Fast and efficient mode and quantizer selection in the rate distortion sense for H.263 *

*Guido M. Schuster, †Aggelos K. Katsaggelos

Northwestern University
Department of Electrical Engineering and Computer Science
McCormick School of Engineering and Applied Science
Evanston, IL 60208-3118
Email: *gschuster@nwu.edu, †aggk@eecs.nwu.edu

ABSTRACT

In this paper, a fast and efficient method for selecting the encoding modes and the quantizers for the ITU H.263[1] standard is presented. H.263 is a very low bit rate video coder which produces satisfactory results at bit rates around 24 kbits/second for low motion quarter common intermediate format (QCIF) color sequences such as "Mother and Daughter". Two major target applications for H.263 are video telephony using public switched telephone network lines and video telephony over wireless channels. In both cases, the channel bandwidth is very small, hence the efficiency of the video coder needs to be as high as possible. The presented algorithm addresses this problem by finding the smallest frame distortion for a given frame bit budget. The presented scheme is based on Lagrangian Relaxation and Dynamic Programming (DP). It employees a fast evaluation of the operational rate distortion curve in the DCT domain and a fast iterative search which is based on a Bezier function.

Keywords: Very Low Bit Rate Video Coding, H.263, Operational Rate Distortion Theory.

1 INTRODUCTION

The recent very low bit rate video coding standard of the ITU, H.263, is mainly aimed for video telephony applications over channels with a very small bandwidth. Therefore the efficiency of this coder should be as high as possible. As for every lossy coding scheme, there is an inherent tradeoff between the rate and the distortion in the sense that a small rate results in a high distortion whereas a small distortion requires a large rate. The rate distortion characteristics of H.263 are largely based on the mode and quantizer selection for each macro block.

For QCIF sequences, which are of dimensions 176×144 pixels, H.263 splits a frame into 99 macro blocks of size 16×16 pixels. This results in 9 rows of 11 macro blocks. Each macro block consists of 4 luminance blocks and 2 chrominance blocks, since the color information is down-sampled by a factor of 2 in each dimension. There are two different picture types, INTRA and INTER. An INTRA picture is basically a JPEG frame and we will

*To appear in the Proceedings of *Visual Communications and Image Processing 1996*

concentrate on the more interesting INTER pictures.

The following discussion assumes that H.263 is run in its default mode, i.e., none of the options is activated. We will treat the "Advanced Prediction mode" in section 4. For INTER pictures, each macro block can be encoded using one of three different modes, Intra, Inter, and Skip. For an Intra coded macro block, a JPEG-like scheme is employed and the motion vector is considered to be zero. In the case of an Inter macro block, a single motion vector is used to predict the macro block using a previously reconstructed frame. The prediction error is then encoded using a similar scheme as for the Intra block. We assume that the motion vector is known, i.e., it has been obtained by some means, most likely by block matching. If the Skip mode is used, then the macro block is replaced by the macro block at the same location in the previously reconstructed frame and the motion vector is also considered to be zero. In the case of an Inter or Intra mode, H.263 offers 31 different quantizers for the DCT coefficients of the original macro block (Intra) or the prediction error (Inter).

In section 2 we define the problem and introduce the necessary notation, in section 3 we derive the optimal mode and quantizer selection algorithm for default H.263 which we extend in section 4 to include the"Advanced Prediction mode". In section 5 we propose a fast evaluation of the rate distortion function and in section 6 we propose an iterative scheme to find the optimal Lagrangian multiplier λ^*. In section 7 we present experimental results and we summerize the paper in section 8.

2 PROBLEM FORMULATION

The goal of this research is to optimally select the quantizers and encoding modes of H.263 on the frame level. Therefore we are not concerned about the bit allocation among the frames of a sequence, which has been studied in Ref.[2] for MPEG coders, but we assume that a rate control algorithm has given us the maximum number of bits available (R_{max}) to code the current frame.

The following notation will be used. Let $m_{r,c}$ be the macro block at row r and column c, where $r \in [0, \ldots, 8]$ and $c \in [0, \ldots, 10]$ for a QCIF video sequence. Let $q_{r,c} \in Q$ denote the quantizer for $m_{r,c}$, where $Q = \{1, \ldots, 31\}$ is the set of all admissible quantizers. Let $e_{r,c} \in E$ denote the encoding mode for $m_{r,c}$, where $E = \{\text{Intra}, \text{Inter}, \text{Skip}\}$ is the set of all admissible encoding modes. Let $x_{r,c} = [q_{r,c}, e_{r,c}] \in X$, be the decision vector for $m_{r,c}$, where $X = Q \times E$ is the set of all admissible decision vectors. Finally let $D(x_{0,0}, \ldots, x_{8,10})$ be the frame distortion and $R(x_{0,0}, \ldots, x_{8,10})$ the frame rate (Number of bits allocated for a given frame).

The optimal bit allocation problem can now be stated as follows,

$$\min_{x_{0,0},\ldots,x_{8,10}} D(x_{0,0}, \ldots, x_{8,10}), \qquad \text{subject to:} \qquad R(x_{0,0}, \ldots, x_{8,10}) \leq R_{max}. \tag{1}$$

This constrained discrete optimization problem is very hard to solve in general. In fact the approach we propose will not necessarily find the optimal solution but only the solutions which belong to the convex hull of the operational rate distortion curve. On the other hand, as we show in section 7, the solutions on the rate distortion curve tend to be dense and hence the convex hull approximation is very good.

We solve this problem using the concept of Lagrangian relaxation,[3,4] which is a well known tool in operations research. It is mainly used to relax some constraints which destroy the integrality property of an integer program. The relaxed integer program can then be solved by linear programming which leads to an efficient method for certain problems. In this application we will use Lagrangian relaxation to relax the maximum rate constraint so that the relaxed problem can be solved by DP.

First we introduce the Lagrangian frame cost function which is of the following form,

$$J_\lambda(x_{0,0}, \ldots, x_{8,10}) = D(x_{0,0}, \ldots, x_{8,10}) + \lambda * R(x_{0,0}, \ldots, x_{8,10}), \tag{2}$$

where $\lambda \geq 0$ is called the Lagrangian multiplier. It has been shown[5,3,4] that if there is a λ^* such that,

$$[x^*_{0,0}, \ldots, x^*_{8,10}] = \arg \min_{[x_{0,0},\ldots,x_{8,10}] \in X^{99}} J_{\lambda^*}(x_{0,0}, \ldots, x_{8,10}), \tag{3}$$

leads to $R(x^*_{0,0}, \ldots, x^*_{8,10}) = R_{max}$, then $x^*_{0,0}, \ldots, x^*_{8,10}$ is also an optimal solution to (1). It is well known that when λ sweeps from zero to infinity, the solution to problem (3) traces out the convex hull of the rate distortion curve, which is a non-increasing function. Hence bisection[6] could be used to find λ^*. We present in section (6) a fast search which is based on a Bezier function. The main problem is therefore to find the optimal solution to problem (3).

In general, the problem of (3) needs to be solved by an exhaustive search which is not a feasible approach. If we assume that the frame rate and the frame distortion can be expressed as the sum of the individual macro block rates and distortions, and that those macro block rates and distortions only depend on a local neighborhood, then problem (3) can be solved efficiently.[7] When H.263 is used in the default mode, then this assumption is clearly satisfied for the frame rate since the rate for a given macro block depends only on three neighboring macro blocks. This dependence results from the encoding of the motion vector of a given macro block. The motion vectors belonging to the macro blocks to the left, directly above and above and to the right of the current macro block are used to predict the current motion vector. The vector median of these three motion vectors is formed and the difference between this vector and the current motion vector is encoded using a variable length codeword. Hence if we use an additive distortion measure with respect to the blocks, then the above assumption holds also for the distortion and we can solve problem (3) efficiently. Because of its popularity and for reasons which will be apparent in section 5, we use the mean squared error (MSE) or a blockwise weighted MSE, as distortion measure.

3 OPTIMIZATION OF DEFAULT H.263

As pointed out in the previous section, the rate for a given macro block depends on three neighboring macro blocks. Even though we derived a general theory which can deal with this dependency,[7] the complexity for the required optimization is quite high. Since this study is geared towards wireless video telephony, the available computing power is restricted and will be restricted for quite some time. On the other hand, wireless channels are noisy and therefore any H.263 implementation for wireless video telephony will employ the Group of Blocks (GOB) structure of H.263. If the GOB structure is used, then each row is sent independently of any other row. Therefore if a transmission error occurs, it does not propagate into other rows.

Besides facilitating a higher degree of robustness, the GOB structure also leads to the fact the optimization of Eq. (3) can be executed for each row separately. Therefore the frame distortion and the frame rate can be expressed as the sum of the row distortions $D_r(x_{r,0}, \ldots, x_{r,10})$ and the row rates $R_r(x_{r,0}, \ldots, x_{r,10})$,

$$D(x_{0,0}, \ldots, x_{8,10}) = \sum_{r=0}^{8} D_r(x_{r,0}, \ldots, x_{r,10}), \tag{4}$$

and

$$R(x_{0,0}, \ldots, x_{8,10}) = \sum_{r=0}^{8} R_r(x_{r,0}, \ldots, x_{r,10}). \tag{5}$$

Let us define the Lagrangian row cost function as follows,

$$J_{\lambda,r}(x_{r,0}, \ldots, x_{r,10}) = D_r(x_{r,0}, \ldots, x_{r,10}) + \lambda * R_r(x_{r,0}, \ldots, x_{r,10}). \tag{6}$$

Problem (3) can now be written as follows,

$$\min_{[x_{0,0},\ldots,x_{8,10}] \in X^{99}} J_{\lambda^*}(x_{0,0}, \ldots, x_{8,10}) = \sum_{r=0}^{8} \min_{[x_{r,0},\ldots,x_{r,10}] \in X^{11}} J_{\lambda^*,r}(x_{r,0}, \ldots, x_{r,10}). \tag{7}$$

Hence if we can solve the following problem optimally for every row r,

$$\min_{[x_{r,0},\ldots,x_{r,10}]\in X^{11}} J_{\lambda,r}(x_{r,0},\ldots,x_{r,10}), \tag{8}$$

then the problem of Eq. (3) can also be solved. The GOB structure also reduces the dependencies between macro blocks, since the predictor for the current motion vector is just the motion vector from the macro block immediately to the left. Recall that the current motion vector cannot be predicted from motion vectors outside the GOB, since that would otherwise lead to a dependency among the different rows. Therefore the rate $R_{r,c}(x_{r,c-1}, x_{r,c})$ to encode the current macro block $m_{r,c}$ only depends on the mode and the quantizer choices for $m_{r,c-1}$ and $m_{r,c}$. Furthermore, the macro block distortion $D_{r,c}(x_{r,c})$ depends only on the mode and quantizer selection for the current macro block. Therefore, the row distortion can be expressed as a sum over the macro block distortions,

$$D_r(x_{r,0},\ldots,x_{r,10}) = \sum_{c=0}^{10} D_{r,c}(x_{r,c}), \tag{9}$$

and the row rate can be expressed as a sum over the macro block rates,

$$R_r(x_{r,0},\ldots,x_{r,10}) = \sum_{c=0}^{10} R_{r,c}(x_{r,c-1}, x_{r,c}), \tag{10}$$

where

$$R_{r,c}(x_{r,c-1}, x_{r,c}) = R_{r,c}^{DFD}(x_{r,c}) + R_{r,c}^{DVF}(x_{r,c-1}, x_{r,c}) + R_{r,c}^{Q}(x_{r,c-1}, x_{r,c}). \tag{11}$$

The above equation states that the rate for the current macro block depends on three different terms: (1) The rate to encode the the prediction error, (also called the displaced frame difference, DFD), $R_{r,c}^{DFD}(x_{r,c})$, which only depends on the quantizer and mode selection for macro block $m_{r,c}$. Recall that we assumed that the motion vector used for the Inter mode has already be selected. Note that we have developed a coder where the motion estimation is also included in the optimal selection process.[7] (2) The rate to encode the displacement vector field (DVF) $R_{r,c}^{DVF}(x_{r,c-1}, x_{r,c})$, which depends on the mode of the previous macro block and the mode of the current one because the difference between the motion vectors is encoded by a variable length codeword. (3) The rate to transmit the quantizer $R_{r,c}^{Q}(x_{r,c-1}, x_{r,c})$, which depends on the quantizer of the current macro block and the quantizer of the previous macro block since the difference between these two quantizers is encoded. Since this difference is limited by the H.263 syntax to be in the range of $-2,\ldots,2$, this rate is set to infinity for every combination of quantizers which are more than two step sizes apart. Therefore the optimization procedure will only pick quantizer sequences which can be encoded using the H.263 syntax. This rate also depends on the encoding mode since the Skip mode does not allow for a change of quantizers. Note that $R_{r,0}(x_{r,-1}, x_{r,0})$ is a special case since $x_{r,-1}$ does not belong to a macro block. This does not have any influence on the DFD rate, since it only depends on $x_{r,0}$. For the purpose of DVF encoding, the motion vector of this outside block is considered zero, and for the first quantizer value in a GOB, $q_{r,0}$, a fixed length five bit code word is sent.

Using the above macro block rates and distortions we define the Lagrangian macro block cost function as

$$J_{\lambda,r,c}(x_{r,c-1}, x_{r,c}) = D_{r,c}(x_{r,c}) + \lambda * R_{r,c}(x_{r,c-1}, x_{r,c}). \tag{12}$$

The minimization problem of Eq. (8) can now be rewritten as,

$$\min_{[x_{r,0},\ldots,x_{r,10}]\in X^{11}} \sum_{c=0}^{10} J_{\lambda,r,c}(x_{r,c-1}, x_{r,c}). \tag{13}$$

The above problem can be solved efficiently using forward Dynamic Programming (DP),[8] which is also known as the Viterbi[9] algorithm. For that purpose we will derive the DP recursion formula. We define the optimal Lagrangian cost up to and including column k for row r as follows,

$$J^*_{\lambda,r,k}(x_{r,k}) = \min_{[x_{r,0},\ldots,x_{r,k-1}]\in X^{k}} \sum_{c=0}^{k} J_{\lambda,r,c}(x_{r,c-1}, x_{r,c}). \tag{14}$$

Note that $\min_{x_{r,10}\in X}(J^*_{\lambda,r,10}(x_{r,10}))$ is the optimal solution to problems (13) and (8), which in turn enables us to solve the original problem of Eq. (3). From the above definition it follows that the optimal Lagrangian cost at $k+1$ is equal to,

$$
\begin{aligned}
& J^*_{\lambda,r,k+1}(x_{r,k+1}) && (15)\\
&= \min_{[x_{r,0},\ldots,x_{r,k}]\in X^{k+1}} \sum_{c=0}^{k+1} J_{\lambda,r,c}(x_{r,c-1},x_{r,c}) && (16)\\
&= \min_{x_{r,k}\in X}\left\{\min_{[x_{r,0},\ldots,x_{r,k-1}]\in X^k}\left[\sum_{c=0}^{k}\left(J_{\lambda,r,c}(x_{r,c-1},x_{r,c})\right)+J_{\lambda,r,k+1}(x_{r,k},x_{r,k+1})\right]\right\} && (17)\\
&= \min_{x_{r,k}\in X}\left\{\min_{[x_{r,0},\ldots,x_{r,k-1}]\in X^k}\left[\sum_{c=0}^{k}\left(J_{\lambda,r,c}(x_{r,c-1},x_{r,c})\right)\right]+J_{\lambda,r,k+1}(x_{r,k},x_{r,k+1})\right\}, && (18)
\end{aligned}
$$

which results in the DP recursion formula,

$$J^*_{\lambda,r,k+1}(x_{r,k+1}) = \min_{x_{r,k}\in X}\left\{J^*_{\lambda,r,k}(x_{r,k}) + J_{\lambda,r,k+1}(x_{r,k},x_{r,k+1})\right\}. \quad (19)$$

The recursion is initialized with

$$J^*_{\lambda,r,0}(x_{r,0}) = J_{\lambda,r,0}(x_{r,-1},x_{r,0}). \quad (20)$$

Having established the DP recursion formula, the Viterbi algorithm can be used to find the optimal sequence of decision vectors $x^*_{r,0},\ldots,x^*_{r,10}$ for each row separately, which in turn leads to the quantizers and modes for the entire frame. The time complexity of the Viterbi algorithm is $O(9*11*|X|^2)$ where we measure the time complexity in numbers of required comparisons. Because of the independence of the rows, the optimization for each row can be executed in parallel. Since $X = Q\times E$, the cardinality of X, $|X|$, equals $31*3=93$. As stated above, this time complexity does not account for the evaluation of the operational rate distortion function, but only for the number of comparisons. We will investigate the evaluation of the rate distortion function in section (5). As pointed out earlier, the Viterbi algorithm is invoked for every iteration of the search for the optimal λ^*. We will present in section (6) a search for λ^* which usually requires only three Viterbi runs.

4 OPTIMIZATION OF H.263 WITH ADVANCED PREDICTION MODE

So far we have considered the encoding of an INTER picture for default H.263. One of the H.263 options is the "Advanced Prediction mode". This mode is set for the entire picture in the picture header. In the "Advanced Prediction mode", the motion compensation is accomplished by overlapped block motion compensation and a new encoding mode, which we call Advanced, is introduced. If the Advanced encoding mode is used, then each of the four luminance blocks of a macro block is predicted using its own motion vector. Again we assume that these motion vectors have already been selected. Therefore the set of all admissible encoding modes is now $E = \{\mathrm{Intra},\mathrm{Inter},\mathrm{Skip},\mathrm{Advanced}\}$. The Inter and Advanced encoding modes require a motion compensated prediction of the current macro block. In the "Advanced Prediction mode", the motion compensation is accomplished by overlapped block motion compensation. The overlapped block motion compensation is only applied to the luminance blocks and works as follows. Each pixel in a 8×8 luminance block is predicted as a weighted average of three luminance values pointed to by three motion vectors. The three motion vectors involved are the vector of the current luminance block and the two vectors which belong to the two neighboring 8×8 blocks which are the closest to the pixel under consideration. Therefore the luminance prediction of the current macro block depends not only on the decision vector of the preceeding macro block, but also on the decision vector of the succeeding one since motion vectors of both of these blocks are used to predict the current macro block. Again, because of the GOB structure, the current macro block is independent of any macro block which is not in the same row.

Since the current macro block depends on both the preceeding and the succeeding macro blocks, a second order DP can be used to solve this optimization problem.[7] The time complexity of such a solution approach is $O(11*9*|X|^3)$ which might be too expensive. We therefore propose an approximation which uses a first order DP. Clearly the increased dependency between macro blocks is based on the overlapped block motion compensation. Since this is only applied to the luminance blocks we concentrate on the luminance rate and the luminance distortion to derive a fast and accurate approximation. It is interesting to notice that the macro block luminance distortion $D_{r,c}^{Y}(x_{r,c-1}, x_{r,c}, x_{r,c+1})$ which depends on the preceeding, current and succeeding decision vectors can be decomposed as follows,

$$D_{r,c}^{Y}(x_{r,c-1}, x_{r,c}, x_{r,c+1}) = D_{r,c}^{LY}(x_{r,c-1}, x_{r,c}) + D_{r,c}^{RY}(x_{r,c}, x_{r,c+1}), \tag{21}$$

where $D_{r,c}^{LY}(x_{r,c-1}, x_{r,c})$ denotes the distortion of the two luminance blocks which form the left half of the macro block $m_{r,c}$ and $D_{r,c}^{RY}(x_{r,c}, x_{r,c+1})$ denotes the distortion of the two luminance blocks which form the right half of the macro block $m_{r,c}$. This follows from the way the overlapped block motion compensation is accomplished. The luminance DFD rate can be decomposed in a similar fashion,

$$R_{r,c}^{DFDY}(x_{r,c-1}, x_{r,c}, x_{r,c+1}) = R_{r,c}^{CBPY}(x_{r,c-1}, x_{r,c}, x_{r,c+1}) + R_{r,c}^{LDFDY}(x_{r,c-1}, x_{r,c}) + R_{r,c}^{RDFDY}(x_{r,c}, x_{r,c+1}), \tag{22}$$

where $R_{r,c}^{LDFDY}(x_{r,c-1}, x_{r,c})$ denotes the rate of the two luminance DFD blocks on the left and $R_{r,c}^{RDFDY}(x_{r,c}, x_{r,c+1})$ denotes the rate of the two luminance DFD blocks on the right, and $R_{r,c}^{CBPY}(x_{r,c-1}, x_{r,c}, x_{r,c+1})$ is the rate required for encoding the variable length parameter CBPY, which is called the "Coded block pattern for luminance". The luminance DFD blocks are numbered in raster scan order from one to four. CBPY is used to indicate which luminance DFD blocks contain nonzero transform coefficients. Let t_i be one if the i^{th} block contains nonzero coefficients and zero otherwise. Then the left side of the macro block can be represented by $u_{r,c}(x_{r,c-1}, x_{r,c}) = [t_1, t_3]^T$ and the right side by $v_{r,c}(x_{r,c}, x_{r,c+1}) = [t_2, t_4]^T$. In the remainder of this section, we will use the variables u and v without arguments, but it should be understood that they depend on decision vectors of the preceeding, current and succeeding macro blocks.

The rate for CBPY, which is part of the luminance DFD rate, depends on all four luminance blocks of a macro block and hence depends on all $x_{r,c-1}$, $x_{r,c}$ and $x_{r,c+1}$ decision vectors and cannot be separated. This can be proven by using table 6/H.263 in the H.263 standard document.[1] Assume that the rate for CBPY is separable into a right and a left rate. According to table 6/H.263, $u = [1,1], v = [1,1]$ leads to a size of four bits for CBPY. The pattern $u = [1,1], v = [0,0]$ also requires four bits. By the separability assumption the rate for $v = [1,1]$ needs to be equal to the rate for $v = [0,0]$. According to table 6/H.263, the pattern $u = [1,0], v = [1,1]$ requires five bits, and therefore, by the separability assumption, the bit rate for the pattern $u = [1,0], v = [0,0]$ should also be five bits. Using table 6/H.263, this pattern only requires four bits, which contradicts the separability assumption. Hence we have proven that the rate is not separable.

To be able to employ a first order DP, we like to predict the CBPY rate $R_{r,c}^{CBPY}(x_{r,c-1}, x_{r,c}, x_{r,c+1})$ using only the previous and the current decision vectors. We denote this predictor by $\hat{R}_{r,c}^{CBPY}(x_{r,c-1}, x_{r,c})$. To develop this predictor, we treat the decision vector from the succeeding macro block $x_{r,c+1}$ as a random variable. If we knew $x_{r,c+1}$ we could infer $v_{r,c}(x_{r,c}, x_{r,c+1})$, the luminance block pattern which belongs to the right side of the macro block. Therefore $v_{r,c}(x_{r,c}, x_{r,c+1})$ is a random variable which we denote with V. Knowing V results in the rate for CBPY since u is already known. Let $l(u,v)$ denote the code word length. Since Table 6/H.263 represents an entropy code for the u, v patterns, then the following holds,

$$p_{U,V}(u,v) = 2^{-l(u,v)}. \tag{23}$$

In other words, table 6/H.263 can be used to generate the joint probability mass function $p_{U,V}(u,v)$. Our goal is to predict the length of the CBPY code word $l(U = u, V)$, which is a random variable since it depends on V. The classical predictor for such a problem is the conditional mean, which is an unbiased, Linear Minimum Mean Square Error predictor,

$$\hat{R}_{r,c}^{CBPY}(x_{r,c-1}, x_{r,c}) = E[l(U,V)|U = u] \tag{24}$$

Since the joint probability mass function $p_{U,V}(u,v)$ is known, this expectation can be evaluated and the result is,

$$\hat{R}_{r,c}^{CBPY}(x_{r,c-1}, x_{r,c}) = \begin{cases} 20/7 & : \quad u = [0,0]^T \\ 48/11 & : \quad u = [0,1]^T \\ 48/11 & : \quad u = [1,0]^T \\ 13/3 & : \quad u = [1,1]^T \end{cases} \tag{25}$$

We can now define the row rate in terms of the above expressions,

$$\begin{aligned} &R_r(x_{r,0},\ldots,x_{r,10}) = \\ &\sum_{c=0}^{10} \Big\{ R_{r,c}^{DFDC}(x_{r,c}) + R_{r,c}^{DVF}(x_{r,c-1},x_{r,c}) + R_{r,c}^{Q}(x_{r,c-1},x_{r,c}) + \\ &\hat{R}_{r,c}^{CBPY}(x_{r,c-1},x_{r,c}) + R_{r,c}^{LDFDY}(x_{r,c-1},x_{r,c}) + R_{r,c-1}^{RDFDY}(x_{r,c-1},x_{r,c}) \Big\}, \end{aligned} \tag{26}$$

where $R_{r,c}^{DFDC}(x_{r,c})$ is the rate required for the encoding of the color channels. Note that the first three terms are equivalent to the ones in Eq. (11) with the exception that the Advanced mode, like the Skip mode, does not permit the quantizer to be changed. Hence $R_{r,c}^{Q}(x_{r,c-1},x_{r,c})$ needs to be adjusted to reflect this fact. The boundary conditions for this summation are as follows: $R_{r,-1}^{RDFDY}(x_{r,-1},x_{r,0})$ is set to zero and $R_{r,10}^{LDFDY}(x_{r,9},x_{r,10})$ contains also the luminance rate of the right side of macro bock $m_{r,10}$. Equivalently, we can define the row distortion in terms of the above expressions,

$$D_r(x_{r,0},\ldots,x_{r,10}) = \sum_{c=0}^{10} D_{r,c}^{C}(x_{r,c}) + D_{r,c}^{LY}(x_{r,c-1},x_{r,c}) + D_{r,c-1}^{RY}(x_{r,c-1},x_{r,c}), \tag{27}$$

where $D_{r,c}^{C}(x_{r,c})$ is the distortion in the color channels. The boundary conditions for this summation are as follows: $D_{r,-1}^{RY}(x_{r,-1},x_{r,0})$ equals 0 and $D_{r,10}^{LY}(x_{r,9},x_{r,10})$ contains also the luminance distortion of the right side of macro bock $m_{r,10}$.

We can now define a Lagrangian cost function at the macro block level,

$$\begin{aligned} J_{\lambda,r,c}(x_{r,c-1},x_{r,c}) \quad = \quad & D_{r,c}^{C}(x_{r,c}) + D_{r,c}^{LY}(x_{r,c-1},x_{r,c}) + D_{r,c-1}^{RY}(x_{r,c-1},x_{r,c}) + \\ & \lambda * \Big(R_{r,c}^{DFDC}(x_{r,c}) + R_{r,c}^{DVF}(x_{r,c-1},x_{r,c}) + R_{r,c}^{Q}(x_{r,c-1},x_{r,c}) + \\ & \hat{R}_{r,c}^{CBPY}(x_{r,c-1},x_{r,c}) + R_{r,c}^{LDFDY}(x_{r,c-1},x_{r,c}) + R_{r,c-1}^{RDFDY}(x_{r,c-1},x_{r,c}) \Big). \end{aligned} \tag{28}$$

Using the above definition of $J_{\lambda,r,c}(x_{r,c-1},x_{r,c})$ the derivation of the DP recursion formula in the previous section is still valid. Therefore the same Viterbi algorithm can be used to find the solution. Note that we estimate the rate for the CBPY parameter, which allows us to make a decision about the shortest path using a first order DP formulation. After the decision is made, the rate for the CBPY parameter can be observed and we should correct the prediction to reflect the knowledge gained. We formalize this predictor-corrector scheme below.

We use Eq. (20) to initialize the forward DP, i.e.,

$$J_{\lambda,r,0}^{*}(x_{r,0}) = J_{\lambda,r,0}(x_{r,-1},x_{r,0}), \qquad \forall x_{r,0} \in X. \tag{29}$$

Note that $J_{\lambda,r,0}^{*}(x_{r,0})$ contains the prediction for the CBPY rate, $\hat{R}_{r,0}^{CBPY}(x_{r,-1},x_{r,0})$. To be able to remember the optimal decision vector sequence, we introduce a back pointer $i_{r,c}(x_{r,c})$,

$$i_{r,0}(x_{r,0}) = x_{r,-1}, \qquad \forall x_{r,0} \in X. \tag{30}$$

Next, the recursion is started, hence the DP recursion formula (19) is applied for $k = 0, \ldots, 9$,

$$J^*_{\lambda,r,k+1}(x_{r,k+1}) = \min_{x_{r,k} \in X} \left\{ J^*_{\lambda,r,k}(x_{r,k}) + \left[R^{CBPY}_{r,k}(i_{r,k}(x_{r,k}), x_{r,k}, x_{r,k+1}) - \hat{R}^{CBPY}_{r,k}(i_{r,k}(x_{r,k}), x_{r,k}) \right] + J_{\lambda,r,k+1}(x_{r,k}, x_{r,k+1}) \right\}, \quad \forall x_{r,k+1} \in X. \tag{31}$$

The back pointers are calculated as follows,

$$i_{r,k+1}(x_{r,k+1}) = \arg\min_{x_{r,k} \in X} \left\{ J^*_{\lambda,r,k}(x_{r,k}) + \left[R^{CBPY}_{r,k}(i_{r,k}(x_{r,k}), x_{r,k}, x_{r,k+1}) - \hat{R}^{CBPY}_{r,k}(i_{r,k}(x_{r,k}), x_{r,k}) \right] + J_{\lambda,r,k+1}(x_{r,k}, x_{r,k+1}) \right\}, \quad \forall x_{r,k+1} \in X. \tag{32}$$

Note that the term inside the square brackets is the prediction error and we update the optimal Lagrangian cost function at each iteration. When we apply the Viterbi algorithm to the problem of section 3, no prediction is necessary and hence we omit the term inside the square brackets. Finally, the decision vector sequence for the proposed fist order DP approximation to problem (8) can be found be selecting the decision vector which leads to the smallest Lagrangian cost function at macro block $m_{r,10}$ and backtracking the solution, i.e.,

$$x^*_{r,10} = \arg\min_{x_{r,10} \in X} J^*_{\lambda,r,10}(x_{r,10}), \tag{33}$$

$$x^*_{r,k-1} = i_{r,k}(x^*_{r,k}), \qquad \text{for } k = 10, \ldots, 1. \tag{34}$$

5 FAST RATE DISTORTION EVALUATION

We previously defined the time complexity as the number of required comparisons. This definition does not capture the time spent evaluating the operational rate distortion function, which is discussed in this section. As we pointed out earlier, we use the mean squared error (MSE) or a block-wise weighted MSE, as distortion measure.

The rate distortion points for the Skip mode are the MSE between the original macro block and the macro block at the same location in the previously reconstructed frame. Note that these are independent of the quantizers. The Intra, Inter and Advanced modes are similar in the sense that for each of them we have to apply the DCT transform to a macro block (in the case of the Intra mode, this macro block is from the original frame, whereas for the Inter and the Advanced mode, it is the DFD of that macro block). Then the quantizer is applied in the DCT domain, the quantized macro block is run-length coded to evaluate the rate, and the inverse DCT transform is applied to create the reconstructed macro block. The reconstructed macro block is then used to evaluate the distortion, hence the MSE between the original macro block and the reconstructed macro block is calculated. In this straightforward implementation the evaluation of the rate distortion function requires therefore $1 * \text{MSE} + (|E| - 1) * |Q| * (\text{DCT} + \text{Quantization} + \text{Run-length} + \text{IDCT} + \text{MSE})$ operations per macro block.

This evaluation of the rate distortion function can be significantly sped up by using the fact that the employed DCT is a distance preserving transformation. This means that the squared sum of the elements in the original domain, is equal to the squared sum of the elements in the DCT domain. Therefore, the MSE can be computed in the DCT domain and no inverse DCT operation is required. We propose the following procedure for evaluating the rate distortion function for the Intra, Inter and Advanced encoding modes.

1. Select a mode, apply the DCT and store the result.
2. Select a quantizer, calculate the MSE (distortion) between the stored result and the quantizer output.

3. Run-length encode the quantizer output which results in the rate.

4. Go to 2 until all quantizers have been employed

5. Go to 1 until all modes have been employed

For the above procedure the evaluation of the rate distortion function requires $1 * \text{MSE} + (|E| - 1) * \text{DCT} + (|E| - 1) * |Q| * (\text{Quantization} + \text{Run-length} + \text{MSE})$ operations per macro block, which is substantially smaller than that of the straightforward implementation.

The number of operations for the evaluation of the rate distortion function is linear, and the time complexity of the optimization procedure is quadratic, in the number of quantizers employed. Since a nearly constant distortion is usually targeted, a reduced admissible quantizer set, which is centered around the quantizer step size which results in the desired quality, can be used without any noticeable loss of performance. The set employed in the presented experiments is $Q = \{8, 9, 10, 11, 12\}$.

6 FAST ITERATIVE SEARCH FOR λ^*

To find λ^* which leads to the desired rate R_{max}, an iterative search needs to be performed. Clearly the rate and the distortion for every macro block has to be calculated only once, but the Viterbi algorithm has to be invoked at every iteration. Consider Fig. 1 which shows the situation at hand. The proposed iterative search makes use of the fact, that for a continuous rate distortion curve, the derivative is equal to $\frac{-1}{\lambda}$. Assume that the bracketing points $P_0 = (D_0, R_0)$, $P_2 = (D_2, R_2)$ and the derivatives d_0 and d_2 are given. The goal is to approximate the convex hull between P_0 and P_2 as accurately as possible, such that, this approximation can be used to estimate the derivative d_{max} at R_{max}. This estimate of the derivative $\hat{d}_{max}$ is then used to generate the new $\lambda = \frac{-1}{\hat{d}_{max}}$ for the next iteration. The bisection[6] method, which is based on the fact that the rate and the distortion are monotonic functions of the Lagrangian multiplier λ is commonly used to find λ^*. in Ref.[10] a fast convex search is used, which is based on the convexity property of the rate distortion function. This search can be interpreted as approximating the convex hull with a straight line between P_0 and P_2, which is not the best possible approximation since the knowledge of d_0 and d_2 is not used.

A good approximation to the convex hull is a function which goes through the given bracketing points P_0 and P_2 and matches the derivatives d_0 and d_2 at these points. Also the approximation should be convex, differentiable, and in closed form. A parabola would be an ideal choice but it has only three degrees of freedom which is in general not enough to match four independent requirements (P_0, P_2, d_0, d_2). A second order Bezier curve[11] is a parabola which can be tilted in any direction in the plane, hence it has an additional degree of freedom and can therefore match the four independent requirements. A second order Bezier curve is of the following parametric form,

$$P(u) = (1-u)^2 * P_a + 2 * (1-u) * u * P_b + u^2 * P_c. \tag{35}$$

By setting $P_a = P_0$ and $P_c = P_2$, two of the four conditions are immediately satisfied, i.e., $P(0) = P_0$ and $P(1) = P_2$. The point P_b can be chosen so that the two derivative conditions are satisfied. The derivative of the second order Bezier curve is,

$$d(u) = \frac{(u-1) * R_0 + (1 - 2*u) * R_b + u * R_2}{(u-1) * D_0 + (1 - 2*u) * D_b + u * D_2}. \tag{36}$$

Therefore

$$d_0 = d(0) = \frac{R_b - R_0}{D_b - D_0} \tag{37}$$

and

$$d_2 = d(1) = \frac{R_2 - R_b}{D_2 - D_b}, \tag{38}$$

which leads to $P_b = (D_b, R_b)$ where,

$$D_b = \frac{(R_0 - d_0 * D_0) - (R_2 - d_2 * D_2)}{d_2 - d_0}, \tag{39}$$

and

$$R_b = \frac{d_2 * (R_0 - d_0 * D_0) - d_0 * (R_2 - d_2 * D_2)}{d_2 - d_0}. \tag{40}$$

As can be seen in Fig. 1, P_b is the point where the two tangents of P_0 and P_2 intersect.

Since the goal is to find $\hat{d}_{max}$, the derivative of $P(u)$ has to be evaluated at R_{max}. Hence u_{max} has to be found first which leads to $P(u_{max}) = (D_{max}, R_{max})$. This can be achieved since $P(u_{max})$ is a second order function of u_{max} for which the quadratic equation can be solved explicitly. Therefore, from Eq. (35) we obtain,

$$u^{\pm}_{max} = -\frac{R_b - R_0}{(R_2 - R_b) - (R_b - R_0)} \pm \sqrt{\frac{(R_b - R_0)^2}{((R_2 - R_b) - (R_b - R_0))^2} + \frac{R_{max} - R_0}{(R_2 - R_b) - (R_b - R_0)}}, \tag{41}$$

where u_{max} is picked such that $1 \geq u_{max} \geq 0$. In the case where $(R_2 - R_b) - (R_b - R_0)$ equals zero, u_{max} equals $(R_{max} - R_0)/(2 * (R_b - R_0))$. Hence the estimate of the derivative at R_{max}, is given by $\hat{d}_{max} = d(u_{max})$.

In our experiments, the iterative search which is based on the second order Bezier approximation exhibits a very fast convergence. As pointed out earlier, since every iteration requires a Viterbi algorithm, it is desirable to stop the iteration when a solution is found which is within a certain ϵ around the optimal solution. We also pointed out in section 4, that for the "Advanced Prediction mode" the first order DP only finds an approximate solution. In this case we cannot guarantee that the rate and the distortion are monotonic functions of the Lagrangian multiplier λ. Since the approximation is very accurate, this only becomes a problem for small changes in λ, and the initial iteration steps are not affected by this approximation. The proposed algorithm is very well suited for such cases since the initial convergence is very fast. In fact, for an ϵ of ± 50 bits in the presented experiments, a satisfactory solution was usually found with only one iteration. Note that the two bracketing points need to be found before the iterative search can be used and therefore two Viterbi algorithms need to be run regardless of the number of iterations.

7 EXPERIMENTAL RESULTS

We implemented the propose algorithm for the test model 4 (TMN4)[12] of H.263. Figure 2 shows a comparison between TMN4 using the heuristic described in the specifications and TMN4 using the optimal decision strategy. The encoded sequence is "Mother and Daughter" at 7.5 frames/second and the resulting average bit rate for TMN4 is 23.36 kBits/second with a resulting average distortion of 33.0 dB PSNR. With the proposed scheme we can solve the problem formulated in Eq. (1), but we can also solve the dual problem, that of minimizing the rate of each frame for a fixed frame distortion.

In the left figures, the distortion of each frame was minimized by the proposed algorithm while the rate for every frame of the sequence was set equal to the number of bits used by TMN4. In the right figures, the rate of each frame was minimized by the proposed algorithm while the distortion for every frame of the sequence was set equal to the distortion generated by TMN4. It is clear from Figure 2 that the optimal scheme is superior to the TMN4 heuristics in both cases, when their rates are matched and when their distortions are matched. In fact the optimal strategy leads to an average distortion improvement of 0.3 dB PSNR for the same bit rate, whereas the same distortion can be achieved with a bit rate which is 1.9 kbits/second smaller than the one of TMN4, i.e., in both cases an improvement of almost 10% is achieved. In addition to this clear advantage of the optimal scheme it is important to notice that the scheme is able to follow any arbitrary rate distortion profile in an optimal fashion, which is very important for a rate control scheme.

8 SUMMARY

We presented a fast and efficient method for selecting the encoding modes and the quantizers for the ITU H.263 standard. We showed how Lagrangian relaxation and dynamic programming can be applied to find a good solution on the convex hull. We studied the optimization procedure for default H.263 and for H.263 when the "Advanced Prediction mode" is used. We presented a fast procedure for evaluating the rate distortion function and a fast iterative algorithm for selecting the optimal tradeoff parameter λ. Finally we presented results of an implementation of the proposed algorithm for TMN4.

Note that independently of this work, a similar algorithm has been presented by Wiegand et. al.[13] for the efficient mode selection for block-based motion compensated video coders. Besides the fact that this research has been conducted independently, this paper is distinct from Ref.[13] by the following points: (1) We find the optimal mode and quantizer sequences jointly; (2) we develop the solution algebraically; (3) we propose a fast and accurate approximation for the "Advanced Prediction mode"; (4) we present a procedure for the fast evaluation of the rate distortion function and (5) we introduce a fast iterative scheme to find the optimal λ.

9 REFERENCES

[1] Expert's group on very low bit rate video telephony, ITU-Telecommunication standardization sector, *Draft Recommendation H.263.*

[2] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders," *IEEE Transactions on Image Processing*, vol. 3, pp. 533–545, Sept. 1994.

[3] H. Everett, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Operations Research*, vol. 11, pp. 399–417, 1963.

[4] M. L. Fisher, "The Lagrangian relaxation method for solving integer programming problems," *Management science*, vol. 27, pp. 1–18, Jan. 1981.

[5] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 36, pp. 1445–1453, Sept. 1988.

[6] C. F. Gerald and P. O. Wheatley, *Applied Numerical Analysis.* Addison Wesley, fourth ed., 1990.

[7] G. M. Schuster and A. K. Katsaggelos, "A video compression scheme with optimal bit allocation between displacement vector field and displaced frame difference," *IEEE Transactions on Image Processing*, submitted Dec. 1995.

[8] D. Bertsekas, *Dynamic Programming.* Prentice-Hall, 1987.

[9] G. D. Forney, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[10] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Transactions on Image Processing*, vol. 2, pp. 160–175, Apr. 1993.

[11] F. S. Hill, *Computer Graphics.* Macmillan Publishing Company, 1990.

[12] Expert's Group on Very Low Bitrate Visual Telephony, *Video Codec Test Model, TMN4 Rev1.* ITU Telecommunication Standardization Sector, Oct. 1994.

[13] T. Wiegand, M. Lightstone, T. G. Campbell, and S. K. Mitra, "Efficient mode selection for block based motion compensated video coding," in *Proceedings ICIP-95*, vol. 2, pp. 559–562, Oct. 1995.

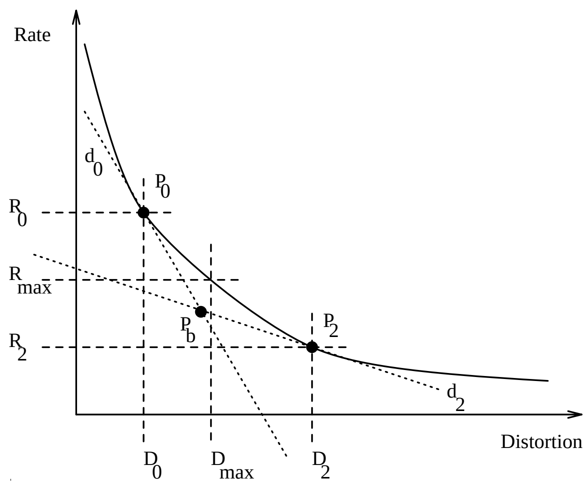

Figure 1: Typical convex hull with two bracketing points

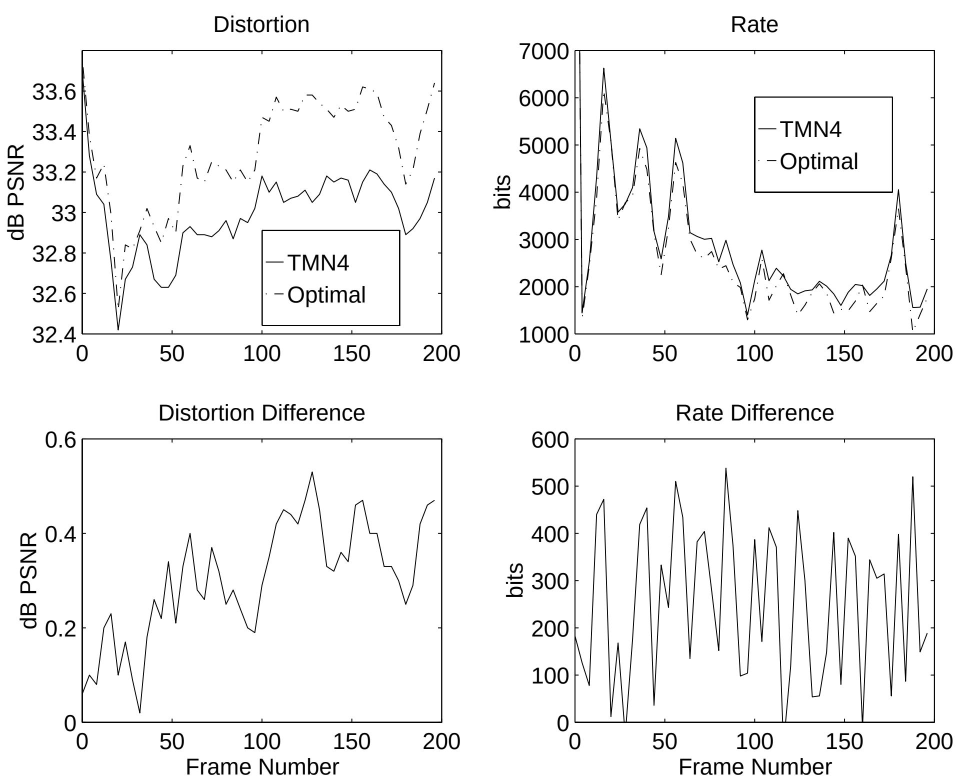

Figure 2: Left figures: Matched rate; Right figures: Matched distortion