In Proc. of IS&T/SPIE, Digital Video Compression '95, San Jose, CA, Feb. 1995

Forward–adaptive quantization with optimal overhead cost for image and video coding with applications to MPEG video coders

Antonio Ortega

Dept. of EE-Systems and Signal and Image Processing Institute
University of Southern California
Los Angeles, CA

Kannan Ramchandran

Beckman Institute
University of Illinois
Urbana, IL

ABSTRACT

We address the problem of optimal forward–adaptive quantization in the video and image coding framework. In this framework, as is consistent with that of most practical coders like MPEG, the encoder has the capability of changing the quantizer periodically (e.g. at a macroblock interval in MPEG). In this paper, we formulate an optimal strategy, based on dynamic programming, for updating the quantizer choice for coding an image or video signal. While in some coding environments the overhead needed to specify the quantizer used by each block is equal for every choice of quantizer, in other situations (e.g. MPEG) the overhead cost is higher if the quantizer changes from one block to the next. We concentrate on the latter case which will be more likely encountered in situations where the overhead represents a significant fraction of the overall rate, as can be the case if a low bit rate is used (e.g. error frames in a typical motion-compensated video coder). We provide empirical evidence of the performance gain that can be obtained when applying our optimal algorithm to typical motion-compensated prediction error frames in MPEG, showing how the popular Viterbi algorithm can be used to find the optimal solution.

Keywords: Image compression, video compression, MPEG, overhead, bit allocation, quantization.

1 Introduction

In video and image coding systems, the ability to intelligently adapt the quantization strategy in order to best match the varying input signal characteristics is essential to achieving high performance. This leads to the paradigm of an adaptive quantization framework. In forward-adaptive systems, the encoder "does all the work" by determining the optimal adaptation policy and broadcasting it periodically to the decoder, which can therefore be kept simple. The price paid for this decoder simplicity is the increased

complexity at the encoder as well as the cost of explicitly conveying the updated quantizer parameters via side-information to the decoder. In backward-adaptation systems like pel-recursive coders [1], the decoder "infers" the adaptation parameters without the encoder needing to send any side-information [2]. The price paid for this zero side-information is, of course, the increased decoding complexity.

In a number of image and video coding applications (e.g. broadcast, CD-ROM storage or centralized processing applications like remote-data-base retrieval), the decoder complexity needs to be kept small, although encoder complexity need not. Backward-adaptive systems are obviously inappropriate for such applications. Motivated by this, in this paper, we focus on optimizing forward-adaptive quantization systems, and show how to apply our algorithm to the popular MPEG [3] video coder. We will address the optimal quantizer choice for coding an image or video sequence, *including the overhead* of updating the quantization parameters in our analysis.

Note that optimal quantization of independently coded signal units (a topic which has been studied in the literature [4]), where each signal unit is quantized optimally based on a discrete quantizer set choice, can be construed as a special case of adaptive quantization, where the overhead cost of sending the quantization choice is either ignored (as it is negligible in comparison to the actual cost of coding the signal) or is included independently for each signal unit (see Figure 1(a)). Typical overhead schemes for these frameworks involve either a constant-cost for each quantizer in the set or a fixed tree-map (for tree-structured quantizers) as in [5].

In this work, we concern ourselves with coding models where the quantizer choice cost is *not* independent of the quantizer choices for preceding signal units, leading to a more complex optimization framework. As an example, the MPEG video coder falls under this framework, as it permits periodic quantization updates (changing the MQUANT or QP choices [3] every macroblock), with a *dependent* overhead cost associated with each such choice, depending on the choice made for the previous interval (macroblock). This dependent framework is motivated by the slowly varying nature of the image or video characteristics (except for edges or object boundaries in images and video signals), which make it highly probable that neighboring signal units will have identical quantizers. Thus, it is more efficient to differentially encode the quantizer choices, assigning a lower cost to the quantizer choice which is the same as the one in the previous signal unit (e.g. MPEG assigns a one-bit code to the quantizer choice of a macroblock if it matches the choice made for the previous macroblock, else it assigns a 6-bit cost for changing the choice, assuming a 32-quantizer choice set, see Figure 1(c)). Here, we will consider the more general scheme of the overhead cost being dependent on the quantizer choice as well as its current runlength (see Figure 1(b)). In this regard, our problem becomes an optimal segmentation or parsing problem, where the segment lengths are the runlengths of the quantizer choices. Note that optimal parsing using weighted universal VQ codebooks has been addressed in [6], while optimal time-frequency segmentation using adaptive wavelet-packet decompositions has been tackled in [7]. This paper will also address an important special case involving the MPEG coder, which leads to a formulation using the Viterbi Algorithm with much reduced computational complexity.

We show applications of our optimal algorithm to the well-known MPEG coding framework and show the gain in performance attainable over Lagrange-multiplier based independently-optimized methods. It must be pointed out that this performance gain is significant only if the overhead cost forms a significant contribution to the total cost. Thus, for low bit rate coding applications or when coding prediction error signals (like the B and P frames in MPEG), these gains can be non-negligible. Further, our algorithm can serve the important role of providing an upper-bound benchmark with which to assess the performance of cheaper heuristics and other ad hoc methods.

Another application of our optimal algorithm of this paper is the growing interest in higher compression image coding techniques for a future "second generation JPEG". While for complexity reasons the JPEG standard assigns a single quantization parameter to all the blocks in a frame, future standards will almost certainly resort to blockwise adaptive quantization (assuming a block-based scheme is chosen). Furthermore, future algorithms may allow more coding parameters to be adapted on block per block basis. For example, in a typical DCT based scheme one would be able to adapt not only the quantization scale but also the quantization matrix itself. These added choices, while increasing the overhead, may provide performance gains, especially if coupled with an allocation strategy such as that presented here. We emphasize thus that while our experimental results apply to MPEG coders our algorithm is applicable to other coding scenarios.

The rest of the paper is organized as follows. Section 2 outlines the Lagrangian optimization that will be used and analyzes the problem. Section 3 addresses an optimal dynamic-programming based algorithm using the Viterbi Algorithm to solve the problem. Section 4 shows an application involving an MPEG-like coder.

2 Encoding of Overhead Information

In this section we start by reviewing several alternative ways of transmitting overhead information as well as the type of applications where each is useful. As a lead-in to the more complex case where the overhead cost is considered jointly with the rate–distortion cost of the coded signal units or blocks, we will first tackle the case where the overhead cost is "free" (or negligible, in practice) or where the overhead cost is fixed and independent of the quantizer choice for each block.

2.1 Independent case: the constant slope condition

We will briefly summarize the optimal algorithm for the independent case, i.e. what is the optimal policy to assign a quantizer choice (from a discrete admissible set) to independently coded signal units assuming a zero cost (this can be trivially extended to a nonzero fixed cost as well) for making updates? This problem has been addressed extensively in the literature in the context of optimal bit allocation in independently coded frameworks like subband coding [8,4]. This corresponds to encoding the overhead as in Figure 1(a).

The classical rate–distortion optimal bit allocation problem consists of minimizing the average distortion D of a collection of signal elements or blocks subject to a total bit rate constraint R_{budget} for all blocks. Let us without loss of generality consider the two unit (or block) case, where $\{Q_1, R_1(Q_1), D_1(Q_1)\}$ and $\{Q_2, R_2(Q_2), D_2(Q_2)\}$ refer to the quantizer, distortion and bit rate of each unit respectively, the independent allocation problem is:

$$\min_{Q_1,Q_2} \; [D_1(Q_1) + D_2(Q_2)] \tag{1}$$

$$\text{such that } R_1(Q_1) + R_2(Q_2) \leq R_{budget}. \tag{2}$$

The "hard" constrained optimization problem of (1), (2) can be solved by being converted to an "easy" equivalent unconstrained problem. This is done by "merging" rate and distortion through the Lagrange multiplier $\lambda \geq 0$ [4], and finding the minimum Lagrangian cost $J_i(Q_i) = \min_{Q_i}[D_i(Q_i) + \lambda R_i(Q_i)]$ for $i = 1, 2$. The search for the optimal R–D operating points for each signal block can be done *independently*, for the fixed quality "slope" λ (which trades off distortion for rate) because at R–D optimality, all blocks *must operate at a constant slope point λ on their R–D curves* [4,9]. The desired optimal constant slope value λ^* is not known *a priori* and depends on the particular target budget or quality constraint, but can be obtained via a fast convex search [9].

To summarize, in the case of the overhead being *independently* considered or not considered at all, the optimal quantization choice is given by the "constant slopes" condition, where the optimal quantizer choice for each block is given by the one which "lives" at a rate–distortion tradeoff of λ^* (corresponding to the desired bit budget criterion) on its operational rate–distortion characteristic. Note that in this paper, as we consider the more general case where the overhead needed to convey the quantizer choice information is *not* independent of the actual quantizer chosen, the above method is not applicable.

2.2 Dependent case

Now we turn to the case at hand, where the overhead cost is dependent on the quantizer choice. The first parameter to consider is the size of the block for which overhead is transmitted (e.g. blocks or macroblocks in the MPEG case). Obviously, the larger the block we are considering, the cheaper it will be to send the overhead information. Indeed, the result of optimizing the overhead information is to identify block sizes for which the overhead information pays for itself. When considering tree–structured quantizers like TSVQ [10] or quadtree segmentations [5] involving split/merge decisions, there is an extra cost added every time a decision to split is made. The optimization algorithm decides when that extra cost is useful and "pays for itself". In this work we will not consider tree-structured data structures, but rather more general sequential ways of representing the overhead, formulating the optimal way of fusing the overhead choice with the optimal quantization choice for a sequence of signal or image blocks.

In a sequential overhead scheme, we first define an order in which to scan the input blocks, e.g. a row order scan as in MPEG [3] or a Peano scan as in [6]. The sequential model amortizes the cost of using a particular quantizer over the number of consecutive blocks over which it is used. This biases the decision for every block quantization choice in favor of what has been used in previous blocks, making a decision to change this choice only if the new quantizer is worth the cost (in the rate–distortion sense) it incurs for "breaking up" the previous run. The precise relationship of the overhead cost in bits with respect to the quantizer runlengths are assumed to be known, either based on realistic models or through training as in [6].

A simpler "first-order" overhead model would be to have the overhead cost of using a particular quantizer choice for the current block depend only on whether or not this choice differs from that for the previous block. Thus, a single bit would indicate whether or not a *new* quantizer (i.e. different from that used for the previous block) is used, followed by more bits in case there has been a change, to resolve the uncertainty in the new quantizer choice by specifying the new quantizer index. See Figure 1(c). The obvious advantage of this approach is its simplicity, and the fact that it is not heavily reliant on training models derived from specific training data. The approach used in MPEG-2 follows this rule, where a fixed-rate code is used to assign the index of the new quantizer choice.

An alternative way of encoding the overhead would be to resort to runlength codes. Given prior

assumptions on how often quantizers are changed, and thus on the expected lengths of runs of consecutive blocks using the same quantizer, we can have predetermined codewords associated with the runlengths. Thus encoding the overhead would be done by sending, with the first block of a run of blocks using the same quantizer, the choice of quantizer as well as the corresponding runlength code (see Figure 1(b)). Such an approach is used in [6] where, additionally, the procedure for optimally finding the runlengths codes is also described.

Because entropy coding will be used to encode the runlengths, this approach has the disadvantage of being quality-dependent: for a high quality, high rate encoding, there is no incentive to using the same quantizer for consecutive blocks, since the overhead represents a minimal fraction of the total rate. Therefore we can expect shorter average runlengths (and thus shorter runlengths will be assigned short codewords). Conversely, at low rates it might be worthwhile to save the overhead bits and thus runlengths will be longer on average (and thus relatively long runlengths may shorter codelengths). The "MPEG method", on the other hand, relies on a simpler model which just assumes that consecutive blocks are more likely to use the same quantizer (hence the shorter codeword assigned) than different ones. This assumption seems to apply in general and thus the performance of this simpler approach would not depend on the quality level at which the coder is operated. Obviously a runlength approach designed for a specific quality would always be more efficient at that specific target.

3 Problem Statement and Solution

3.1 General case

Given N signal units $\{x_1, x_2, \ldots, x_N\}$, an M-quantizer set $Q = \{q_1, q_2, \ldots, q_M\}$ with which to code each signal, we have an operational rate-distortion performance for coding the ith unit x_i using the jth quantizer q_j given by $(R(\hat{x}_i(q_j)), D(\hat{x}_i(q_j)))$. The overhead cost of using the jth quantizer for the ith unit, given that using the jth quantizer for the ith unit would give it a runlength of t (i.e. the jth quantizer is being used for signal units $i-t+1, i-t+2, \ldots, i-1$) is $R_{overhead,i}(j,t)$, we want to find the optimal quantizer choice sequence $\{q^{(1)}, q^{(2)}, \ldots, q^{(N)}\}$ for the entire sequence.

In the algorithm we will proceed sequentially to find the optimal segmentation under the above conditions, where our goal is to minimize the Lagrangian cost with respect to a desired rate–distortion tradeoff factor given by the Lagrange multiplier $\lambda \geq 0$ (see Section 2.1). We will denote by J_i^* the minimum cost solution where a new runlength is started at stage i.

We initialize $J_0^* = 0$ and then define $\Delta J_{k,i}$ as the minimum cost solution which has a runlength between blocks k and i, i.e. where blocks k through i use the same quantizer. We have thus:

$$\Delta J_{k,i} = \min_{q_j \in Q} [\sum_{l=k}^{i} D(\hat{x}_l(q_j)) + \lambda R(\hat{x}_l(q_j)) + \lambda R_{overhead,i}(j, i-k+1)], \tag{3}$$

$\forall k < i$ and for $k = 1, 2, \ldots, N-1$.

Then for $k = 1, \ldots, N-1$, form the Dynamic Programming (DP) [11] recursion:

$$J_i^* = \min_k [J_{k,i}] = \min_{0 \leq k < i} [J_k^* + \Delta J_{k+1,i}] \tag{4}$$

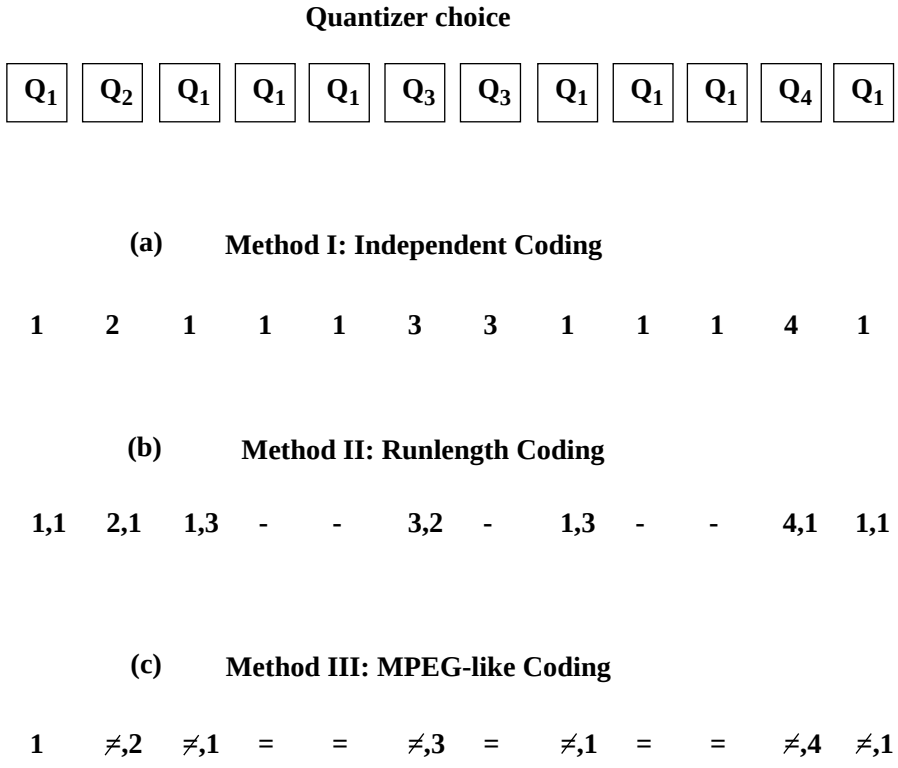

Figure 1: Three approaches to encode the segmentation overhead. (a) The overhead for each block is coded independently. This would be a good method when the overhead is negligible with respect to the rate per block. (b) The overhead is coded using runlength codes, i.e. quantizer choice and number of consecutive blocks with same assignment. (c) For every block a code indicates whether the quantizer choice is the same as in the previous block. If different the new choice is transmitted. This is the method used in MPEG.

where we choose the best way of reaching the end of a runlength at stage i by comparing all combinations of the best runlength ending at stage k and the "cheapest" runlength from $k+1$ to i. See Figure 2. The predecessor of i in the segmentation is thus:

$$\text{predecessor}\,(i) = \arg\min_k(J_{k,i}). \tag{5}$$

Once all the J_i^* have been found for $i = 1, \cdots, N$ the optimal segmentation is (backtracking from the end):

$$[\ldots, \text{predecessor}\,(\text{predecessor}\,(\text{N})), \text{predecessor}\,(\text{N}), \text{N}]. \tag{6}$$

Once the R-D data is known and if M is the cardinality of the set of quantizers we have a complexity on the order $\mathcal{O}(N^2M)$ since at each of the N stages i we have to perform Q comparisons for each the preceding $i-1$ stages.

Note that in order to be optimal we have to always keep the best segmentations which result in a new runlength for each node i. In practice though, there will likely be a maximum runlength, say L, and thus we would only keep $J_{i-L}, J_{i-L+1}, \ldots, J_{i-1}$ when considering node i. The complexity would thus be $\mathcal{O}(NLM)$, although in general this would be suboptimal unless $L = N$.

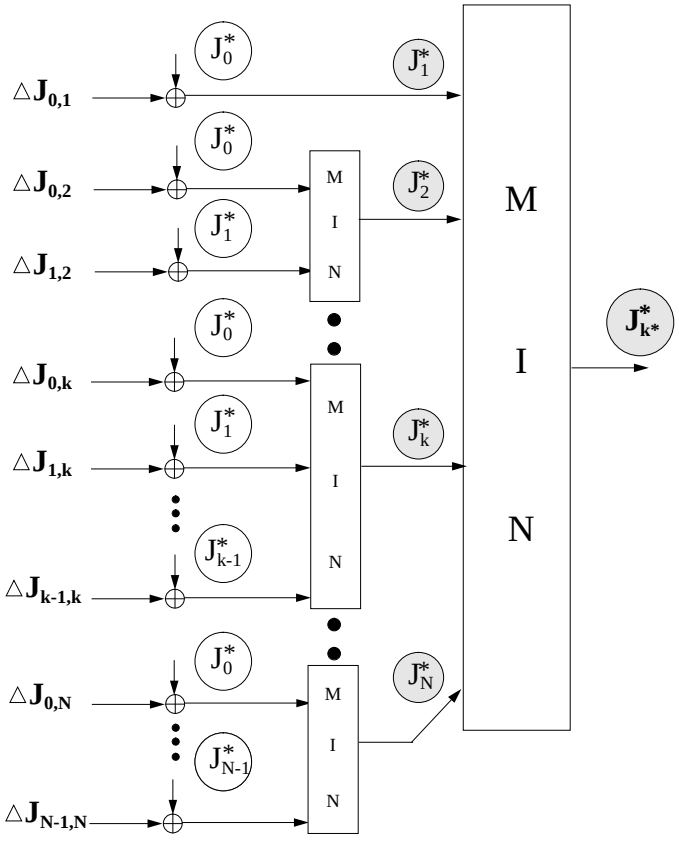

Figure 2: General dynamic programming segmentation algorithm. Each of the J_i^*'s represents the cost of the best segmentation up to block i such that the last run length finishes at block i.

3.2 Overhead in the MPEG case

We now study the particular case of the MPEG video coder, where the overhead is not sent as a runlength (see Figure 1). In this case we have that:

$$R_{overhead,i}(t) = \begin{cases} R_1 \text{ bits} & \text{if } t = 1 \\ R_2 \text{ bits} & \text{if } t > 1 \end{cases} \qquad \forall i, \tag{7}$$

where typically we will have $R_2 = 1$ and $R_1 = 1 + \lceil \log_2(|Q| - 1) \rceil$.

(7) reflects the specific case of MPEG, where the overhead quantizer cost for a given block depends

only on the choice for the previous block. When (7) applies, the DP recursion can be made into a sequential trellis search. The optimal segmentation can be found as follows. Our trellis has N stages, each corresponding to an input block x_i. At each stage, we have M possible states corresponding to the M quantization choices. Then we populate each of the nodes at stage i, $\{x_i, j\}_{j=1}^{M}$ with costs:

$$J_\lambda(\hat{x}_i(j)) = D(\hat{x}_i(j)) + \lambda R(\hat{x}_i(j)), \tag{8}$$

and we populate each of the branches with the cost $\Delta J_{\lambda,overhead}(j_1, j_2)$ such that for the branch linking node (x_{i-1}, j_1) to (x_i, j_2) the cost is

$$\Delta J_{\lambda,overhead}(j_1, j_2) = \begin{cases} \lambda R_1 \text{ bits} & \text{if } j_1 \neq j_2 \\ \lambda R_2 \text{ bits} & \text{if } j_1 = j_2 \end{cases} \tag{9}$$

To find the optimal solution (i.e. the path with the minimum total cost) we use the Viterbi Algorithm (VA). The algorithm operates sequentially, starting at the first stage and at each node of stage i it consists of keeping the best of all branches arriving at that node. See Figure 3. Thus at node (x_i, j_2) we keep the path arriving from node (x_{i-1}, j_1) where we have

$$j_1 = \arg \min_{j=1,\ldots,M} \{J_\lambda(x_{i-1}(j)) + \Delta J_{\lambda,overhead}(j, j_2)\} \tag{10}$$

Note that with this type of overhead we need only keep M candidate solutions at any given time. Furthermore, the complexity of the search is $\mathcal{O}(M^2 N)$ rather than $\mathcal{O}(N^2 M)$ in the more general runlength case, which can be significant for $M \ll N$, as is normally the case.

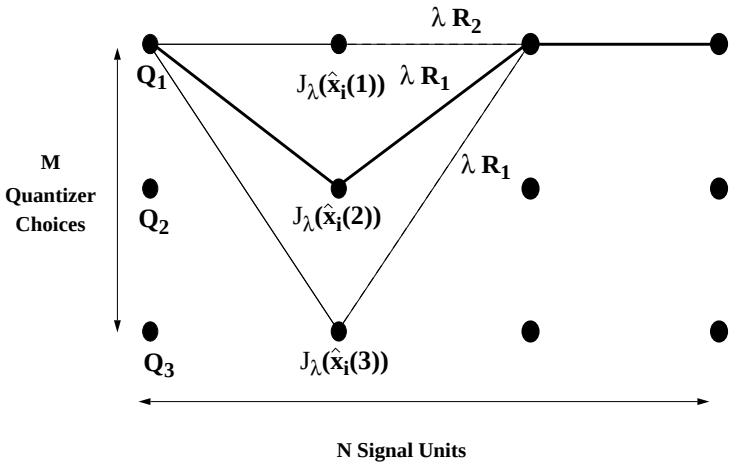

Figure 3: For the special case of non runlength overhead the VA can be used. The overhead cost depends only on the choice for the previous block. Thus branches which maintain the same quantizer have cost λR_1 and branches which change quantizer have cost λR_2. At every stage only the minimum cost path into each node is kept.

4 Experimental results

In this section we present an example of application of the algorithm of Section 3.2 to a DCT-based encoder. Our example is motivated by the so called MQUANT parameter within the MPEG-1 [3] and

MPEG-2 [12] standards. In MPEG, a frame is divided into slices, or groups of consecutive (following a row-wise scan) macroblocks, each of which is assigned some quantization step. Each of the macroblocks within the slice is in turn composed of 8 by 8 blocks of pixels (typically 4 luminance blocks and 1 block of each of the 2 luminance components for a 4:2:0 sampling format). Even though all the macroblocks within a slice are assigned the same quantizer, the MPEG syntax allows for change of the quantization step size at any macroblock within a slice by setting the MQUANT parameter (then the new quantization scale is used for all remaining macroblocks in the slice or until a new MQUANT is encountered). Thus, our algorithm would provide a means of deciding in an optimal fashion when to switch quantization choices and what to change them to, taking into account the cost associated with each change.

Note that our algorithm is of interest at low bit rates or for low energy images such as difference, P, B frames within MPEG. Thus, in our experiment we choose as our target image a predicted, P, frame obtained using MPEG-1 on the MIT sequence. We choose a set of 32 quantization scales within JPEG and perform blockwise adaptive quantization (only luminance blocks are considered). To encode the overhead we employ $R_1 = 1$ bits and $R_2 = 6$ bits. Note that we are considering a simplified version of the MPEG syntax. In the general case there is a single variable length code that describes the type of the macroblock. This code includes information on whether the macroblock was motion compensated, coded intra, as well the MQUANT parameter. However, we assume that the other choices (Motion/No Motion/Intra, e.g.) are made independently of the choice on MQUANT and thus for a given choice of the other parameters, we can assign the corresponding "differential" cost to having an MQUANT or not. This cost will depend on the other parameters but for simplicity we assume only two choices R_1 and R_2. Our algorithm can be used in the more general case as well.

Our results (see Fig. 4) compare the optimal allocation using the overhead assignment algorithm and that obtained using the simpler Lagrangian optimization [4] approach where the overhead cost is not included in the optimization but the same scheme for encoding the overhead is used. Our results indicate that including the overhead in the cost is useful at very low bit rates (although for low rates using several quantizers becomes in itself too expensive and a simple, single quantizer JPEG would do better). We use this result as proof of concept, while we leave for future work the application of our algorithm to situations where the gain of using optimal overhead allocation would be more substantial (e.g. when blocksizes smaller than 8x8 are used).

5 REFERENCES

[1] A. Netravali and J. Robbins, "Motion compensated television coding: Part I," *Bell Syst. Tech. J.*, vol. 58, pp. 631–670, Mar. 1979.

[2] A. Ortega and M. Vetterli, "Adaptive quantization without side information," in *Int'l Conf. on Image Proc., ICIP'94*, vol. 3, (Austin, Texas), pp. 856–860, Nov. 1994.

[3] D. LeGall, "MPEG: a video compression standard for multimedia applications," *Communications of the ACM*, vol. 34, pp. 46–58, Apr. 1991.

[4] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. on Signal Proc.*, vol. 36, pp. 1445–1453, Sept. 1988.

[5] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video," *IEEE Trans. on Image Proc.*, vol. 3, pp. 327–331, May 1994.

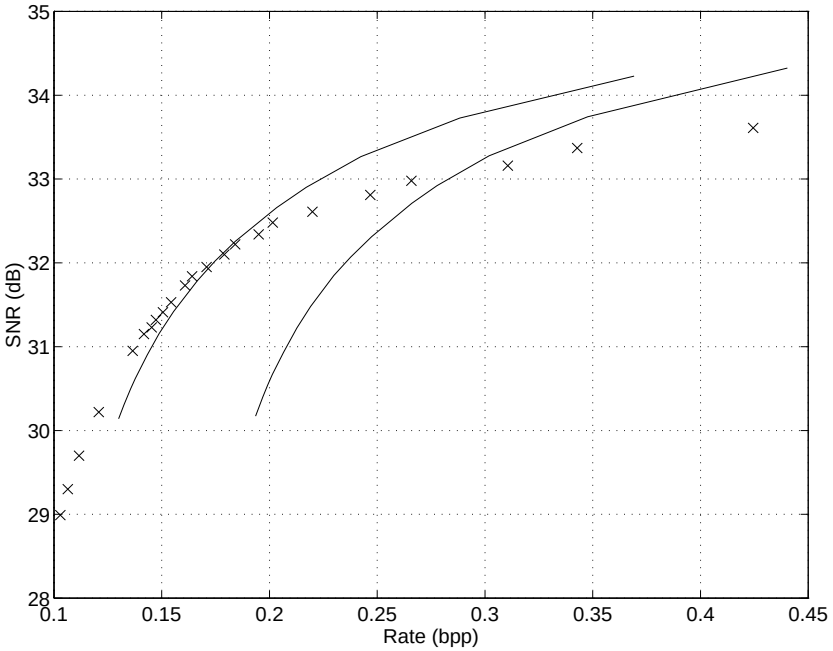

Figure 4: Comparison of the optimal allocation considering overhead (top curve) with the best independent allocation, i.e. where overhead is not considered in the allocation, (bottom curve). The "x" represent points obtained using a single quantization, so that overhead is not required

[6] M. Effros, P. A. Chou, and R. M. Gray, "Variable dimension weighted universal vector quantization and noiseless coding," in *Proc. of the Data Compression Conference, DCC'94*, (Snowbird, UT), pp. 2–11, Mar. 1994.

[7] Z. Xiong, C. Herley, K. Ramchandran, and M. Orchard, "Optimal segmentations for time-varying wavelet packets," in *Workshop on time-frequency analysis*, Oct. 1994.

[8] N. S. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Englewood Cliffs, New Jersey: Prentice-Hall, 1984.

[9] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate–distortion sense," *IEEE Trans. on Image Proc.*, vol. 2, Apr. 1993.

[10] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree-structured source coding and modeling," *IEEE Trans. on Info. Th.*, vol. IT-35, pp. 299–315, Mar. 1986.

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

[12] Inform. Technology - Generic Coding of Moving Pictures and Associated Audio, ITU Draft Rec. H.262, ISO/IEC 13818-2, Mar. 1994.