CONVEXITY RESULTS FOR A PREDICTIVE VIDEO CODER

Yegnaswamy Sermadevi and Sheila S. Hemami

School of Electrical and Computer Engineering
Cornell University, Ithaca, NY 14853
Email: \{swamy, hemami\}@ece.cornell.edu

ABSTRACT

The problem of bit allocation among frames in a predictively encoded video sequence is fundamental to video compression. This paper examines the convexity of multivariate distortion-rate data generated by predictive encoding of video. The underlying convex structure of the distortion-rate data is used to motivate an efficient near-optimal Steepest-Descent (SD) algorithm for the bit allocation problem. Faster variations on the SD method are also constructed and evaluated. Both the convex data model and the SD algorithm are experimentally validated using an MPEG-2 encoder.

Keywords: Convexity, Video Coding, Rate-Distortion optimization, Bit allocation, Rate control

1. INTRODUCTION

Storage and transmission of video impose constraints on the bit rate available to code frames in a video sequence. A video coding standard (MPEG-2, MPEG-4, H.26x) specifies a flexible syntax which allows an encoder to adapt parameters in order to achieve rate-distortion trade-offs. For a given set of rate usage constraints, an encoder seeks to allocate bits to the video sequence by a suitable choice of syntax parameters that minimizes the overall distortion while meeting the rate constraints. For computational reasons, bit allocation in video is usually split into two steps: (i) implicitly or explicitly specifying the number of bits to be used for encoding a frame, and (ii) allocating bits to blocks within a frame. This paper analyzes the former problem in the case when the entire sequence is known a priori. An efficient algorithm for the allocation of bits within each frame is assumed [1].

There are efficient rate-distortion based bit allocation algorithms for independent coding of signal blocks [2]. However, these algorithms are not directly applicable to predictively encoded video. This paper presents and experimentally validates a convexity model for multivariate distortion-rate data which is used to motivate a Steepest Descent (SD) algorithm [3] for the bit allocation problem. Experiments with an MPEG-2 encoder show that the SD algorithm produces near-optimal solutions in practice. Faster versions of the SD method are also derived.

The paper is organized as follows. Section 2 provides background for the bit allocation problem and presents the problem formulation. The convexity model for multivariate distortion-rate functions is presented in section 3. The SD algorithm is presented in Section 4 and experimental results showing near-optimal performance are provided. Section 5 concludes the paper.

2. BACKGROUND

In motion-compensated video coding (MPEG-2, MPEG-4, H.26x), scalar quantization of block transform coefficients is the only lossy part of the encoding process. Although a vector choice for each frame is possible, for computational reasons, the rate and distortion of a frame are parameterized by a single quantization parameter that controls the step-sizes of quantizers applied to the block transform coefficients [4, 5]. In the simplest case, the encoder has to choose one among a set of admissible quantization parameter values for each frame in order to minimize the overall distortion while not exceeding a target number of bits.

Given a set of $m_k$ quantization parameter values $\{1, \ldots, m_k\}$ for coding the $k^{th}$ frame, where the corresponding quantizers are ordered from finest to coarsest, the bit allocation problem for predictive coding formulated in [4] is given by

$$Q^*_k = \arg\min_{Q=(Q_1, \ldots, Q_k)} \sum_{k=1}^{n} D_k(Q_1, \ldots, Q_k)$$

subject to $R(\tilde{Q}) = \sum_{k=1}^{n} R_k(Q_1, \ldots, Q_k) \leq R_{tot}$

where $Q_k \in \{1, \ldots, m_k\}$ is the quantization parameter choice for the $k^{th}$ frame, $D_k(\cdot)$ and $R_k(\cdot)$ are distortion and rate functions for the $k^{th}$ frame which possibly depend on the quantization parameter choices for previous encoded frames, $R_{tot}$ is the total number of bits allowed, and $n$ is the number of frames in the video sequence. Each

Although [4] admits an weighted distortion measure, an unweighted distortion measure is used herein.
admissible \( \tilde{Q} \) generates a point \( p = (\tilde{R}(\tilde{Q}), D(\tilde{Q})) \) where 
\[
\tilde{R}(\tilde{Q}) = (R_1(\tilde{Q}_1), R_2(\tilde{Q}_2), \ldots, R_n(\tilde{Q}_1, \ldots, \tilde{Q}_n)) \text{ and } \\
D(\tilde{Q}) = \sum_{k=1}^{n} D_k(\tilde{Q}_1, \ldots, \tilde{Q}_k).
\]
We define the set of all points \( p \in \mathbb{R}^{n+1} \) above as the operational distortion-rates set (ODRS). The operational rate-distortions set (ORDS) is similarly defined for the related problem where a constraint on the total distortion is specified and the rate of the sequence is to be minimized

\[
Q_{D_{tot}}^* = (Q_1^*, \ldots, Q_n^*) \quad (4)
\]

\[
= \arg\min_{Q=(Q_1, \ldots, Q_n)} \sum_{k=1}^{n} R_k(Q_1, \ldots, Q_k) \quad (5)
\]

subject to 
\[
D(\tilde{Q}) = \sum_{k=1}^{n} D_k(Q_1, \ldots, Q_k) \leq D_{tot} \quad (6)
\]

Here \( D_{tot} \) is the total distortion allowed.

If every frame in the sequence were coded independently, i.e. if \( D_k(\cdot) \) and \( R_k(\cdot) \) were functions of \( Q_k \) alone, (1-3) could be solved efficiently using Lagrangian optimization [2]. Lagrangian optimization relaxes the constrained problem to an unconstrained version by incorporating the constraint function into the objective via a “Lagrange” multiplier \( \lambda \geq 0 \).

\[
\tilde{Q}^* = \arg\min_{(Q_1, \ldots, Q_n)} \sum_{k=1}^{n} D_k(\cdot) + \lambda \sum_{k=1}^{n} R_k(\cdot) \quad (7)
\]

If an optimal solution to the unconstrained problem for a particular \( \lambda \geq 0 \) has rate \( R(\tilde{Q}) \) equal to \( R_{tot} \), it is also optimal for the original constrained problem [6]. In the independent case, the unconstrained problem (7) can be solved efficiently by separately minimizing each component in the sum.

\[
Q_k^* = \arg\min_{Q_k} [D_k(Q_k) + \lambda R_k(Q_k)] \quad (8)
\]

A solution to the constrained problem is then found by searching for the \( \lambda \) that gives the largest rate value \( R(\tilde{Q}) \) less than or equal to \( R_{tot} \).

The Lagrangian optimization technique was applied to the predictive coding problem in [4]. The optimal allocation of bits to frames in a predictively coded sequence is complicated by the fact that the distortion and rate functions of frames depend on the quantization parameter choices for prior encoded frames. In the predictive coding case it is no longer as easy to solve (7) due to the dependencies in the distortion and rate functions. Therefore, in the absence of additional structure, a full search over all possible quantization parameter vectors \( \tilde{Q} \) is necessary to solve the problem optimally. This search has complexity exponential in the prediction depth (the number of frames that a reference frame affects through prediction). That is, the encoder must be called an exponential number of times to evaluate the distortion and rate functions for all admissible \( \tilde{Q} \). This complexity is significantly reduced in [4] by pruning the solutions based on the observation that distortion-rate functions for predicted frames are usually monotonic in the fineness of the quantizers chosen for the reference frame. This paper demonstrates that multivariate distortion-rate data also possess an underlying convex structure.

An important fact highlighted in [4] is that unlike the independent case, data gathering in a predictive setting is computationally expensive. Distortion-rate information has to be specified only for \( \sum_k m_k \) points in the independent case. In the predictive case, it may have to be specified for up to \( \prod_k m_k \) points. Therefore, it is important to sample as few points as possible while trying to compute the optimal solution. An approach to reduce sampling is to build a model for the ODRS by using as few samples as possible. Such a model based approach for rate-distortion optimization was proposed in [5]. The distortion and rate functions are modeled by sampling a subset of quantization parameter values within each frame. The focus is on deriving fast practical algorithms and the optimality of the model solutions to the original problem is not investigated.

An alternate approach to Lagrangian optimization for the independent coding problem uses the generalized BFOS algorithm [7, 8]. It can be shown that all solutions on the vertices of the lower convex hull of achievable \( \left( \sum_k D_k, \sum R_k \right) \) pairs can be generated by sweeping \( \lambda \) from 0 to \( \infty \) in (7). These set of solutions are efficiently generated via the generalized BFOS [7, 8] algorithm by formulating the problem in terms of tree-pruning. A Steepest Descent (SD) method that reduces to a variant of the BFOS algorithm in the independent case was introduced for the predictive coding problem in [3]. This algorithm starts at a low rate solution and uses gradient information to construct solutions at higher rates. The procedure can be terminated once the desired rate is achieved. The SD method solves the problem without explicitly constructing a model of the distortion and rate functions as in [5]. A similar algorithm is used in [9] for the selection of JPEG quantization matrices.

**3. CONVEXITY MODEL**

The discrete set of points in the ODRS is defined to be convex if the points can be fit by a convex \((n+1)\)-dimensional surface \( D_S(R) = D_S(R_1, \ldots, R_n) \), where the \((n+1)^{th}\) coordinate of the ODRS is viewed as a dependent function of the first \(n\) coordinates. Similarly the ORDS is defined to be convex if the set of points can be fit by a convex surface \( R_S(\tilde{D}) = R_S(D_1, \ldots, D_n) \). The discussion will henceforth mostly refer to the ODRS. It is straightforward to extend the development to include the ORDS.

The convexity of the ODRS is tested experimentally by considering the Lower Convex Hull (LCH) of the points in the ODRS. For instance, Figure 1 shows the LCH of the
ODRS for a two frame video sequence, where the distortion of the sequence is plotted as a function of the rates of the individual frames. If every point in the ODRS lies on the LCH, the LCH is itself a convex surface that perfectly fits the data. If a point does not lie on the LCH, the deviation of the point from the LCH is computed as follows. Let \( x = ( R, D(R) ) \) be a point that does not lie on the LCH. Let \( y = ( R, D_{LCH}^C(R) ) \) be the point on the LCH directly below \( x \). We define the relative deviation from the LCH for point \( x \) as \( \Delta_x = \frac{D(R) - D_{LCH}^C(R)}{D_{LCH}^C(R)} \). The relative deviation is a normalized measure of the distance of a point from the LCH.

A Linear Programming based technique adapted from [10] is used to calculate the relative deviation for each point in the ODRS. The idea of the method is that if a point lies on the LCH, a hyperplane of some orientation passing through the point can be found such that the rest of the points in the ODRS lie above it. If a point does not lie on the LCH, the distortion of the point is reduced just until such a hyperplane can be found. This occurs exactly when the new point falls on the LCH. For each point in the ODRS a Linear Program (LP) is solved in order to compute the deviation of the point from the LCH. The LP for a point \( v = ( R_v, D(R_v) ) \) is given by

\[
d^*_v = \min d_v \\
\text{subject to } a_{0v} + \bar{a}^T R_v = D(R_v) - d_v \\
a_{0v} + \bar{a}^T R_w \leq D(R_w), \forall w \in \text{ODRS, } w \neq v
\]

The orientation and offset of the hyperplane are represented by \( \bar{a} \) and \( a_{0v} \), respectively. The optimal value \( d^*_v \) specifies the amount by which the distortion of the point \( v \) must be reduced in order to place the point on the LCH. The relative deviation for point \( v \) can be expressed as \( \Delta_v = d^*_v / (D(R_v) - d^*_v) \).

Table 1 shows the relative percentage deviation for four frame segments of standard SIF sequences encoded in an I-P-P frame-type structure using the motion compensation and mode selection algorithm of MPEG-2 TM5 [11]. Mean Squared Error (MSE) is used as the distortion measure at the frame-level. The encoder uses a non-linear quantization table where \( Q_1 \) and \( Q_2 \) take values in the admissible set \( \{1, \ldots, 31\} \). Mean Squared Error (MSE) is used as the distortion measure at the frame-level.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Points ( \notin \text{LCH} ) (%)</th>
<th>Mean ( \Delta ) (%)</th>
<th>Max ( \Delta ) (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile</td>
<td>6.63</td>
<td>1.46 ( \times 10^{-2} )</td>
<td>0.163</td>
</tr>
<tr>
<td>Foreman</td>
<td>7.50</td>
<td>1.98 ( \times 10^{-1} )</td>
<td>2.25</td>
</tr>
<tr>
<td>Football</td>
<td>6.99</td>
<td>7.28 ( \times 10^{-2} )</td>
<td>0.609</td>
</tr>
<tr>
<td>Garden</td>
<td>4.72</td>
<td>1.52 ( \times 10^{-2} )</td>
<td>0.232</td>
</tr>
<tr>
<td>Tennis</td>
<td>11.26</td>
<td>9.25 ( \times 10^{-1} )</td>
<td>4.85</td>
</tr>
</tbody>
</table>

Table 1 shows that close to 90% of the points lie exactly on the LCH of the ODRS. Moreover, the average relative deviation of the points that do not lie on the LCH is less than 1%. Also, the maximum relative deviation of any single point is less than 5%. The same holds for the ORDS as well, as shown in Table 2, except for the Tennis sequence which exhibits slightly higher deviation values. Hence, both the ODRS and the ORDS are predominantly convex in the sense that the relative deviation from the LCH is small. Note that the points in the ODRS are computed solely for the purpose of validating the convexity model. The actual SD algorithm (Section 4) samples significantly fewer points by assuming a convex ODRS.

### 4. STEEPEST DESCENT ALGORITHM

Assuming an underlying smooth structure for the discrete ODRS allows an easy derivation of optimality conditions. Therefore, a continuous version of the problem (1-3) is analyzed and the resulting optimality conditions are used to motivate an SD algorithm for the discrete case. Consider
replacing (1-3) by the continuous problem

\[ \hat{Q}_{R_{tot}}^* = \left( Q_1^*, \ldots, Q_n^* \right) \]

\[ = \arg \min_{Q=(Q_1, \ldots, Q_n)} D_S(\hat{R}(\bar{Q})) \]

subject to \( R(\bar{Q}) = \sum_{k=1}^{n} R_k(Q_1, \ldots, Q_k) \leq R_{tot} \)

where \( \hat{R}(\bar{Q}) \) is as defined in section 2. Assume that \( D_S(\hat{R}) \) is a smooth, strictly convex, monotonic decreasing function. Further assume that \( R_k(\cdot) \) are smooth functions of their arguments and that \( Q_k \) are continuous variables. It can be shown that strict convexity guarantees a unique solution to (12-14) for each total rate \( R_{tot} \). Ignoring non-negativity constraints on the bits allocated to each frame, the optimal solution occurs at the point where the partial derivatives are equal, i.e. when \( \frac{\partial D_S}{\partial R_i} = \frac{\partial D_S}{\partial R_j} = -\lambda, 1 \leq i, j \leq n \). The Lagrange multiplier \( \lambda \geq 0 \) depends on \( R_{tot} \). Since \( \frac{\partial R}{\partial R_i} = 1 \), \( \frac{\partial D_S}{\partial R_i} = \frac{\partial D_S}{\partial R_j} = -\lambda \). It can be shown under suitable assumptions that if \( \frac{\partial D_S}{\partial Q_j} / \frac{\partial R}{\partial Q_j} = -\lambda \) at \( \bar{Q} \), then \( \bar{Q} \) is an optimal solution to (12-14). The Coordinate Steepest Descent algorithm presented in [3] attempts to equalize a discrete version of the last expression, namely \( \frac{D(Q+\Delta Q_i) - D(Q)}{R(Q+\Delta Q_i) - R(Q)} \). A Coordinate SD algorithm that solves (4-6) can be motivated in the same manner. A similar algorithm is used in [9] for the selection of JPEG quantization matrices. The Coordinate SD algorithm is summarized below.

### 4.1. Coordinate Steepest Descent

**Definition 1** Let \( e_j \) denote the \( n \)-vector with a 1 in the \( j \)th position and 0 elsewhere. The slope in the \( j \)th direction at \( \bar{Q} \) is defined as \( s_j(\bar{Q}) = \frac{-D(Q-e_j) - D(Q)}{R(Q-e_j) - R(Q)} \).

**Coordinate SD Algorithm**

1. Set all components of the quantization parameter vector \( Q \) to their maximum values \( m_i \). Include indices \( 1, \ldots, n \) in the active list of coordinate indices that can be reduced.

2. Remove from the active list indices for which the corresponding coordinate value cannot be reduced or where the reduction by one would cause a rate constraint violation. Stop if the active list is empty.

3. Let \( j \) be an index that has the largest value of \( s_j(\bar{Q}) \) among all active indices. Set \( \bar{Q} \leftarrow \bar{Q} - e_j \).

4. Repeat from step 2.

The Coordinate SD algorithm exploits convexity to sample a number of points that is at most quadratic in the prediction depth [3]. Step 3 of the algorithm computes the index \( j \) that provides the largest ratio of distortion decrement per unit rate increment for a reduction in the \( j \)th coordinate \( Q_j \) by a single step. This gives information about how the change in parameter allocation within a frame affects the overall distortion and rate of the sequence as a whole. Starting at the optimal allocation at the lowest total rate (which is assumed to be the allocation with all the coordinates of the quantization parameter vector set to their highest values) at Step 1, allocations at higher rates are recursively derived until all the available rate is spent. Each computed allocation is nearly optimal for the corresponding rate.

The Coordinate SD algorithm reduces to a variant of the BFOS algorithm in the independent case. In the BFOS algorithm each index can be reduced by any (feasible) number of steps in order to handle non-convexities in the ODRS. In the predictive case, however, non-convexities cannot always be detected coordinate-wise. Therefore, the SD algorithm depends on the data being structured in a suitable way. Like the BFOS algorithm, the Coordinate SD algorithm does not backtrack, i.e. the value of \( Q_j \) is never increased during the course of the algorithm. Convexity alone is not enough to preclude backtracking. That it is unnecessary to backtrack is a consequence of an additive separability property exhibited by the ORDS. This property will be addressed by future work.

Ten frames of the Tennis sequence are encoded in an I B B P B P B P B P frame-type structure using the motion compensation and mode selection algorithm of MPEG-2 TM5. Each \( Q_i \), \( i = 1, \ldots, 10 \), is allowed to take values only in the set \{2, 5, 8, 13, 21, 31\}. Figure 2 shows a PSNR versus rate plot comparing Coordinate SD solutions with optimal solutions computed using brute force search. The SD algorithm (also constrained to use quantization parameter values in the set \{2, 5, 8, 13, 21, 31\}) samples at most 500 \((5 \times 10^2)\) of the possible 60466176 \((6^{10})\) points, and gives 50 nearly-optimal solutions spanning the range of rates.

### 4.2. Long-Step Steepest Descent

A faster SD algorithm can be derived by changing more than one coordinate of the quantization parameter vector at the same time and by taking longer steps in the coordinate values. The following version actually makes use of the slope values to derive a direction of descent. Step 3 in the Coordinate SD algorithm is changed to \( \bar{Q} \leftarrow \bar{Q} - [\alpha \sum_{j=1}^{n} s_j e_j] \), where \( \alpha \) is a step-size scaling parameter and \([\cdot]\) denotes rounding to the nearest integer. The vector \( \bar{Q} \) is reduced in the direction of the resultant vector \( \sum_{j=1}^{n} s_j e_j \). The length

<table>
<thead>
<tr>
<th>Sequence</th>
<th>points / LCH (%)</th>
<th>Mean Δ (%)</th>
<th>Max Δ (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile</td>
<td>5.35</td>
<td>5.68 x 10^{-2}</td>
<td>0.583</td>
</tr>
<tr>
<td>Foreman</td>
<td>8.21</td>
<td>1.83 x 10^{-1}</td>
<td>1.14</td>
</tr>
<tr>
<td>Football</td>
<td>5.37</td>
<td>2.28 x 10^{-2}</td>
<td>0.496</td>
</tr>
<tr>
<td>Garden</td>
<td>7.34</td>
<td>9.57 x 10^{-4}</td>
<td>0.584</td>
</tr>
<tr>
<td>Tennis</td>
<td>14.6</td>
<td>1.44</td>
<td>17.4</td>
</tr>
</tbody>
</table>
of the resultant vector is controlled using $\alpha$. For the experiments, $\alpha$ is set to $\alpha_{\max}$, which is chosen such that the largest component of the resultant vector equals 4. This choice of $\alpha$ gave consistent results over a wide range of sequences. Scaling $\alpha$ from 0 to $\alpha_{\max}$ produces intermediate rate solutions. For the MPEG-2 encoding of a GOP of standard CIF sequences using TM5 motion compensation and mode selection, the Long-Step SD algorithm resulted in a speed-up of 30 to 50 times. For all tested sequences encoded at rates less than 1.5 Mbps, the PSNR for the Long-Step SD algorithm was never worse than 0.2 dB as compared to the PSNR of the Coordinate SD algorithm.

4.3. Changing the Encoder

Since the SD algorithm treats the encoder as a black box, the encoder can be changed. Figure 3 shows the SD solution for two different MPEG-2 encoders; one using the motion compensation and mode selection algorithm of TM5 and the other using the Rate-Distortion optimized motion and mode selection algorithm proposed in [1]. Both curves are generated using the Long-step SD algorithm described above. The plot clearly shows that improving the encoder leads to higher PSNR at all rates. The R-D optimized motion and mode selection version achieves lower rate solutions. For the MPEG-2 encoding of a GOP of standard SIF sequences using TM5 motion compensation and mode selection, the Long-Step SD algorithm resulted in a speed-up of 30 to 50 times. For all tested sequences encoded at rates less than 1.5 Mbps, the PSNR for the Long-Step SD algorithm was never worse than 0.2 dB as compared to the PSNR of the Coordinate SD algorithm.

5. CONCLUSION

This paper shows that the overall mean squared error distortion of a video sequence is predominantly a convex function of the number of bits allocated to each frame in the sequence. This property is used to motivate an intuitive steepest descent (SD) algorithm for bit allocation. Both the data model as well as the algorithm are validated experimentally. The SD method produces near-optimal solutions in practice and makes a number of calls to the encoder at most quadratic in the prediction depth. The SD algorithm can be used to generate solutions for multiple rates at minimal additional cost. The resulting solutions can be used for benchmarking bit allocation algorithms, and for encoding sequences for off-line applications.

6. REFERENCES


