions similar or dissimilar.

M. Kazemi is with the Department of Electrical Engineering, University of Isfahan, Isfahan, Iran (e-mail: m.kazemi@eng.ui.ac.ir).

R. Iqbal is with the Department of Computer Science, Missouri State University, USA (e-mail: riqbal@missouristate.edu).

S. Shirmohammadi is with the Distributed and Collaborative Virtual Environments Research Laboratory, School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON K1N 6N5, Canada, and also with the Multimedia Systems Laboratory, Istanbul Sehir University, Istanbul, 34662, Turkey, (e- mail: shervin@eecs.uottawa.ca; shervinshirmohammadi@sehir.edu.tr).

More similarity will lead to better error resiliency, but will also lead to more redundancy and higher bitrate in the transmitted video. If two identical descriptions are sent over two different channels and are successfully received, then half of the received information has no value and is redundant. Therefore, in the cases that both descriptions will probably be received correctly, such similarity is not advantageous because bandwidth is wasted for no gain. On the other hand, the more different the descriptions, the more different the side and central decoder outputs. This mismatch of the side and central decoder outputs contributes to error propagation in MDC [24], because the reconstructed reference frames at the decoder need to be the same as the reference frames at the encoder, otherwise the frames predicted from these reference frames cannot be reconstructed correctly. The reference frames at the encoder are exactly matched with the central decoder outputs. Therefore, decoding a prediction reference frame by the side decoder causes error propagation. This error keeps propagating to the subsequent P or B-frames until a correct I-frame has arrived. Hence, with more different descriptions, we will have more destructive error propagation. Error propagation is not limited to MDC, it is also a challenge in single description coding when the video stream has been sent over noisy channels [3]-[6].

Therefore, finding the optimum similarity within the descriptions such that both resource usage and error propagation are considered, becomes a difficult problem. This similarity can be controlled by the redundancy budget: with more redundancy budget we can have more similar descriptions, and vice versa. Towards this, we need to determine the amount of this redundancy budget, and also we need to efficiently allocate this redundancy budget to the frames of the video. Video frames, based on their order in the Group Of Picture (GOP), have different impacts on error propagation; and this necessitates different redundancies [7]. Therefore to optimize MDCs, we must solve a constraint optimization problem for each GOP. A common solution for the constraint problems is the Lagrangian approach. This approach is usually computationally complex because it needs to solve a system of nonlinear equations (as large as the GOP size) several times until the rate constraint is met. It might also encounter convergence problems. To solve the objective function with less effort, some fast algorithms with simplifying assumptions have

been developed [13][14]. However, these assumptions eventually affect the optimization outputs, as we will show later in this paper.

We propose a method, called maximum weighted mismatch-rate slope (WMRS), in which there is no need to solve the system of nonlinear equations commonly found in optimization algorithms. Therefore, our method does not have the convergence problem, and it has low complexity which is comparable with the fast algorithms developed in [13] and [14]. Allocating an amount of redundancy to every frame leads to distortion reduction, however, if the frame with maximum weighted mismatch-rate slope is selected then we will have the most reduction in distortion. After allocating this amount of redundancy, the slope of this frame is updated and then it competes again with the other frames for more redundancy. This procedure continues as long as the redundancy budget has not been exhausted. This redundancy budget is a portion of the total bitrate and its amount is determined taking into account the channel condition and the video content. In WMRS, we find the global optimum point with a sequence of locally optimum decisions. Furthermore, WMRS utilizes the available bandwidth efficiently because the redundancy rate is allocated in a step-by-step manner and with fine resolution. Therefore, compared to existing works, WMRS consumes the bitrate budget completely.

A. Related Work

In the literature, there is evidence of existing research addressing total redundancy calculation, e.g. [8], in which the total redundancy is equally divided among the frames; whereas in our work we allocate the redundancy to each frame in an optimum way. The optimization algorithms presented in [9][10] are based on Recursive Optimal Per-pixel Estimate (ROPE) algorithm [10]. In ROPE, in a recursive manner, the first and second moments of the reconstructed pixel value are computed, and then the expected distortion associated with that pixel is minimized. However, this method needs pixel-wise computations and hence is too complex. In [12], authors calculate the contribution of each Macroblock (MB) in motion compensation of the future frames which is then used to determine the importance of each MB in error propagation. Then, based on the MB's importance, the redundancy to be allocated to each MB is

calculated. It is noteworthy that block-level motion path analysis is computationally complex and it also needs the motion information of the future frames. Therefore, the encoder either has to wait to capture the motion information of the next frames, or it needs to estimate them, which in most cases is not possible, for example in sport videos. An optimization technique based on conventional Lagrangian optimization has been presented in [15]. This technique is suitable for a specific MDC scheme in which the MDC parameters' variation domain is naturally limited. But this MDC scheme is not compatible with the standard, and it does not provide enough redundancy tunability. Authors in [13] and [14] also solved the optimization problem with Lagrangian approach, but with some approximations in the modeling and/or in the optimization process. In particular, in [13], the mismatch model is for the case of low redundancy rate, and hence it is not accurate enough. In addition, the authors assumed a power transfer function for error propagation modeling which is only valid for low Packet Loss Rate (PLR). As a result, they formulated a very simple and straightforward solution for the Lagrange problem; however, their rate-distortion performance is not as promising as ours, as will be shown in Section VI. In [14], authors presented a more accurate modeling at a cost of higher complexity compared to [13]. However, in [14], authors did not take into account the variation of temporal error propagation factor with the redundancy rate; therefore, their rate-distortion curve is also lower than ours, as shown in Section VI. Finally, we show in Section IV that WMRS can optimally adjust the parameters so that the given bandwidth is utilized completely, which is not the case for [13] and [14].

The rest of this paper is organized as follows. The objective function for our proposed MDC optimization method is described in Section II. In Section III, we describe the process to optimally determine the redundancy budget and allocate it among the frames. Bitrate tunability of the algorithm is presented in Section IV. The complexity and how to manage delay of the algorithm are discussed in Section V. Experimental results showing the performance of the proposed method are presented in Section VI, and finally, we present our concluding remarks in Section VII.

II. THE MDC SCHEME AND THE OBJECTIVE FUNCTION

A. Preliminaries

In the literature, there are various MDC schemes, especially in spatial [19][20], temporal [21][22], frequency [12]-[18] and compressed [23] domains. In [24], we have provided a comparative study of these schemes. The MDC under study in this paper is a frequency domain scheme originally proposed in [13]. This scheme is easily implementable and has some useful features such as standard compatibility and redundancy tunability. In this two-description MDC scheme, the DCT coefficients are coded twice using two different quantization parameters (QP). The coefficients quantized with lower QP are coded as Primary slices and the coefficients quantized with higher QP are coded as Redundant slices, named as Primary coefficients and Redundant coefficients, respectively. These primary and redundant slices are alternatively inserted into the descriptions when encoding. On the receiver side, when both descriptions are available, only primary coefficients are used and redundant coefficients are discarded. Otherwise, if the primary coefficients are not received correctly, the redundant coefficients are decoded. In this MDC scheme, the redundancy rate can be easily controlled by the quantization parameter of the redundant coefficients, QP_r.

Now, for a channel with the PLR of P_0 , the expected end-to-end distortion for the i^{th} frame of the GOP, as stated in [14], is:

$$D_i = (1-P_0)D_{p,i} + \ P_0(1-P_0)D_{r,i} + \ P_0(1-P_0)\sum_{j=1}^{i-1} D_{mismatch,j}\, f[i-j] \qquad (1)$$

Note that in (1), error concealment distortion has been removed because this distortion is highly related to the error concealment technique and the motion activity of the lost area, and it slightly affects the MDC optimization. In the above equation, $D_{p,i}$ is the distortion resulting from decoding the primary coefficients, or simply, primary distortion, which is defined as follows:

$$D_{p,i} = E\left[\left(x_i - \hat{x}_{p,i}\right)^2\right] \qquad (2)$$

In (2), x_i and $\hat{x}_{p,i}$ represent the original and reconstructed primary coefficients, respectively. Similarly, $D_{r,i}$ in (1) is the redundant distortion, which is defined by (3):

$$D_{r,i} = E\left[\left(x_i - \hat{x}_{r,i}\right)^2\right] \tag{3}$$

in which $\hat{x}_{r,i}$ is the reconstructed redundant coefficient. $D_{mismatch}$ in (1) originates from the difference between the decoded primary and redundant coefficients as defined by (4):

$$D_{mismatch,i} = E\left[\left(\hat{x}_{p,i} - \hat{x}_{r,i}\right)^2\right] \tag{4}$$

The last term of (1) can be denoted as:

$$\sum_{j=1}^{i-1} D_{mismatch,j}\, f[i-j] = \delta_i \tag{5}$$

where δ_i is the accumulated mismatch from all previously decoded reference frames, and this is the term that represents the error propagation in MDC. Error propagation has a decay factor which is modeled by a power transfer function in (5), $f[i-j]$. We compute this power transfer function as per equation (6):

$$f[i-j] = a^{(i-j)} \prod_{t=j+1}^{i} (1-\beta_t) \tag{6}$$

This function includes the mismatch attenuation due to intra-coded blocks, denoted as $(1-\beta_t)$, as well as attenuation due to the deblocking filter and sub-pixel motion compensation, denoted by parameter a, which is set to 0.92 [14]. Now, equation (1) can be rewritten as:

$$D_i = (1-P_0)\left[D_{p,i} + P_0\left(D_{r,i} + \delta_i\right)\right] \tag{7}$$

By averaging the frames' distortions given by (7), we can now obtain the end-to-end distortion of a GOP as:

$$D_{e2e} = \frac{(1-P_0)}{N} \sum_{i=1}^{N} \left[D_{p,i} + P_0\left(D_{r,i} + \delta_i\right)\right] \tag{8}$$

and then the objective function as follows:

$$\min_{\{\boldsymbol{QP_p},\boldsymbol{QP_r}\}}\{D_{e2e}\} = \min_{\{\boldsymbol{QP_p},\boldsymbol{QP_r}\}}\left\{\frac{(1-P_0)}{N}\sum_{i=1}^{N}\left[D_{p,i} + P_0\left(D_{r,i} + \delta_i\right)\right]\right\}$$
$$s.t. \sum_{i=1}^{N}\left[R_{p,i} + R_{r,i}\right] \le R_t \tag{9}$$

In (9), N is the size of GOP, $\boldsymbol{QP_p}$ and $\boldsymbol{QP_r}$ are N-size vectors of quantization parameters used for primary and redundant coefficients of the frames, respectively, $R_{p,i}$ and $R_{r,i}$ are the primary and redundant rates of the i^{th} frame, and R_t is the total rate budget for the GOP under consideration.

The basic solution for the above constrained problems is Lagrangian approach. For example, the Lagrangian function of (9) can be written as follows:

$$\min_{\{\boldsymbol{QP_p},\boldsymbol{QP_r}\}}\{J\} = \min_{\{\boldsymbol{QP_p},\boldsymbol{QP_r}\}}\left\{\frac{(1-P_0)}{N}\sum_{i=1}^{N}\left[D_{p,i} + P_0\left(D_{r,i} + \delta_i\right)\right] + \lambda\left(\sum_{i=1}^{N}\left[R_{p,i} + R_{r,i}\right] - R_{r,t}\right)\right\} \tag{10}$$

where $\lambda > 0$ is the Lagrange parameter. As shown in [34], the optimal point of problem (10) is the solution for (9), but the resultant constraint may not be $R_{r,t}$ and it depends on the value of λ. By properly adjusting the value of λ iteratively, the bitrate constraint can also be satisfied. For $N = 30$, however, at each iteration of λ, we need to find two 30-size vectors $\boldsymbol{QP_p}$ and $\boldsymbol{QP_r}$ which minimize function $\boldsymbol{J}$. Therefore, for a sufficient number of λ values, as many as needed to meet the rate constraint, a system of nonlinear equations must be solved which makes this approach very complex and it might encounter a convergence problem. These issues will be more crucial for larger GOP sizes. Other approaches such as Sequential Quadratic Programming (SQP) [35] are also applicable, but as we will show later, SQP still suffers from extensive computational complexity.

However, the underlying variables are actually discrete, and solving the continuous functions for this purpose in the above scenario is not efficient. Therefore, exploiting the discrete nature of the problem, we propose our method that does not need to solve the system of nonlinear equations, and does not have any

convergence problem, plus it has low complexity.

B. Problem formulation for the quantization parameter of redundant coefficients

We first assume that $\boldsymbol{QP_p}$ has been determined; therefore, the objective function of (9) can be reduced to:

$$\min_{\{\boldsymbol{QP_r}\}}\left\{\frac{1}{N}\sum_{i=1}^{N}\left(D_{r,i}+\ \delta_i\right)\right\} \qquad s.t.\ \sum_{i=1}^{N}R_{r,i}\le R_{r,t} \tag{11}$$

where $R_{r,t}$ represents the total rate budget allocated to redundant coefficients, which we refer to as *redundancy budget* in this paper. For simplicity, in (11), we removed the coefficient $P_0(1-P_0)$ of equation (9) because it is a constant term and has no role in minimization. Therefore, as can be seen from (11), the optimal redundancy allocation does not depend on the PLR. We will discuss the role of PLR to determine the optimal redundancy budget in section III.B. Now, by substituting (5) in (11) and simplifying, we get:

$$\min_{\{\boldsymbol{QP_r}\}}\left\{\frac{1}{N}\sum_{i=1}^{N}\left(D_{r,i}+\ \Gamma_i D_{mismatch,i}\right)\right\}=\ \min_{\{\boldsymbol{QP_r}\}}\left\{\frac{1}{N}\sum_{i=1}^{N}\Delta_i\right\}=\min_{\{\boldsymbol{QP_r}\}}\{\Delta_{GOP}\} \qquad s.t\ \sum_{i=1}^{N}R_{r,i}\le R_{r,t} \tag{12}$$

where

$$\Delta_i = D_{r,i}+\ \Gamma_i D_{mismatch,i} \tag{13}$$

Δ_i represents the contribution of the redundant and mismatch distortions of frame i in the overall distortion of the GOP. Γ_i in (13) denotes the weight of the mismatch distortion and is obtained as follows:

$$\Gamma_i=\sum_{k=i+1}^{N}f[k-i]=\sum_{k=i+1}^{N}\left[a^{(k-i)}\prod_{t=i+1}^{k}(1-\beta_t)\right] \tag{14}$$

As per equation (14), Γ_i represents this fact that the mismatch in frame i ($D_{mismatch,i}$) will exist in the next $N-i$ frames, frame $i+1$ to frame N. As we mentioned earlier, the weights are not the same for frames of GOP. In the next section, we address this essential property for redundancy allocation.

III. PROPOSED METHOD

In this section, the proposed method for optimal redundancy allocation is presented. In the first subsection, given the redundancy budget, the optimum redundancy for each frame is determined using WMRS. In the second subsection, at the given PLR, the optimum redundancy budget in order to have minimum end-to-end distortion is obtained.

A. Redundancy allocation based on weighted mismatch-rate slope (WMRS)

We first analyze the behavior of Δ_i (defined in (13)) with redundancy. For this purpose, we obtain the variation of Δ_i for two frames of *Mobile* and *Foreman* sequences where GOP size = 30, $QP_p = 20$ and the same QP_r is used for all frames which varies from QP_p to $QP_p + 10$. Fig. 2 shows the variation of Δ with redundancy for 5th and 25th frames as well as Δ_{GOP} (the average of Δ_i's, as defined in (12)). The absolute amount of redundancy does not make sense and hence we use normalized redundancy as the horizontal axis of the figures. The normalized redundancy is defined as follows:

$$Normalized\ Redundancy = \frac{R_{r,t}}{R_{p,t}} \tag{15}$$

where $R_{p,t}$ is the total rate needed for coding the primary coefficients. As mentioned earlier, in this sub-section we assumed that $\boldsymbol{QP_p}$ is known and fixed, and therefore, $R_{p,t}$ is constant.

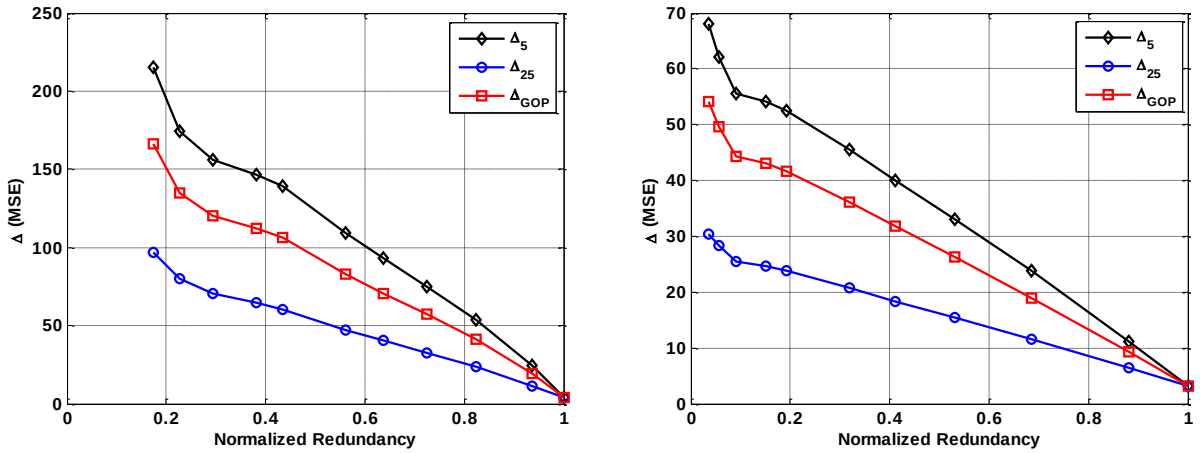


Fig. 2. Variation of Δ_i when equal QP_r is used for all frames for (a) *Mobile* and (b) *Foreman* sequences

From Fig. 2 we can observe that:

a) As the normalized redundancy increases, Δ_i decreases because at higher redundancies QP_p and QP_r are more similar. At a normalized redundancy equal to 1 ,$QP_p = QP_r$, $D_{mismatch,i} = 0$, and Δ_i reaches its minimum value which is $D_{r,i}$.

b) Δ_i is relatively larger for the initial frames of the GOP. Mathematically, the main reason is the larger weights (Γ_i) for those frames. It can be explained intuitively also –any mismatch in the 5th frame propagates to the next 25 frames of the GOP (with GOP size of 30), while any mismatch in the 25th frame propagates to the next 5 frames only.

c) The variables QP_r are integers and cannot be varied continuously. Therefore, we cannot continuously change the normalized redundancy and hence Δ. This means that there are a limited number of feasible Δ-redundancy points, which is an important factor in deciding the optimization algorithm.

d) Δ_i is larger for *Mobile* than for *Foreman* sequence because *Mobile* video has more content.

After analyzing the behavior of Δ with variation in redundancy, we now explain our WMRS method. In this method, we do not solve any equation such as that given by (10); rather, we gradually increase the redundancy of the frames so that the average distortion is minimized. The redundancy, part by part, is allocated to the most effective point of the end-to-end distortion function. At first, we assign the

minimum acceptable redundancy for all frames; i.e., QP_r is set to its maximum value. At this point, since the redundancy budget is higher than this minimum redundancy, we can decrease QP_r by one for frame i. By doing this, we start utilizing one portion of the redundancy budget. Now, if the total consumed redundancy is still lower than the redundancy budget, then we can decrease QP_r by one again for frame i, or for another frame. The key point here is to select the best frame whose QP_r must be decreased in each step. Based on WMRS, the best candidate frame is the frame that, at the current QP_p and QP_r, has the steepest $\Delta_i - R_{r,i}$ slope. In other words, the frame whose $-\frac{\partial \Delta_i}{\partial R_{r,i}}$ is maximum is the best choice for QP_r decreasing or redundancy allocation. Using this policy for selecting the target frame of the redundancy allocation, the allocation continues until the whole redundancy budget is utilized. By derivation from (13), $-\frac{\partial \Delta}{\partial R_r}$ is obtained as follows:

$$-\frac{\partial \Delta_i}{\partial R_{r,i}} = -\frac{\partial D_{r,i}}{\partial R_{r,i}} - \Gamma_i \frac{\partial D_{mismatch,i}}{\partial R_{r,i}} \tag{16}$$

In Fig. 3, we show the slopes order for the first three frames of *Mobile*, with GOP size of 4 frames. This small GOP size is chosen so that the difference of the slopes can be visible. When GOP = 30, the frames' Γ_i are very close to each other, and hence, Δ_i curves for sequential frames are too similar to be easily distinguishable in the figure. The order of redundancy allocation is determined by the order of the slopes. We decrease QP_r's in this order until the whole redundancy budget is utilized.

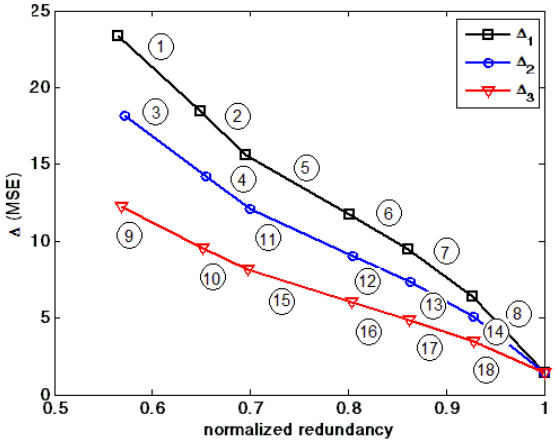


Fig.3. The order of redundancy allocation in WMRS

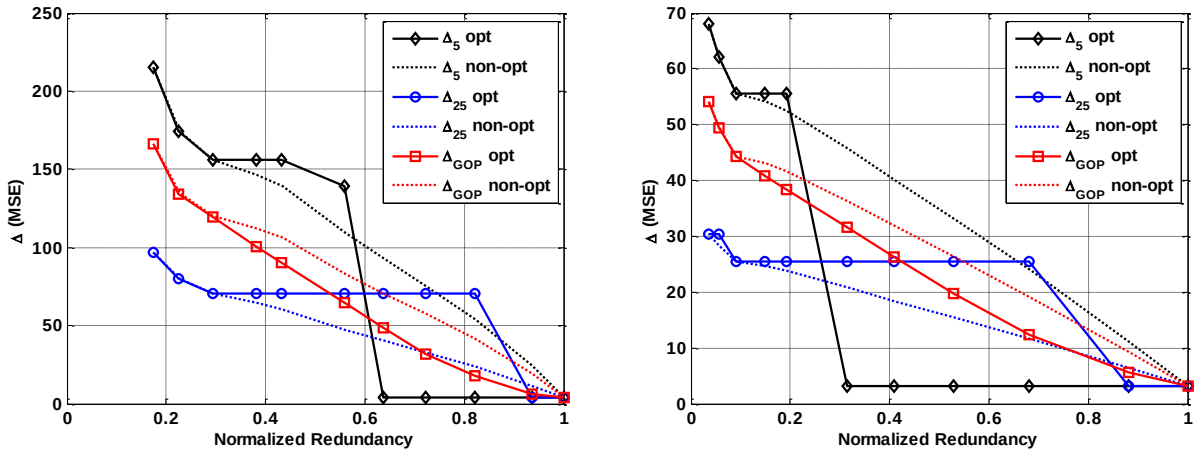
The pseudocode for the WMRS method is given in *Algorithm1*.

Algorithm 1: WMRS method	
Step 1:	Set initial point as $QP_{r,1} = QP_{r_{max}}$, $QP_{r,2} = QP_{r_{max}}$, $QP_{r,3} = QP_{r_{max}}$, … $QP_{r,N} = QP_{r_{max}}$
Step 2:	Calculate the slopes , $-\frac{\partial \Delta}{\partial R_r}$, for all frames at the current $\boldsymbol{QP_r}$
Step 3	find the frame j which has maximum slope $\left(j = \arg\max\left\{ -\frac{\partial \Delta_1}{\partial R_{r,1}} , -\frac{\partial \Delta_2}{\partial R_{r,2}} , -\frac{\partial \Delta_3}{\partial R_{r,3}} \dots \right\} \right)$
Step 4:	Set $QP_{r,j} = QP_{r,j} - 1$
Step 5:	Calculate the utilized redundancy up to now, R_r
Step 6:	If ($R_r < R_{r,t}$) then calculate $-\frac{\partial \Delta_j}{\partial R_{r,j}}$ at the new $QP_{r,j}$ and goto step 3 Else If ($R_r > R_{r,t}$) then $QP_{r,j} = QP_{r,j} + 1$ and goto step 7 Else goto step7
Step 7:	End

As can be seen from *Algorithm 1*, QP_r of the best frame is decreased by one only, even if we had enough redundancy budget to spare. The reason is that the slopes in Fig. 3 are not constant and vary with redundancy. Therefore, for each value of redundancy (i.e. updated $\boldsymbol{QP_r}$), we are searching again for the frame with maximum $-\frac{\partial \Delta}{\partial R_r}$. Actually, in WMRS method, the redundancy is allocated part by part with a sequence of locally optimum decisions. It is noteworthy that when $QP_{r,j}$ changes, only Δ_j is affected and must be recomputed; and the other slopes remain valid.

We calculated the optimal $\boldsymbol{QP_r}$ and Δ using the proposed WMRS algorithm for two test video sequences (*Mobile* and *Foreman*) and present the results in Fig. 4. In order to show the WMRS optimization performance, the same settings like those of Fig. 2 ($QP_p = 20$ and $QP_{r_{max}} = QP_p + 10$) are used in this experiment. For ease of comparison, the corresponding curves (for non-optimal $\boldsymbol{QP_r}$) from Fig. 2 are plotted in this figure with dotted lines.

From Fig. 4, we can see a wide range of redundancy at which optimum Δ_5 is smaller than the non-optimum Δ_5. On the other hand, it can be seen that Δ_{25} for optimal $\boldsymbol{QP_r}$ is not smaller than that for non-optimum parameters except for very high redundancies. However, this actually does not matter because the aim of optimal allocation is not to minimize each frame's Δ individually; but to minimize Δ_{GOP} (equation (12)), and as per figures 4(a) and 4(b), it is lower for optimal case in all redundancies.



(a) (b)

Fig. 4. Variation of Δ_i when optimal $\boldsymbol{QP_r}$ is applied for (a) *Mobile* and (b) *Foreman* sequences

Our proposed method follows a pattern similar to Greedy algorithm [26][27]. In Greedy algorithms, the global solution of an optimization problem is approximated by a solution achieved by a sequence of locally optimum decisions. Advantages of Greedy algorithms are their low complexity and ease of implementation without convergence problem. Examples of applications using Greedy algorithms can be found in [28]-[33]. However, the output of these algorithms may not be the global optimum point of the objective functions, and they are known as approximations methods. In particular, for our specific application, adding an amount of redundancy based on WMRS is a locally optimum strategy, and the output may not be the optimum solution. However, we have shown in the appendix that for the convex problems, WMRS method gives the exact optimum points. We have also shown there that our redundancy allocation problem has a small deviation from a convex problem.

Finally, as per equation (16), $-\frac{\partial \Delta_i}{\partial R_{r,i}}$ which is the decision criterion in our method, has two terms. However, except for the last few frames of the GOP, the second term (i.e. the weighted mismatch) is the dominant term. Hence we used weighted mismatch-rate slope in our nomination, even though the first term is not ignored in our calculations.

B. The optimal value for redundancy budget

In subsection *A*, we explained how the given redundancy budget is optimally divided among the frames.

In this subsection, we discuss how to find the optimal amount of redundancy budget at the given PLR.

As mentioned earlier, with more amount of redundancy, QP_r's can be closer to QP_p's resulting in more similar descriptions, lower mismatch and less error propagation. On the other hand, the higher amount of redundancy leads to lower primary rate budget at the given bandwidth, and hence higher primary (central) distortion will be incurred. Therefore, the optimal value of the redundancy depends on the impact factors of error propagation and primary distortion in the end-to-end distortion expressed in (8). It is evident from (8) that for low enough PLR, primary distortion becomes dominant and hence the optimizer must allocate more rate to the primary rate. In contrast, for high PLRs the redundancy rate should be increased.

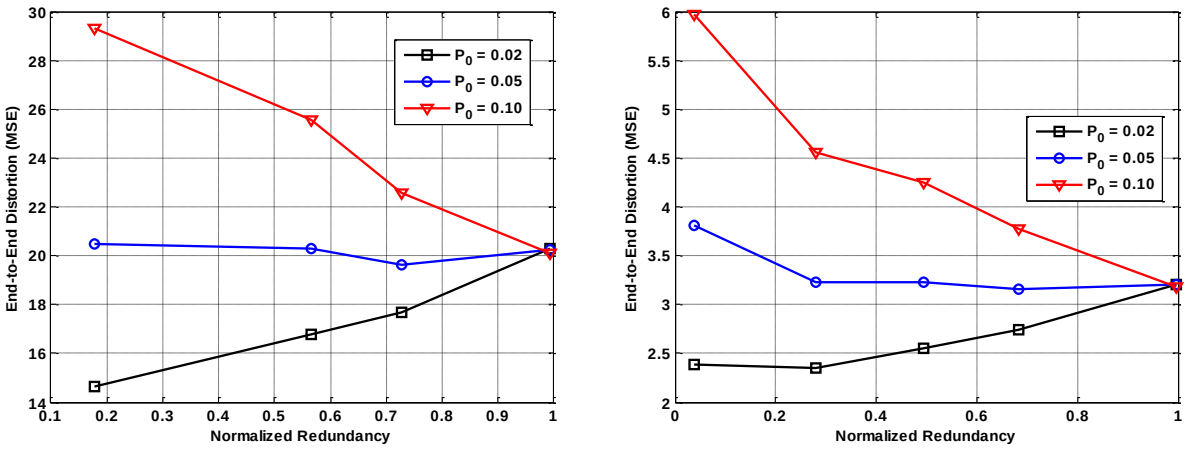
Now, to calculate the primary distortion, we need to have the primary quantization parameter, $\boldsymbol{QP_p}$. In order to determine $\boldsymbol{QP_p}$, we follow the algorithm presented in [14]. If $QP_{p,1}$ is the quantization parameter for the first frame, then $QP_{p,i}$ of the i^{th} frame of the GOP is obtained by solving the following equation:

$$QP_{p,i} = QP_{p,1} + 3\ \log_2 \frac{\psi_1}{\psi_i} \tag{17}$$

where ψ_i is the temporal propagation factor for the primary slices and its definition and derivation can be found in [14]. Therefore, QP of the primary coefficients can be obtained independent of the coding parameters of the redundant coefficients. We solved these equations for our test videos with the GOP size of 30 and 60 frames. We changed PLRs from 0.02 to 0.15, and it is observed that almost all QPs are the same and equal to $QP_{p,1}$. In exceptional cases, for the last two or three frames of the GOP, the resulting QP might be $QP_{p,1} + 1$. Therefore, with a good approximation, all frames can have the same QP_p. Now, the problem is to find the optimum value for $QP_{p,1}$ which has a closer relation to the optimal redundancy budget.

If $QP_{p,1}$ is known, we can find $\boldsymbol{QP_p}$ using (17), and therefore the primary rate, $R_{p,t}$, is directly determined. Then, at the given total rate, the redundancy rate budget is obtained by calculating $R_{r,t} = R_t - R_{p,t}$ and this amount of redundancy can be optimally allocated to the frames using WMRS. We now solve the optimization problem for different values of $QP_{p,1}$ and study the behavior of end-to-end distortion. In

Fig. 5, we show the end-to-end distortion curves versus normalized redundancy for *Mobile* and *Foreman* video sequences where R_t = 4 Mb/s and PLR = [0.02 , 0.05 , 0.10] and GOP size = 30. We can now find the optimum value for the redundancy budget, the point at which the minimum end-to-end distortion is achieved.



(a) (b)

Fig. 5. End-to-end distortion with R_t = 4 Mb/s and various PLRs for (a) *Mobile* and (b) *Foreman* sequences

From Fig. 5 we see that for PLR = 0.02, the end-to-end distortion increases with the normalized redundancy. This means that for this value of PLR, the minimum distortion is achieved at minimum redundancy. On the other hand, for PLR = 0.10, the minimum distortion is obtained at maximum redundancy. Therefore, the optimum strategy is to have same values for primary and redundancy rates which results in completely similar descriptions. For PLR = 0.05, the curve is not monotonic and the best point is something in between minimum and maximum possible redundancies.

In order to find the minimal point of the curves in Fig. 5, we use the procedure described in *Algorithm 2*. Starting with an initial value, we first find the smallest $QP_{p,1}$ that provides the minimum redundancy. The redundancy budget is then allocated to the frames using WMRS. With larger $QP_{p,1}$, we have more redundancy rate budget and move to the right hand side of the curves in Fig. 5. As long as the end-to-end distortion keeps decreasing, we can enlarge $QP_{p,1}$. The first $QP_{p,1}$ that leads to the higher end-to-end distortion terminates the iteration. Since the problem has one minimum, and we start from the minimum

acceptable redundancy, the first increase in the end-to-end distortion means that we just left the global minimum of the problem, and thereby, we stop the iteration at this point. This way, for a given PLR and a video content, the optimal redundancy budge is obtained.

Algorithm 2: Optimal end-to-end distortion at different redundancy budgets and constant total rate

Step 1:	Set $QP_{p,1} = QP_{p0}$, $\boldsymbol{QP_{r-init}} = \boldsymbol{QP_{r-max}}$, $D_{e2e-min} = MAX\_INTEGER$
Step 2:	Obtain $\boldsymbol{QP_p}$ for frames of the GOP using (17)
Step 3:	Obtain $R_{p,t}(\boldsymbol{QP_p})$ and then $R_{r,t} = R_t - R_{p,t}$
Step 4:	If ($R_{r,t} < R_{r,min}$) then $QP_{p,1} = QP_{p,1} + 1$ and goto Step 2 (in order to have the minimum redundancy)
Step 5:	If ($R_{r,t} > R_{p,t}$) then goto Step 11 (the redundancy rate must not be more than primary rate)
Step 6:	Find $\boldsymbol{QP_{r-tmp}}$ by the procedure described in *Algorithm 1* (WMRS)
Step 7:	Calculate D_{e2e} using (8)
Step 8:	If ($D_{e2e} \geq D_{e2e-min}$) then goto Step 11
Step 9:	If ($D_{e2e} < D_{e2e-min}$) then $D_{e2e-min} = D_{e2e}$, $\boldsymbol{QP_{r-opt}} = \boldsymbol{QP_{r-tmp}}$
Step 10:	$QP_{p,1} = QP_{p,1} + 1$ and goto Step 2
Step 11:	End

IV. BITRATE TUNABILITY

The basic optimization problem for the MDC under study is the equation given by (9); i.e., $\boldsymbol{QP_r}$ and $\boldsymbol{QP_p}$ vectors along with the rate constraint must be solved simultaneously. However, this fact is not considered in the solutions proposed in [13] and [14]. In particular, in [14], $\boldsymbol{QP_p}$ is obtained solving equation (17) and $\boldsymbol{QP_r}$ using the following equation:

$$QP_{r,i} = QP_{p,1} + 3\log_2\left(\frac{1}{P_0(1-P_0)}\frac{\psi_1}{\phi_i}\right) \quad i = 1 \ldots N \tag{18}$$

And the following two equations are used in [13] in order to find $\boldsymbol{QP_r}$:

$$QP_{r,1} = QP_p - 3\log_2(p\phi_1) \tag{19}$$

$$QP_{r,i} = QP_{r,1} + 3\log_2\left(\frac{\phi_1}{\phi_i}\right) \quad i = 1 \ldots N \tag{20}$$

The definition of ϕ_i in these two papers is not identical. The only parameter that can be used to control the total coding rate and satisfy $\sum_{i=1}^{N}[R_{p,i} + R_{r,i}] \leq R_t$ is $QP_{p,1}$ in [14] and QP_p in [13]. When this

parameter has minimum variation, all elements of $\boldsymbol{QP_r}$ vectors vary accordingly, and therefore the change in redundant coding rates is significant. As a result, these algorithms fail to adjust the rate precisely. On the other hand, in our method, even though $\boldsymbol{QP_p}$ and $\boldsymbol{QP_r}$ are found independently, each elements of $\boldsymbol{QP_r}$ is tuned considering the bitrate budget as well as the end-to-end distortion. As long as the redundancy rate budget (and hence the rate budget) is not exhausted, we can still find the maximum slope and decrease one QP_r such that the optimality of the distortion is retained while not exceeding the bitrate budget. To show this fact with illustration, we solve the redundancy allocation problem using WMRS and the algorithms proposed in [13] and [14] for various rate constraints and two PLRs. The bitrates for [13] and [14] are tuned by *Algorithm 3*. Starting from minimum QP_{p0} guarantees that the first quantity of QP_{p0} that meets the rate constraint is the optimum point.

Algorithm 3: To adjust the total bitrate in [13] and [14]

Step 1:	Set $QP_{p0} = QP_{min}$
Step 2:	for [13] : set $QP_p = QP_{p0}$, $\boldsymbol{QP_p} = [QP_p]$, obtain $QP_{r,1}$ using equation (19) for [14]: set $QP_{p,1} = QP_{p0}$ and obtain $\boldsymbol{QP_p}$ using (17)
Step 3:	for [13] : Obtain $QP_{r,i}$ $(i = 1 \dots N)$ using equation (20) for [14]: Obtain $QP_{r,i}$ $(i = 1 \dots N)$ using equation (18)
Step 4:	Find the total rate $R_{t-tmp} = rate\ of\ primary\ slices + rate\ of\ redundant\ slices$
Step 5:	If $R_{t-tmp} \leq R_t$ then save $QP_{r,i}$s as optimal $\boldsymbol{QP_r}$ and goto Step 7
Step 6:	If $R_{t-tmp} > R_t$ then $QP_{p0} = QP_{p0} + 1$ and goto Step 2
Step 7:	End

The vertical axis of Fig. 6 is the percentage of the given bandwidth that can be optimally utilized by the three methods. As we can see from this figure, the bitrate constraint is perfectly utilized by WMRS. However, for the other two methods, depending on the bitrate, the bandwidth usage sometimes reaches 65%. As we described earlier, this is due to this fact that in [13] and [14] there is only one discrete variable for tuning the rate of a GOP, and therefore precise bitrate adjustment is not possible.

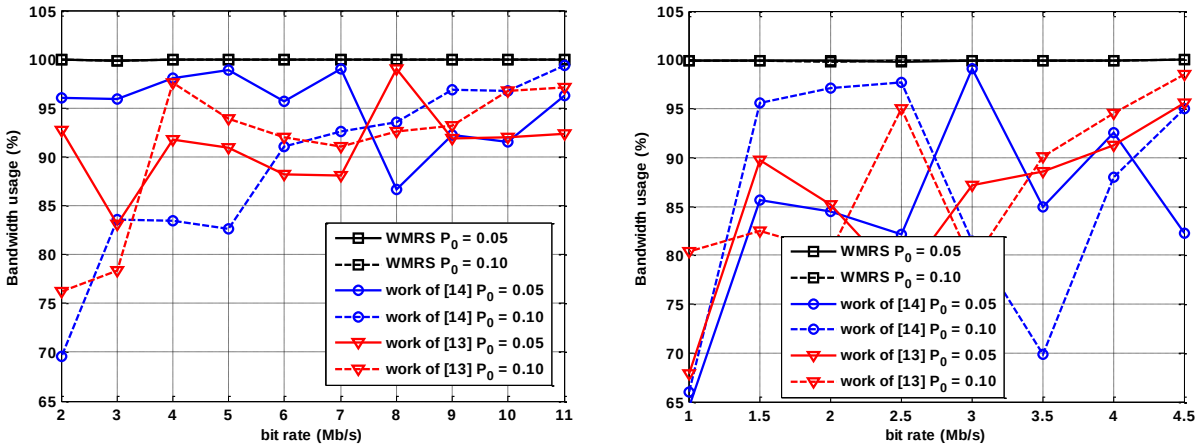


Fig. 6. Bandwidth usage in three methods for (a) *Mobile* and (b) *Foreman* sequences

V. COMPLEXITY AND DELAY ISSUES

One of the important issues for any optimization algorithm is its computational complexity. In WMRS for a GOP of 30 frames and $QP_{r_{max}} = QP_p + 15$, if we assume that $\boldsymbol{QP_p}$ is known, we need to obtain an array with the size of at least 30 slopes (if lowest redundancy is needed) and at most 30*15 = 450 slopes (if highest redundancy is needed). For computing these slopes, we need to calculate the redundant and mismatch distortions as well as the rates along with some divisions. Details of the procedure are given below.

We assume that DCT coefficients (residual signals after DCT transform) follow the Laplacian distribution [25]:

$$f(x) = \frac{\eta}{2} e^{-\eta\ |x|} \tag{21}$$

where η is the Laplacian distribution parameter. For each frame, the distribution parameters of the coefficients are obtained, and then the quantization distortion ($D_{r,i}$) is calculated using the equation given by (22) [36]:

$$D_{r,i} = \frac{\eta Q_{ss} e^{f\eta Q_{ss}}(2 + \eta Q_{ss} - 2f\eta Q_{ss}) + 2 - 2e^{\eta Q_{ss}}}{\eta^2(1 - e^{\eta Q_{ss}})} \tag{22}$$

where Q_{ss} is the quantization step size (it is simply obtained from QP), and f is the quantization rounding offset. We calculate the mismatch distortion ($D_{mismatch,i}$) by (23) [14]:

$$D_{mismatch,i} = \int_{-\infty}^{\infty} \left[Q^{-1}\left(Q\left(x_i, QP_{r,i}\right), QP_{r,i}\right) - Q^{-1}\left(Q\left(x_i, QP_{p,i}\right), QP_{p,i}\right)\right]^2 f(x_i)dx_i \quad (23)$$

where x_i is the DCT coefficient of frame i in a specific frequency and $f(x_i)$ is its Laplacian distribution. $Q(.)$ and $Q^{-1}(.)$ represent the quantization and inverse quantization operations respectively, both are functions of the used QP. This integration is calculated and saved as a look up table (similar to [14]). We also used Entropy function for rate calculation (Entropy rate is closely matched with the Context-Adaptive Binary Arithmetic Coding (CABAC) output of the encoder [37]). Eventually, to calculate the partial derivations, we use the following differentiations:

$$\frac{\partial D_{r,i}}{\partial R_{r,i}} \approx \frac{D_{r,i}\left(QP_{r,i}\right) - D_{r,i}(QP_{r,i}+1)}{R_r\left(QP_{r,i}\right) - R_r(QP_{r,i}+1)} \quad (24)$$

$$\frac{\partial D_{mismatch,i}}{\partial R_{r,i}} \approx \frac{D_{mismatch,i}\left(QP_p,\ QP_{r,i}\right) - D_{mismatch,i}(QP_p,\ QP_{r,i}+1)}{R_r\left(QP_{r,i}\right) - R_r(QP_{r,i}+1)} \quad (25)$$

Having the above values, we can calculate the slopes using equation (16).

As we previously mentioned, [13] and [14] are simplified and fast algorithms based on Lagrangian approach. For the given $\boldsymbol{QP_p}$, equations (20) and (18) must be solved in [13] and [14] respectively. Equation (20) is very straightforward because there is no unknown variable at the right hand side of the equation. In equation (18), ϕ_i is a function of $QP_{r,i}$, and cannot be solved analytically. Therefore a numerical method is needed and we used Bisection method for this purpose.

However, the above mentioned computations are for the case that $\boldsymbol{QP_p}$ is known. Actually, instead of $\boldsymbol{QP_p}$, the total bitrate budget is given and we have to find optimal $\boldsymbol{QP_p}$ and $\boldsymbol{QP_r}$ satisfying the bitrate constraint. For this reason, we executed *Algorithm 2* for our method and *Algorithm 3* for [13] and [14] in order to compare the complexity. These algorithms are written with the same approach and the same

iterations step-size.

In Table 1, we present the processing time profile of the three works, based on our test machine: Core i5 , 2.2 GHz CPU and 4 GB of RAM. For all these algorithms, the time counter starts when QP_p satisfies the minimum redundancy, and terminates when optimal $\boldsymbol{QP_p}$ and $\boldsymbol{QP_r}$ meet the total rate constraint. The processing time needed to solve (17) is excluded here because this equation finds the variation of QP_p among the frames of the GOP, which is not related to the redundancy allocation. PLR changes the redundancy rate, and bitrate is the constraint of the problem, therefore the processing time is PLR and bitrate dependent, since they affect how many times *Algorithm 2* or *Algorithm 3* must be executed. In this table, the maximum times for each case are reported. In Table 1, we also added the processing time to solve (9) using the SQP technique. In SQP, a quadratic approximation of the problem is iteratively solved, and then the approximation is updated and solved again. This procedure continues until the global minimum is obtained. The advantage of this technique is converting the nonlinear problem into a quadratic problem which is relatively easier to solve. We utilized the built-in "fmincon" function in MATLAB and selected SQP as the solver algorithm. We set the tolerance and iteration step-size large enough so as not to affect the optimal results. During this experiment, we found that the results are almost similar to those of WMRS but the computation times are significantly longer. It is noteworthy that the MATLAB functions are general purpose which can be made faster if it is rewritten in C language and as an application specific program; however, as described in [35], SQP inherently has extensive computations.

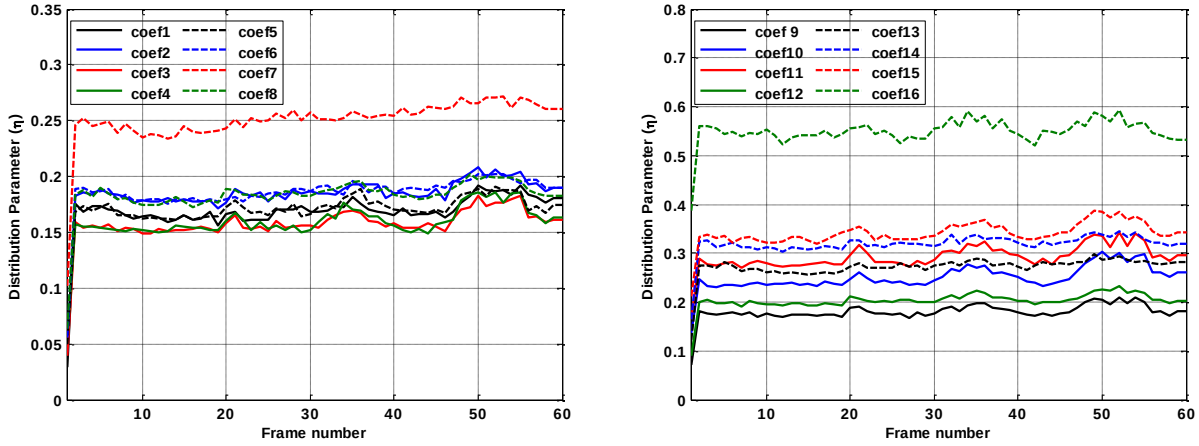
Table 1. Processing time (in second) per frame for WMRS method and two other works

	Mobile		Foreman		Flower		Bus	
	GOP = 30	GOP = 60	GOP = 30	GOP = 60	GOP = 30	GOP = 60	GOP = 30	GOP = 60
WMRS	0.102	0.103	0.107	0.108	0.096	0.116	0.130	0.097
[14]	0.138	0.143	0.088	0.109	0.142	0.127	0.147	0.145
[13]	0.002	0.002	0.002	0.001	0.002	0.002	0.002	0.002
MATLAB-SQP	13.942	27.071	13.059	26.229	12.977	26.222	13.744	27.373

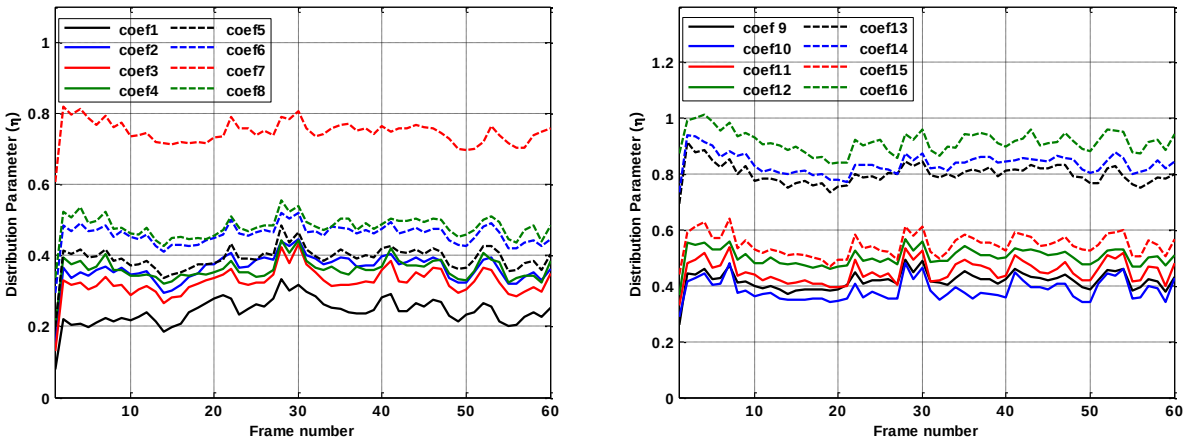
As can be seen in Table 1, [13] has the fastest performance, and WMRS (with the exception of *Foreman* and GOP=30) stands in second place. Moreover, complexity of [13],[14] and WMRS is significantly less than the computations needed to solve equation (9) using SQP. The reason is that finding $\boldsymbol{QP_p}$ and $\boldsymbol{QP_r}$ in [13],[14] (and in our approach) are decoupled and the vector variables are reduced to scalar variables, as can be seen in (17)-(20). However, this assumption affects the performance of [13] and [14] in terms of bitrate tunability and rate-distortion performance as we discussed in Section IV and VI, respectively.

We now discuss the delay issue. As per equation (9), in order to minimize the error propagation, we need to have DCT coefficients distribution of all frames in the GOP. Therefore, the encoder must wait for buffering of all frames from the GOP and then run the optimization process. This delay is natural and it is common in all GOP-wise optimization algorithms, e.g.[8][12]-[14]. The commonly used solution is to predict the distribution parameters of the DCT coefficients of the future frames [14][15]. One safe assumption is that the next frames will have coefficients distribution similar to the ones presently under processing. For example, if we are encoding the 3rd frame, then we assume that 4th , 5th , 6th and other frames of the GOP will have the distribution parameters like those of the 3rd frame. We gave a detailed explanation of this procedure in our previous work [15].

The above-mentioned assumption is effective because the distribution of residual signals do not change suddenly, due to the various motion estimation block-sizes and sub-pixel estimation techniques in H.264/AVC. Therefore, as long as the background of the sequence is steady, the coefficients distribution changes are smooth and our assumption remains valid. As mentioned, the DCT coefficients distribution can be modeled by Laplacian function of (21). In Fig. 7, we show the variation of η for all 16 output coefficients of the 4x4 DCT transform for the first 60 frames of *Mobile* and *Foreman* video sequences with $QP = 20$. As expected, η is usually larger for higher frequencies; however, for a specific frequency, its changes through the sequential frames are smooth. The first frame is an I-frame and therefore has the lowest η in all frequencies because all blocks are coded in intra mode in this frame.



(a)

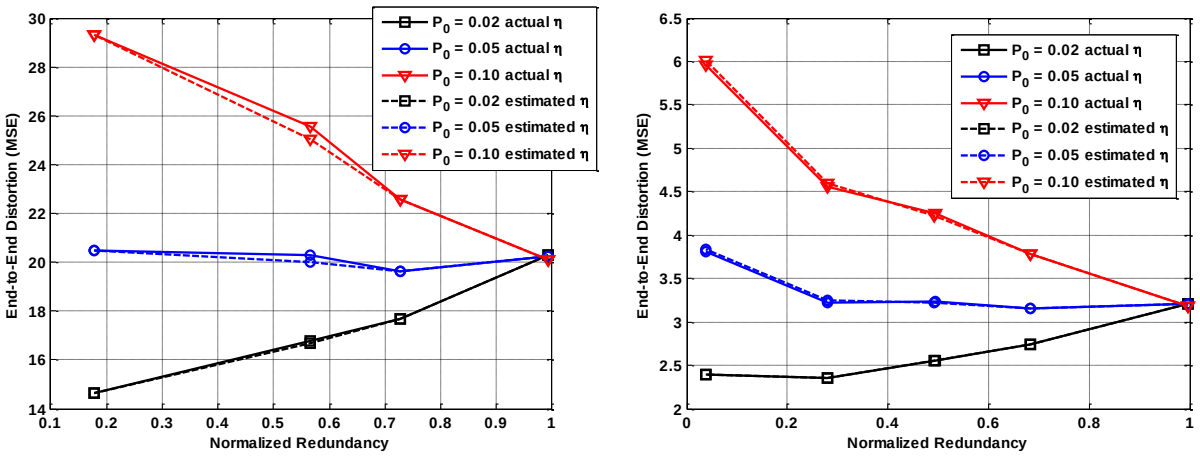


(b)

Fig. 7. Variation of Laplacian distribution parameter (η) for frames of (a) *Mobile* and (b) *Foreman* sequences

In Fig. 8, we show the optimal end-to-end distortion with the parameters corresponding to those of Fig. 5 but using η estimation discussed above. It can be seen that even without using the actual η, the optimal end-to-end distortion and hence the optimal $\boldsymbol{QP_r}$ do not change significantly. However, since the data distribution parameters are updated after each frame encoding, WMRS allocation must be recomputed on a frame-by-frame basis for each GOP. In other words, when encoding of frame i is finished, the distribution parameters are updated and WMRS is run again for the remaining frames. Therefore, this data extension policy triggers an increase in the computational complexity with an order of $GOP/2$ times. However, this increase in processing time is common for WMRS, [13] and [14]. But, the encoding time in our system; e.g. for *Mobile* CIF sequence (with all possible block-sizes available, sub-pixel and fast full

search algorithm for motion estimation, Rate Distortion Optimization (RDO) enabled, $QP = 16$ and without rate control) is about 15 seconds for P-frames, and 30 seconds for B-frames. Therefore, the additional time needed for optimization is still much less than the encoding time.



(a) (b)

Fig. 8. End-to-End distortion with WMRS redundancy allocation, when actual η (solid lines) and estimated η (dashed lines) are used, for (a) *Mobile* and (b) *Foreman* sequences

VI. EXPERIMENTAL RESULTS

In order to show the performance of the proposed algorithm, we implemented the optimized MDC scheme in the H.264/AVC reference software, JM 16.0 [38]. In the encoder, CABAC is used for the entropy encoder and rate control is off. Without loss of generality, we intentionally skip the B-frames in the GOP structure, because B-frames produce delay and they are not used in delay sensitive applications. In our testbed, each video packet contains an integer number of slices and is smaller than the Maximum Transmission Unit (MTU) of the network in order to avoid fragmentation. For the typical MTU size of 1500 bytes, the payload size is set to be 1460 bytes where 40 bytes are reserved for RTP/UDP/IP header information. For each value of PLR, 20 random packet loss patterns with Bernoulli distribution are generated and applied on each channel. The channels are assumed to be independent.

We carried out the optimization procedure described in *Algorithm 2* for four values of PLR and two GOP sizes, and applied the optimal QP_ps and QP_rs during the MDC encoding of the test videos. By measuring the average video quality at the receiver's side, the RD performance of the MDC optimized by

WMRS is investigated. In order to compare the performance of our method, we also implemented the algorithms presented in [13] and [14] as reference methods. As we discussed in Section IV, their performance is affected by the coding rate constraint, and therefore in order to get the best performance out of those approaches, we did not apply the limitation discussed in that section. We only set several suitable values for QP_{p0} and then find the optimal $\boldsymbol{QP_p}$ and $\boldsymbol{QP_r}$ using equations (17)-(20) which are then applied when MDC encoding.

We used the *Foreman*, *Mobile*, *Flower* and *Bus* video sequences with CIF size and *FourPeople* and *Vidyo1* with HD (1280x720) size for our experiments. The average PSNR for luminance component versus total rate of both descriptions, for four values of PLR, are shown in figures 9-14.

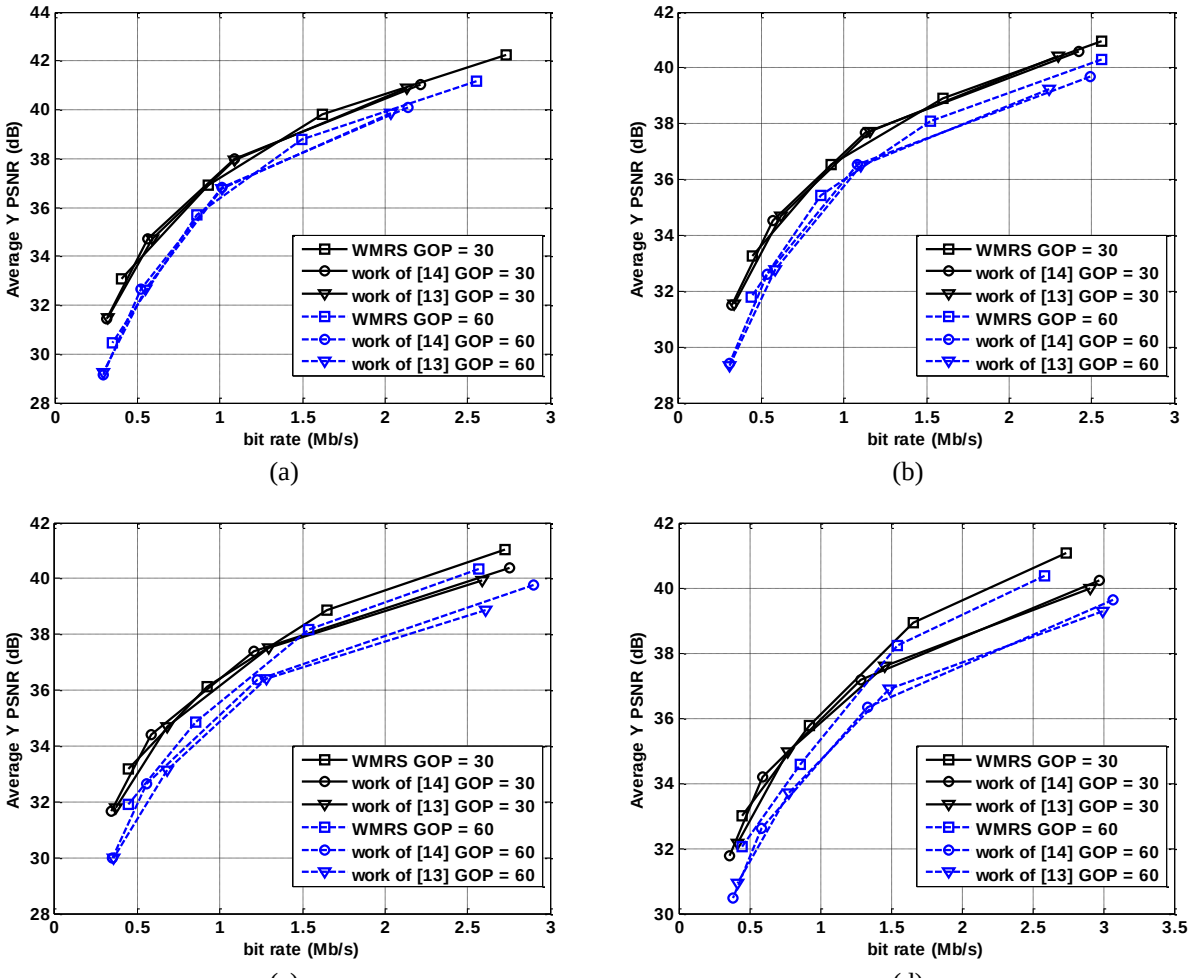


Fig. 9. Rate-Quality performance comparison for *Foreman* sequence for (a) PLR = 0.02 , (b) PLR = 0.05 , (c) PLR = 0.10 and (d) PLR = 0.15

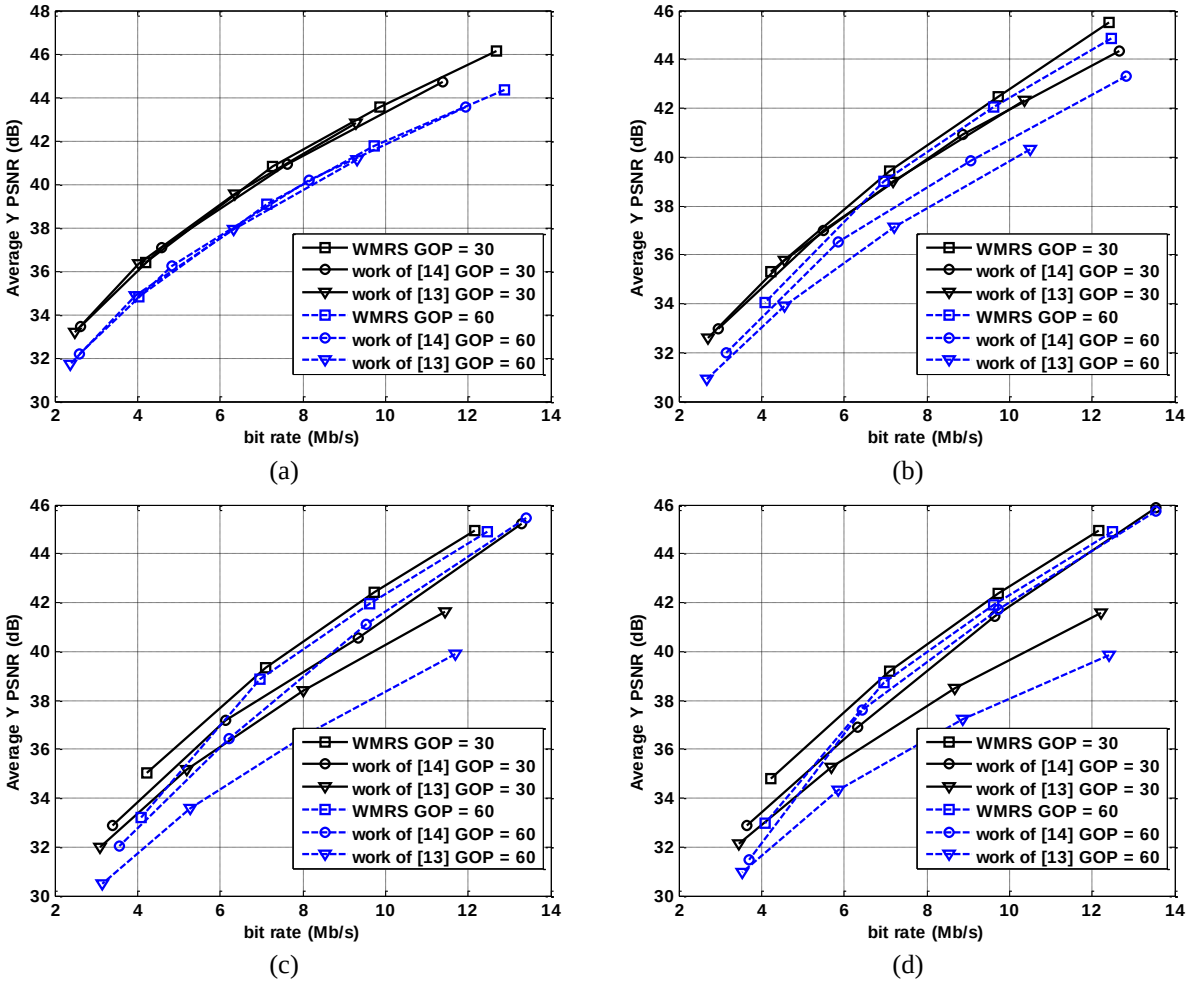


Fig. 10. Rate-Quality performance comparison for *Mobile* sequence for (a) PLR = 0.02 , (b) PLR = 0.05 , (c) PLR = 0.10 and (d) PLR = 0.15

From figures 9-14, we can see that the rate-distortion performance of the WMRS algorithm is better than the schemes presented in [13] and [14], especially for high PLRs and high rates. This is true for both GOP sizes. At low rates, due to the strong quantization, the sensitivity of Δ_i to redundancy is lower. As can be seen from figures 16.a and 16.b, when normalized redundancy varies from 0.1 to 0.8 for frame 10 and $QP_p = 16$ (high rate case), Δ_i changes from 250 to 40, i.e. it becomes about 6 times smaller. However, this variation is from 500 to 150, i.e. 3.3 times smaller when $QP_p = 32$ (lower rate case). Besides the lower dynamic range of Δ_i , its contribution in end-to-end distortion is lower at low rates because quantization distortions are larger. Therefore, optimality of redundancy allocation becomes less effective, and as a result we can see that the three methods have close performance at low rates.

Note that the curves of figures 9-14 compare the average PSNR taken over a GOP (30 or 60 frames). For the first frames of the GOP, the difference of the algorithms is not too much, but going forward to the end of the GOP, the difference becomes more distinguishable. This is due to the error propagation that is larger for last frames of the GOP. Actually the difference for the last frame of the GOP might be as large as about 3 dB in our experiments. In Fig. 15, we show the MDC decoder outputs for the 60th frame of *Foreman* sequence for all three methods.

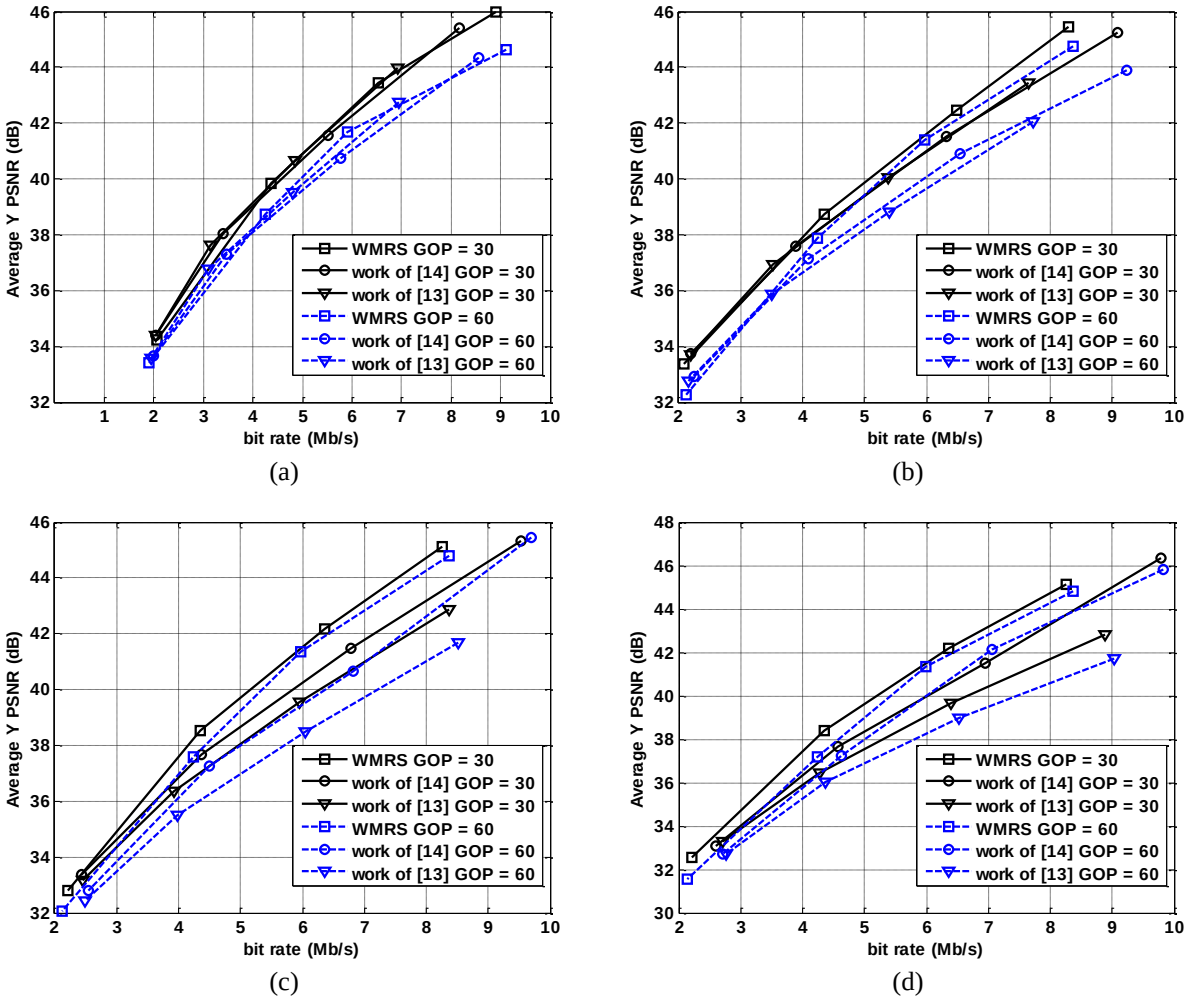
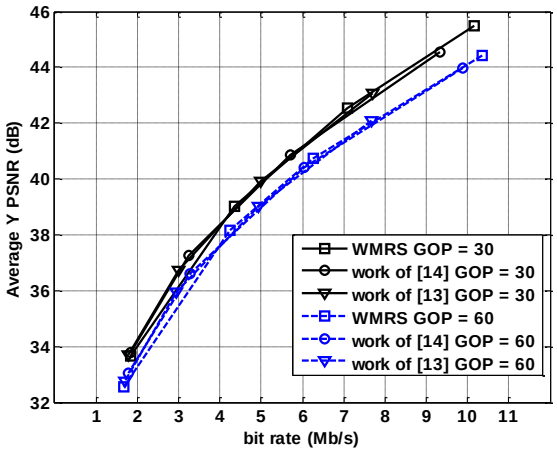


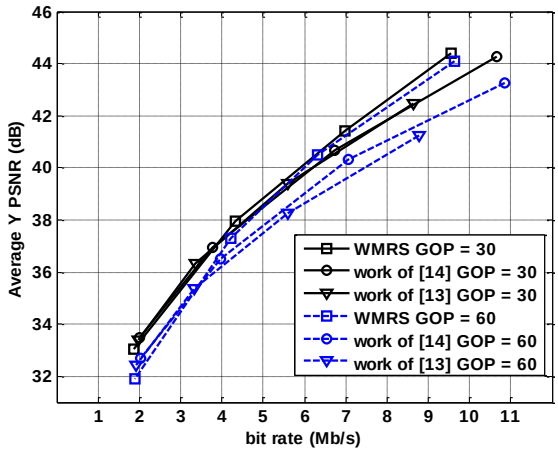
Fig. 11. Rate-Quality performance comparison for *Flower* sequence for (a) PLR = 0.02 , (b) PLR = 0.05 , (c) PLR = 0.10 and (d) PLR = 0.15

VII. CONCLUSION

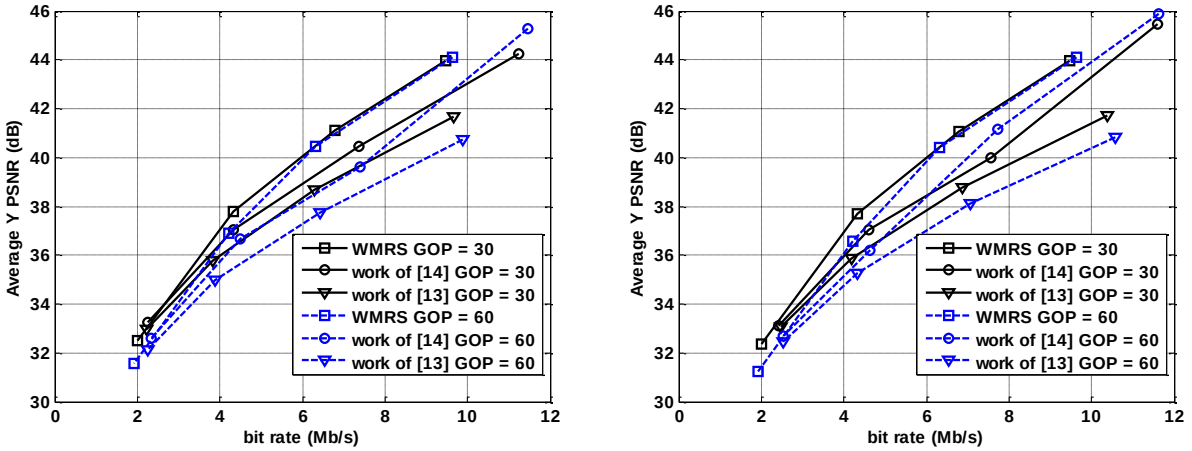
In this paper, we presented our new method WMRS for redundancy allocation in multiple description video coding. Our proposed method solves the optimal redundancy allocation problem in two phases: first, it allocates the given redundancy budget to the frames of the GOP and second, it determines the optimal value of the redundancy budget. In WMRS, a portion of the redundancy budget is allocated to the frame that currently has maximum weighted mismatch-rate slope. This procedure continues until the whole redundancy budget is consumed. Depending on the packet loss rate and the video content, the end-to-end distortion is minimized in a specific and optimal redundancy budget. Our method is naturally a low complexity optimization algorithm comparable with the fast algorithms proposed in [13] and [14]. Due to the step-wise redundancy allocation, the bitrate tunability of our method is nearly perfect, while this is not the case for [13] and [14]. Our experimental results reveal that our optimizer achieves better performance than its competitors, especially for higher packet loss rates and higher bitrates.



(a)

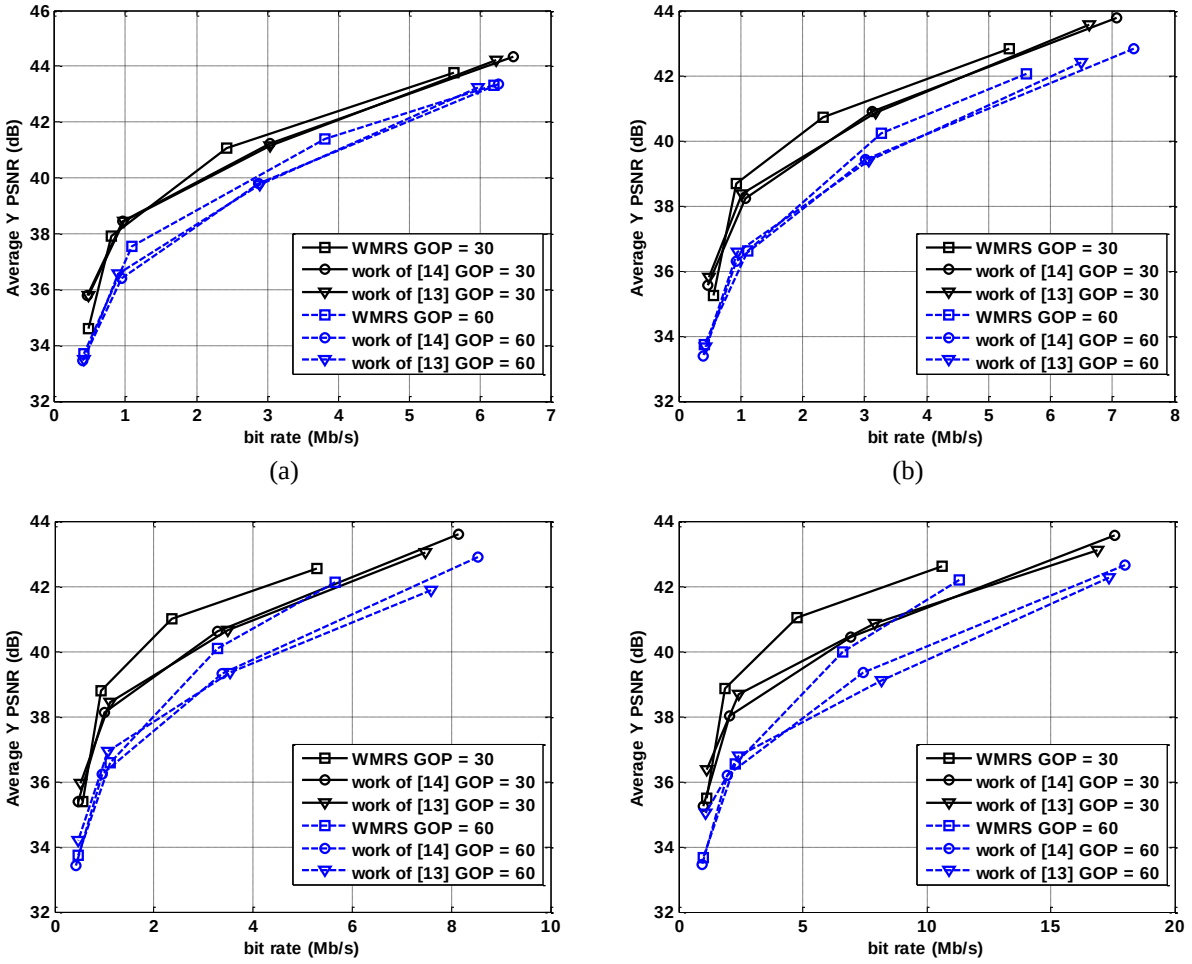


(b)



(c) (d)

Fig. 12. Rate-Quality performance comparison for *Bus* sequence for (a) PLR = 0.02 , (b) PLR = 0.05 , (c) PLR = 0.10 and (d) PLR = 0.15



(a) (b)

(c) (d)

Fig. 13. Rate-Quality performance comparison for *FourPeople* sequence for (a) PLR = 0.02 , (b) PLR = 0.05 , (c) PLR = 0.10 and (d) PLR = 0.15

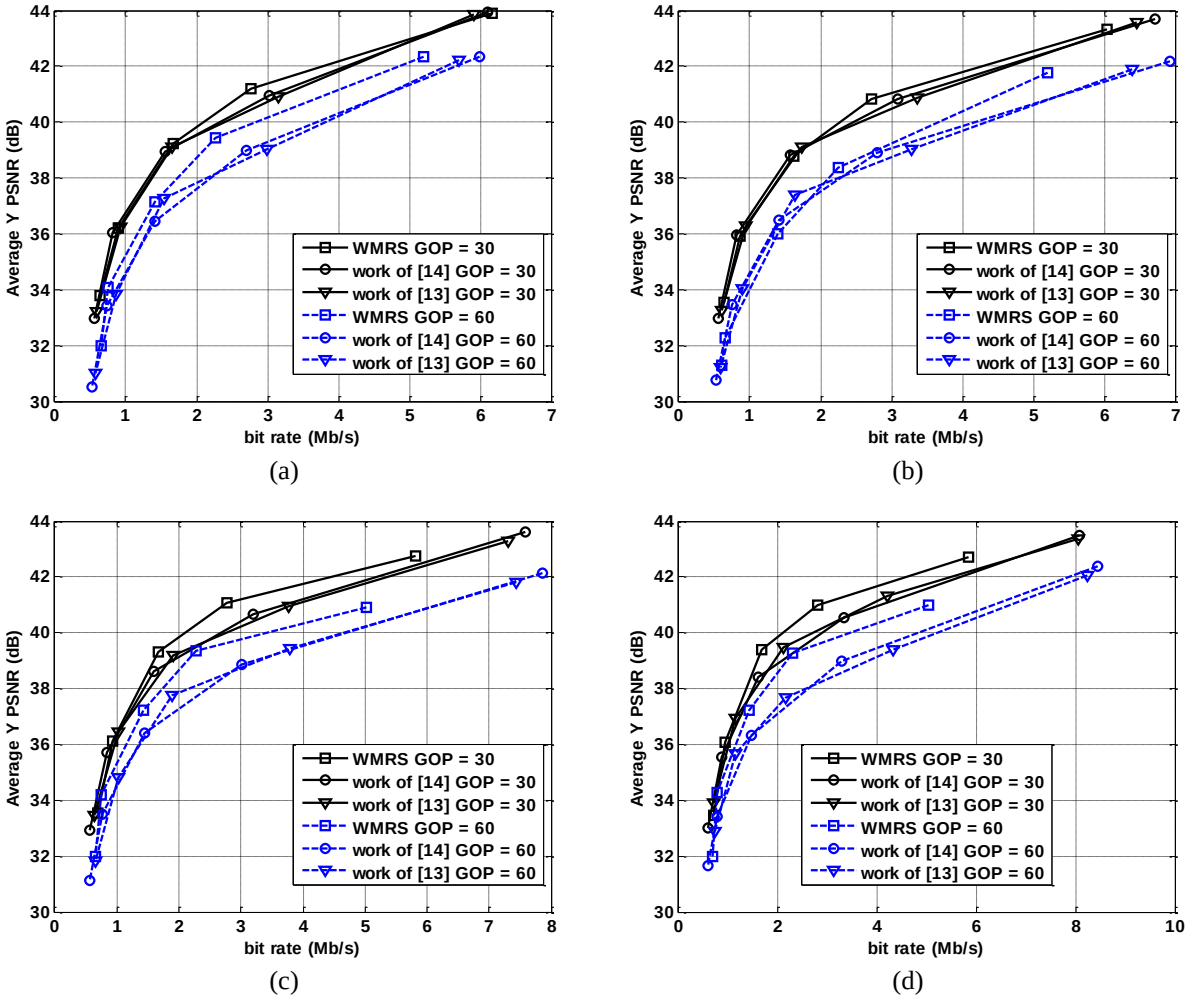


Fig. 14. Rate-Quality performance comparison for *Vidyo1* sequence for (a) PLR = 0.02 , (b) PLR = 0.05 , (c) PLR = 0.10 and (d) PLR = 0.15

(a) (b) (c)

Fig. 15. The receiver side reconstructed 60th frame of *Foreman* sequence, MDC coding rate is 1.5 Mb/s and PLR = 0.15,for (a) WMRS (b) work of [14] and (c) work of [13]

APPENDIX

In this appendix, we first prove that if Δ_i ($1 \le i \le N$) is a convex function of $R_{r,i}$, then the WMRS solution is exactly the optimum solution. The objective function of (12) can be rewritten in a simple form as follows:

$$\min_{\{\boldsymbol{QP_r}\}} \left\{ \sum_{i=1}^{N} \Delta_i \right\} \qquad s.t \sum_{i=1}^{N} R_{r,i} \le R_{r,t} \tag{26}$$

Assume that the global optimum redundancies are denoted by $R^*_{r,i}$ and those of WMRS output by $R^w_{r,i}$. Due to the discrete nature of the problem, we can write that:

$$\begin{aligned} R^*_{r,i} &= R_{i,min} + \Delta R_{i,1} + \Delta R_{i,2} + \cdots + \Delta R_{i,n_i^*} \\ R^w_{r,i} &= R_{i,min} + \Delta R_{i,1} + \Delta R_{i,2} + \cdots + \Delta R_{i,n_i^w} \end{aligned} \tag{27}$$

where $R_{i,min}$ is the redundancy achieved when QP_r is at its maximum value ($QP_r = QP_{r,max}$) , and $\Delta R_{i,1}$ is the increase in rate when we have one unit decrease in QP_r; i.e., $QP_r = QP_{r,max} - 1$. For $\Delta R_{i,k}$, the change in QP_r is from $QP_{r,max} - (k-1)$ to $QP_{r,max} - k$. Accordingly, we can write that:

$$\begin{aligned} \Delta_i^* &= \Delta_{i,max} - \left(S_{i,1}\Delta R_{i,1} + S_{i,2}\Delta R_{i,2} + \cdots + S_{i,n_i^*}\Delta R_{i,n_i^*} \right) \\ \Delta_i^w &= \Delta_{i,max} - \left(S_{i,1}\Delta R_{i,1} + S_{i,2}\Delta R_{i,2} + \cdots + S_{i,n_i^w}\Delta R_{i,n_i^w} \right) \end{aligned} \tag{28}$$

where $\Delta_{i,max}$ is the Δ_i at the minimum redundancy ($QP_r = QP_{r,max}$), and $S_{i,1}$ is the weighted mismatch-rate slope, i.e.:

$$S_{i,1} = -\frac{\partial \Delta_i}{\partial R_{r,i}} QP_r \ : \ QP_{r,max} \longrightarrow QP_{r,max} - 1 \tag{29}$$

The other $S_{i,k}$'s are defined in a similar manner. If the problem is convex (i.e., Δ_i is a convex function of $R_{r,i}$), we have a decreasing order in $S_{i,k}$'s , $1 \le k \le n_i$ for frame i. That is:

$$S_{i,1} \ge S_{i,2} \ge \cdots \ge S_{i,n_i} \tag{30}$$

Now, we assume that the solution of WMRS is not matched with the optimum solution, that is $n_i^* \neq n_i^w$. If $n_i^* > n_i^w$, due to the redundancy constraint of the problem, there exists (at least) one frame j for which $n_j^* < n_j^w$. Then we can write that:

$$\begin{aligned}\left(\Delta_i^* + \Delta_j^*\right) - \left(\Delta_i^w + \Delta_j^w\right) &= \left(\Delta_i^* - \Delta_i^w\right) + \left(\Delta_j^* - \Delta_j^w\right) \\ &= -\left(S_{i,n_i^w+1}\Delta R_{i,n_i^w+1} + S_{i,n_i^w+2}\Delta R_{i,n_i^w+2} + \ldots + S_{i,n_i^*}\Delta R_{i,n_i^*}\right) \\ &+ \left(S_{j,n_j^*+1}\Delta R_{j,n_j^*+1} + S_{j,n_j^*+2}\Delta R_{j,n_j^*+2} + \ldots + S_{j,n_j^w}\Delta R_{j,n_j^w}\right)\end{aligned} \tag{31}$$

Using the inequality of (30), we have:

$$\begin{aligned}S_{i,n_i^w+1}\Delta R_{i,n_i^w+1} + S_{i,n_i^w+2}\Delta R_{i,n_i^w+2} + \ldots + S_{i,n_i^*}\Delta R_{i,n_i^*} \\ \leq S_{i,n_i^w+1}\left(\Delta R_{i,n_i^w+1} + \Delta R_{i,n_i^w+2} + \ldots + \Delta R_{i,n_i^*}\right)\end{aligned} \tag{32}$$

and also

$$\begin{aligned}S_{j,n_j^*+1}\Delta R_{j,n_j^*+1} + S_{j,n_j^*+2}\Delta R_{j,n_j^*+2} + \ldots + S_{j,n_j^w}\Delta R_{j,n_j^w} \\ \geq S_{j,n_j^w}\left(\Delta R_{j,n_j^*+1} + \Delta R_{j,n_j^*+2} + \ldots + \Delta R_{j,n_j^w}\right)\end{aligned} \tag{33}$$

Substitution of (32) and (33) into (31) gives us (34):

$$\begin{aligned}\left(\Delta_i^* + \Delta_j^*\right) - \left(\Delta_i^w + \Delta_j^w\right) \\ \geq -S_{i,n_i^w+1}\left(\Delta R_{i,n_i^w+1} + \Delta R_{i,n_i^w+2} + \ldots + \Delta R_{i,n_i^*}\right) \\ + S_{j,n_j^w}\left(\Delta R_{j,n_j^*+1} + \Delta R_{j,n_j^*+2} + \ldots + \Delta R_{j,n_j^w}\right)\end{aligned} \tag{34}$$

Now, since both WMRS and global optimum solutions meet the same redundancy rate constraint, we should have:

$$\left(\Delta R_{i,n_i^w+1} + \Delta R_{i,n_i^w+2} + \ldots + \Delta R_{i,n_i^*}\right) = \left(\Delta R_{j,n_j^*+1} + \Delta R_{j,n_j^*+2} + \ldots + \Delta R_{j,n_j^w}\right) = A \tag{35}$$

Now, equation (34) can be simplified as:

$$\left(\Delta_i^* + \Delta_j^*\right) - \left(\Delta_i^w + \Delta_j^w\right) \geq A\left(S_{j,n_j^w} - S_{i,n_i^w+1}\right) \tag{36}$$

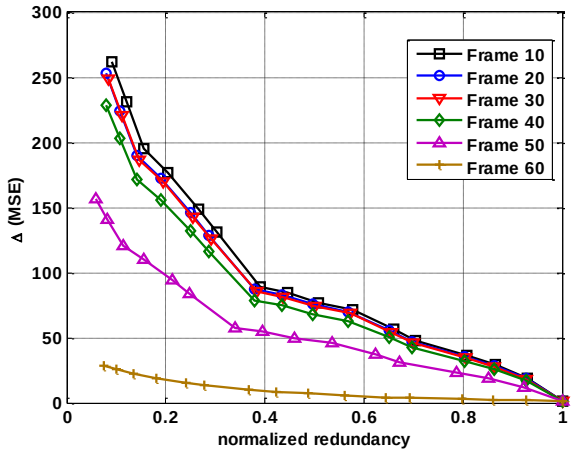
We can say that $S_{j,n_j^w} \geq S_{i,n_i^w+1}$ since based on the WMRS method, we select the slopes in descending order; S_{j,n_j^w} is selected while S_{i,n_i^w+1} is not selected, therefore $S_{i,n_i^w+1}$is smaller than S_{j,n_j^w}. Therefore:

$$\left(\Delta_i^* + \Delta_j^*\right) - \left(\Delta_i^w + \Delta_j^w\right) \geq 0 \quad (37)$$

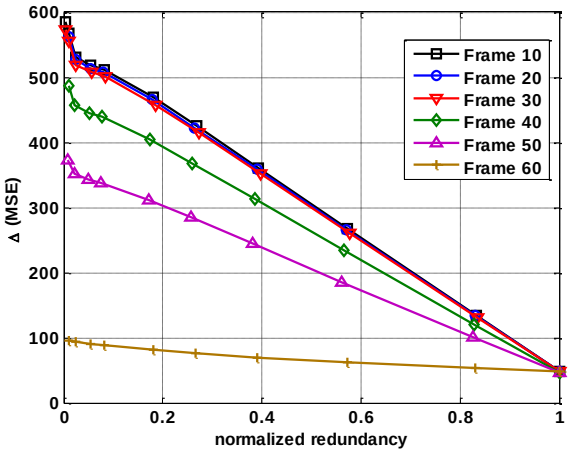
The above equation denotes that the output of WMRS gives a smaller distortion than the global optimum, which is impossible. Therefore, the assumption of $n_i^* > n_i^w$ is not valid. Similarly, we can show that the assumption of $n_i^* < n_i^w$ leads to the same conflict as well.

The above discussion can be extended for the case of having multiple frames for which $n_i^* \neq n_i^w$. Therefore, it is proven that for the convex problems, WMRS converges to the global optimal points.

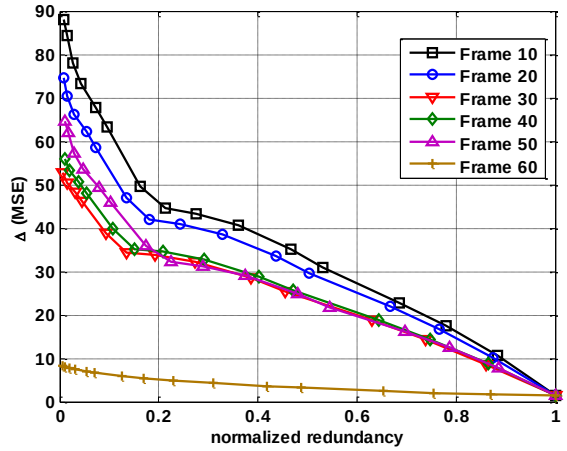
Now, we check the convexity of the function Δ_i with respect to $R_{r,i}$. Fig. 16 shows the behavior of Δ_i with variation in the normalized redundancy, for two typically low and high values of QP_p, and four test videos. It can be seen that this function is either convex or linear most of the times. In some cases, it is concave but its concavity is not significant. This figure is for GOP = 60, and the curves for GOP = 30 can be found in Fig. 2.



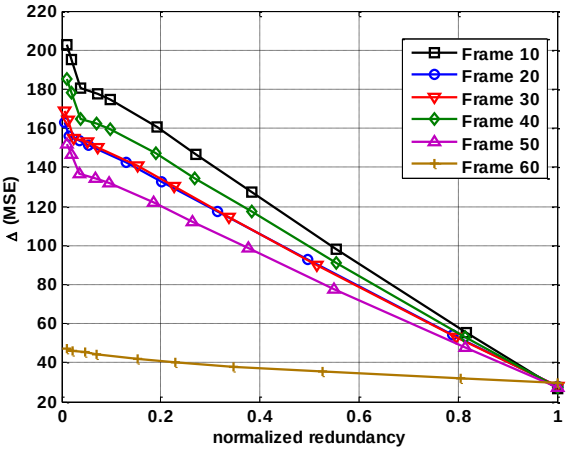
(a) *Mobile* , $QP_p = 16$



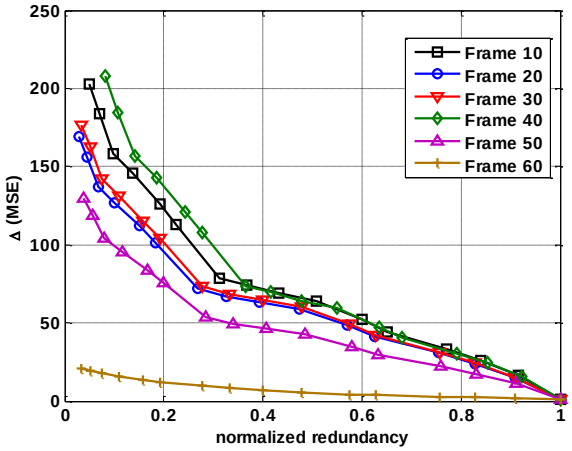
(b) *Mobile* , $QP_p = 32$



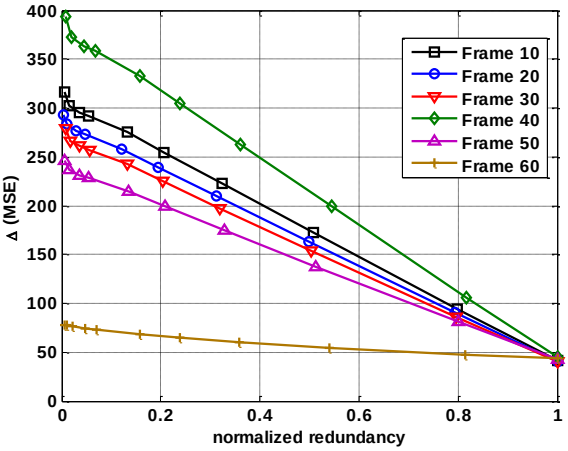
(c) *Foreman* , $QP_p = 16$



(d) *Foreman* , $QP_p = 32$



(e) *Flower* , $QP_p = 16$



(f) *Flower* , $QP_p = 32$

Fig. 12. Convexity of Δ_i with normalized redundancy for some frames of the test sequences with GOP = 60 and two values of QP_p

REFERENCES

[1] V. K. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Processing Magazine.*, vol. 18, pp. 74–93, Sep. 2001.

[2] Y. Wang, A. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proc. IEEE*, vol. 93, issue 1, pp. 57–70, Jan. 2005.

[3] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Sel. Areas Commun.*, vol. 18, issue 6, pp. 1012–1032, Jun 2000.

[4] Z. He, J. Cai, and C. W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. Circuits Syst. Video Technol.*, Special Issue on Wireless Video, vol. 12, issue 6, pp. 511–523, Jun. 2002.

[5] Y. Wang, Z. Wu, and J. M. Boyce, "Modeling of transmission-loss induced distortion in decoded video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, issue 6, pp. 716–732, Jun. 2006.

[6] Z. Chen, and D. Wu, "Prediction of transmission distortion for wireless video communication: analysis," *IEEE Trans. Image Process,* vol. 21, issue 3, pp. 1123–1137, March. 2012.

[7] A. Huszak, and S. Imre, "Analysing GOP structure and packet loss effects on error propagation in MPEG-4 video streams," *4th Inl. Symposium on Communications, Control and Signal Processing (ISCCSP),* pp.1-5, March 2010.

[8] N. Kamnoonwatana, D. Agrafiotis, and N. Canagarajah, "Flexible Adaptive Multiple Description Coding for Video Transmission," *IEEE Trans. Circuits Syst. Video Technol*, vol. 22, issue 1, pp. 1-11, Jan. 2012.

[9] A. Reibman, "Optimizing multiple description video coders in a packet loss environment," *Proceedings of Packet Video Conference, Pittsburgh*, PA, USA, 2002.

[10] I.K. Kun, and N.I. Cho, "Hybrid multiple description video coding using optimal DCT coefficient splitting and SD/MD switching," *Signal Process.: Image Commun*., vol. 21, issue 4, pp. 293-305, April 2006.

[11] R. Zhang, S.L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Sel. Areas Commun*. vol. 18, issue 6, Jun 2000.

[12] C. Lin, T. Tillo, Y. Zhao, and B. Jeon, "Multiple Description Coding for H.264/AVC with Redundancy Allocation at Macro Block Level," *IEEE Trans. Circuits Syst. Video Technol,*,vol. 21, issue 5, pp. 589-600, May 2011.

[13] T.Tillo, M. Grangetto, and G.Olmo, "Redundant Slice Optimal Allocation for H.264 Multiple Description Coding," *IEEE Trans. Circuits Syst. Video Technol*, vol. 18, issue 1, pp. 59-70, Jan. 2008.

[14] Y. Xu and C. Zhu, "End-to-End Rate-Distortion Optimized Description Generation for H.264 Multiple Description Video Coding," *IEEE Trans. Circuits Syst. Video Technol*, vol. 23, issue 9, pp. 1523 –1536, Sept. 2013.

[15] M. Kazemi, K. H. Sadeghi, S. Shirmohammadi, and P. Moallem, "Rate/distortion optimization in multiple description video coding," *Signal Processing: Image Communication*, vol. 36, pp. 95-105, Aug. 2015.

[16] A.R. Reibman, H. Jafarkhani, Y. Wang, M.T. Orchard, and R. Puri, "Multiple-description video coding using motion-compensated temporal prediction," *IEEE Trans. Circuits Syst. Video Technol,* vol. 12, issue 3, pp. 193–204, 2002.

[17] O. Campana, R. Contiero, and G.A. Mian, "An H.264/AVC video coder based on a multiple description scalar quantizer," *IEEE Trans. Circuits Syst. Video Technol*, vol. 18, issue 2, pp. 268–272, Feb. 2008.

[18] M. Kazemi, K.H. Sadeghi, and S. Shirmohammadi, "A mixed layer multiple description video coding scheme," *IEEE Trans. Circuits Syst. Video Technol,*vol 22, issue 2, pp. 202–215, Feb. 2012.

[19] T. Tillo and G. Olmo, "Data-dependent pre- and postprocessing multiple description coding of images," *IEEE Trans. Image Process*, vol. 16, issue 5, pp. 1269–1280, May. 2007.

[20] S. Shirani, "Content-based multiple description image coding," *IEEE Trans. Multimedia*, vol. 8, issue 2, pp. 411–419, April. 2006.

[21] J.G. Apostolopoulos, "Error-resilient video compression through the use of multiple states," *Proc. of International Conference on Image Processing*, vol. 3, pp. 352–355, 2000.

[22] H. Bai, W. Lin, M. Zhang, A. Wang, and Y. Zhao, "Multiple Description Video Coding Based on Human Visual System Characteristics," *IEEE Trans. Circuits Syst. Video Technol*, vol. 24, issue 8, pp. 1390-1394, Aug. 2014.

[23] R. Puri, k. Ramchandran, "Multiple description source coding through forward error correction codes," *Proc. of 33rd Asilomar Conf. on Signals, Systems and Computers*, vol. 1, pp. 342-346, 1999.

[24] M. Kazemi, S. Shirmohammadi, and K. H. Sadeghi, "A Review of Multiple Description Coding techniques for Error-Resilient Video Delivery," *Springer Multimedia Systems,* vol. 20, issue 3, pp. 283-309, April. 2013.

[25] E. Lam, and J. Goodman, "Modeling DCT coefficients for fast video encoding," *IEEE Trans. Circuits Syst. Video Technol*., vol. 9, issue 4, pp. 608–616, Jun. 1999.

[26] D. P. Williamson, and D. B. Shmoys, "The design of approximation algorithms", Cambridge University Press, 2011.

[27] V. Temlyakov, "Greedy approximation in convex optimization," Constr. Approx. vol. 41, issue 2, pp. 269-296, 2015.

[28] C. Zhou, C. Lin, and Z. Guo, "*m*DASH: A markov decision-based rate adaptation approach for dynamic HTTP streaming," *IEEE Trans. Multimedia*, vol. 18, issue 4, pp. 738-751, April 2016.

[29] J. Xiao, T. Tillo, C. Lin, and Y. Zhao, "Dynamic sub-GOP forward error correction code for real-time video applications," *IEEE Trans. Multimedia*, vol. 14, pp. 1298–1308, Aug. 2012.

[30] C. Chang, C. Chou, D. Chan, T. Lin, and M. Chen, "Accurate bitrate model and greedy-based rate controller for low delay video transmission," *IEEE Syst. J.*, vol. 6 , issue 3, pp. 414-425, Sept.2012.

[31] J. Park, H. Lee, S. Lee, and A. C. Bovik, "Optimal channel adaptation of scalable video over a multi-carrier based multi-cell environment," *IEEE Trans. Multimedia*, vol. 11, issue 6, pp. 1062–1071, Oct. 2009.

[32] H. Wang and A. Ortega, "Rate-distortion optimized scheduling for redundant video representations," *IEEE Trans. Image Process.*, vol. 18, issue 2, pp. 225–240, Feb. 2009.

[33] J. Wang, D. K. W. Chiu, Q. Li, and M. Li, " Service Sharing for Streaming Video Multicast," *IEEE Trans. Multimedia*, vol. 10, issue 7, pp. 1393–1405, Nov. 2008.

[34] Y. Shoham, and A. Gersho, "Efficient Bit Allocation for an Arbitrary Set of Quantizers", *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 36, issue 9, pp. 1445-1453, Sept. 1988.

[35] P. T. Boggs, and J. W. Tolle, "Sequential quadratic programming," *Acta Numerica*, vol. 4, pp. 1–52, 1996.

[36] X. Li, N. Oertel, A. Hutter, and A. Kaup, "Laplace distribution based Lagrangian rate distortion optimization for hybrid video coding," *IEEE Trans. Circuits Syst. Video Technol*., vol. 19, issue 2, pp. 193–205, Feb. 2009

[37] N. Kamaci, Y. Altinbasak, and R. M. Mersereau, "Frame bit allocation for H.264/AVC video coder via Cauchy-density-based rate and distortion models," *IEEE Trans. Circuits Syst. Video Technol*., vol. 15, issue 8, pp. 994–1006, Aug. 2005.

[38] http://iphome.hhi.de/suehring/tml/download/old_jm

Mohammad Kazemi received the B.Sc. degree from the Isfahan University of Technology, Isfahan, Iran, in 2003, and the M.Sc. and PhD degrees from the Sharif University of Technology (SUT), Tehran, Iran, both in electrical engineering, in 2005 and 2012, respectively. He currently works with Electrical Engineering Department, University of Isfahan, Iran, as an Assistant Professor. Dr. Kazemi served as Technical Program Committee of IEEE ISM conference and DSVCC IEEE CloudCom workshop. His current research interests include the areas of image/video processing, coding, and transmission, with emphasis on error resilient video coding, as well as digital systems design, specifically for video coding applications.

Razib Iqbal is currently affiliated with Missouri State University, USA as an Assistant Professor in the Department of Computer Science. He started his academic career at Valley City State University, USA as an Assistant Professor of Software Engineering in 2014. From 2011-2014, Iqbal worked at Bridgewater Systems division of Amdocs as a Development Expert. Iqbal earned his PhD in Computer Science from the University of Ottawa, Canada in 2011. He received the "OCRI futures awards – Student researcher of the Year 2010" for his contribution to the compressed-domain adaptive video distribution concept. His academic and professional experiences and interests are in the areas of Multimedia communications, Internet of things, Software engineering, and Software quality assurance. In Fall 2014, Iqbal received "Viewer's choice award" in VCSU faculty scholarship symposium. In Spring 2016, he received MSU CNAS "Faculty excellence for teaching" and "Student award for faculty excellence" awards. Iqbal has published over 20 peer-reviewed articles in the leading journals, conferences and workshops. He reviewed numerous journal and conference articles as well as grant applications and served on the technical program committee of well-known conferences in his research areas, including ACM Multimedia, MMSys, SIGMAP, IEEE ISM and ICME.

Shervin Shirmohammadi, SM '04, received his Ph.D. in Electrical Engineering from the University of Ottawa, Canada, where he is currently a Professor with the School of Electrical Engineering and Computer Science. He is Director of the Distributed and Collaborative Virtual Environment Research Laboratory, doing research in multimedia systems and networks, specifically video systems, gaming systems, and multimedia-assisted healthcare systems. The results of his research, funded by more than $13 million from public and private sectors, have led to 300 publications, over 65 researchers trained at the postdoctoral, PhD, and Master'slevels, over 20 patents and technology transfers to the private sector, and a number of awards. He is the Associate Editor-in-Chief of IEEE Transactions on Instrumentation and Measurement, Senior Associate Editor of ACM Transactions on Multimedia Computing, Communications, and Applications, an Associate Editor of IEEE Transactions on Circuits and Systems for Video Technology, and was an Associate Editor of Springer's Journal of Multimedia Tools and Applications from 2004 to 2012. Dr. Shirmohammadi is a University of Ottawa Gold Medalist, a licensed Professional Engineer in Ontario, a Senior Member of IEEE, and a Lifetime Professional Member of the ACM.