EFFICIENT FRAME VECTOR SELECTION BASED ON ORDERED SETS

Tom Ryen[1], *Guido M. Schuster*[2], *and Aggelos K. Katsaggelos*[3]

[1] Stavanger University College, Department of Electrical and Computer Engineering,
P.O.Box 2557, 4091 Stavanger, Norway. E-mail: tom.ryen@tn.his.no
[2] HSR Hochschule für Technik Rapperswil, Abteilung Elektrotechnik,
Oberseestrasse 10, 8640 Rapperswil, Switzerland. E-mail: guido.schuster@hsr.ch
[3] Northwestern University, Department of Electrical and Computer Engineering,
Evanston, Illinois 60208-3118, USA. E-mail: aggk@ece.nwu.edu

ABSTRACT

The problem of finding the optimal set of quantized coefficients for a frame-based encoded signal is known to be of very high complexity. This paper presents an efficient method of finding the operational Rate-Distortion (RD) optimal set of coefficients. The major complexity reduction lies in the reformulation of the original RD-tradeoff problem, where a new set of coefficients is used as decision variables. These coefficients are connected to the *orthogonalization* of the set of selected frame vectors and not to the frame vectors themselves. By organizing all possible solutions as nodes in a solution tree, we use complexity saving techniques to find the optimal solution in an even more efficient way. Using an ordered vector selection process, the complexity can be again significantly reduced and efficient Run-length encoding becomes feasible. Contrary to the original problem, the new problem can be solved optimally in a reasonable amount of time.

1. INTRODUCTION

The use of frame-based coding has been given attention in recent years in topics like lossy compression, sparse signal representation and classification. A frame, or an *overcomplete dictionary*, is a redundant set of column vectors. Consider a one-dimensional signal, $\mathbf{x}$, consisting of NL samples, divided in L blocks of length N. (We demonstrate the concepts with 1D signals for convenience in this version of the paper, but we will show images in the final paper.) Block l, $\mathbf{x}_l \in \mathbb{R}^N$, can be seen as a vector of length N. In frame-based coding, the purpose is to find a best possible approximation to $\mathbf{x}_l$, $\tilde{\mathbf{x}}_l$, as a linear combination of a *small number* of frame vectors. Let $\mathbf{F}$ denote an $N \times K$ matrix whose columns, $\{\mathbf{f}_k\}$, $k = 1, \ldots, K$, constitute a frame. Let $\tilde{\mathbf{w}}_l$ be a vector of length K, where element k, $\tilde{w}_{l,k}$, is the quantized coefficient for frame vector $\mathbf{f}_k$. The approximated signal vector, $\tilde{\mathbf{x}}_l$, is

$$\tilde{\mathbf{x}}_l = \mathbf{F}\,\tilde{\mathbf{w}}_l = \sum_{k=1}^{K} \tilde{w}_{l,k}\mathbf{f}_k. \quad (1)$$

A large number of the coefficients is set to zero, in order to get a sparse representation. The motivation of using an $N \times K$ frame, where $K > N$, instead of an $N \times N$ transform is to have more column vectors to choose from, and thus have a better chance to find a sparse representation that fits the input signal well. Another advantage of frames compared to orthogonal transforms, is that a frame can be trained for a particular class of signals [1]. One disadvantage is that finding the optimal set of selected vectors from the frame and their corresponding coefficients is very hard. An unacceptable computational effort is required. This is why fast, but suboptimal algorithms like Basic Matching Pursuit (BMP) [2], Orthogonal Matching Pursuit (OMP) [3], and Fast Orthogonal Matching Pursuit (FOMP) [4] has been developed and become popular in frame-based coding.

This paper addresses the problem of finding the minimum distortion of the reconstructed signal subject to a given bit budget. The complexity makes this problem not practical to solve. In Section 2 we restate the problem formulation and obtain an efficient algorithm that finds the operational rate-distortion optimal solution, given the frame and knowledge of the code word lengths. The complexity reduction is connected to the decoupling of the signal blocks and the introduction of a new set of independent coefficients for each signal block. In Section 3 we design a solution tree helping us to get further complexity reduction for the problem defined in Section 2. We further reduce the problem complexity by only allowing ordered vector selection. This consequently allows Run-length encoding (RLE) that in most cases reduces the overall bit rate. Experimental results are presented in Section 4, where we compare the new approach performance to Rate-Distortion optimal (RDO) BMP and RDO FOMP [5]. Section 5 concludes this paper.

2. PROBLEM FORMULATION

Consider a frame-based coder where the bit rate of coding coefficient vector $\tilde{\mathbf{w}}_l$, is $R_l(\tilde{\mathbf{w}}_l)$ and the resulting distortion $D_l(\tilde{\mathbf{w}}_l)$. We want to find the minimal total distortion subject to a given bit budget, R_{budget}. We define both the bit rate and the distortion for each block to be independent, i.e., our initial optimization problem is

$$\begin{aligned} \min_{\tilde{\mathbf{w}}_l} \quad & \sum_{l=1}^{L} D_l(\tilde{\mathbf{w}}_l) \\ \text{s.t.} \quad & \sum_{l=1}^{L} R_l(\tilde{\mathbf{w}}_l) = R_{budget}. \end{aligned} \quad (2)$$

This problem is extremely hard to solve, as discussed below. In this section we will reason out a new problem without lack of validity, that is solvable in a reasonable amount of time.

First, we apply the Lagrangian Multiplier method [6, 7], to transform the constrained problem of (2) into a family of easier, unconstrained problems. As can be seen from the equation below, this results in a decomposition of the original problem into a series of independent block optimizations

$$\min_{\tilde{\mathbf{w}}_l} \Big(\sum_{l=1}^{L} D_l(\tilde{\mathbf{w}}_l) + \lambda \sum_{l=1}^{L} R_l(\tilde{\mathbf{w}}_l) \Big)$$
$$= \sum_{l=1}^{L} \Big[\min_{\tilde{\mathbf{w}}_l} \Big(D_l(\tilde{\mathbf{w}}_l) + \lambda R_l(\tilde{\mathbf{w}}_l) \Big) \Big], \quad (3)$$

for $\lambda \in \mathbb{R}^+$. The optimal solution of this problem is the sum of the best RD-tradeoff for each signal block. In order to find the rate that is equal or close to the given bit budget in (2) the problem in (3) is solved iteratively to find the appropriate λ-value.

The distortion, D_l, for block l is defined by

$$D_l(\tilde{\mathbf{w}}_l) = \left\| \mathbf{x}_l - \tilde{\mathbf{x}}_l \right\|^2 = \left\| \mathbf{x}_l - \mathbf{F}\, \tilde{\mathbf{w}}_l \right\|^2. \quad (4)$$

Since we expect a large number of the K elements in $\tilde{\mathbf{w}}_l$ to be equal to zero, only the quantized *values* of the nonzero coefficients and their corresponding *indices* are encoded. After the last nonzero coefficient in each block, we use an End Of Block (EOB) symbol to indicate the start of the next block. We use two distinct VLC tables of finite length, for *value* and *index* symbols, respectively. The rate, R_l, for block l is

$$R_l(\tilde{\mathbf{w}}_l) = \sum_{k \in nz} (R^{val}_{l,k} + R^{ind}_{l,k}) + R^{EOB}, \quad (5)$$

where nz is the set of indices for the nonzero coefficients, $R^{val}_{l,k}$ and $R^{ind}_{l,k}$ the number of bits used to code the *value* and the *index* for the k-th coefficient, respectively, and R^{EOB} the number of bits needed to transmit the EOB symbol.

Even though the signal blocks are independent, there are still dependencies between the coefficients within the same block. The complexity of the minimization in (3) is high, since for every block we need to search for all possible ways to place M nonzero coefficient in a coefficient vector of length K, where $M = 1, 2,$ and so on. The total number of combinations with M nonzero coefficients is $\binom{K}{M}$. In addition, each nonzero coefficient can take on a given number of different *value* symbols, J. For each combination, the number of solutions is J^M. Thus, the total number of different solutions for one single signal block is $\sum_{M=0}^{M_{max}} \binom{K}{M} J^M$, where M_{max} is the largest number of nonzero coefficients we choose to use per block. In all practical cases, $M_{max} << K$, due to the sparse representation idea. A representative number of elements in the coefficient vector is $K = 32$. Let $J = 32$ and $M = \{2, 3, 4\}$. This leads to $\binom{K}{M} = \{496, 4960, 35960\}$ and $J^M = \{1024, 32768, 1048576\}$. A way to remove the latter combinatorial explosion is presented in [8]: For each combination of M out of K nonzero coefficients, $M = \{1, 2, \ldots, M_{max}\}$, the *QR-decomposition* [9] is found for the corresponding M frame vectors. Let $\mathbf{\Phi}_l = \mathbf{Q}_l \mathbf{R}_l$ (of size $N \times M$) be the set of selected frame vectors. $\mathbf{Q}_l$ is an $N \times M$ matrix with orthonormal vectors and $\mathbf{R}_l$ is an $M \times M$ upper triangular matrix. For all cases, $M_{max} < N$, and $span(\mathbf{\Phi}_l) \subset \mathbb{R}^N$. The best signal reconstruction when using *continuous valued* coefficients, $\hat{\mathbf{x}}_l$, is found by the Best Approximation Theorem [9]:

$$\hat{\mathbf{x}}_l = \mathbf{\Phi}_l (\mathbf{\Phi}_l^T \mathbf{\Phi}_l)^{-1} \mathbf{\Phi}_l^T \mathbf{x}_l = \mathbf{Q}_l \mathbf{Q}_l^T \mathbf{x}_l. \quad (6)$$

The error is orthogonal to any vector spanned by the column vectors in $\mathbf{\Phi}_l$. Due to Pythagorean theorem, the distortion in block l, D_l, can be written as the sum of two errors,

$$D_l = \|\mathbf{x}_l - \tilde{\mathbf{x}}_l\|^2 = \|\mathbf{x}_l - \hat{\mathbf{x}}_l\|^2 + \|\hat{\mathbf{x}}_l - \tilde{\mathbf{x}}_l\|^2. \quad (7)$$

The first term in (7) is fixed when the set of selected frame vectors, $\mathbf{\Phi}_l$, is known. For the second term, we define a new coefficient vector with continuous valued elements and length M, $\mathbf{v}^o_l$, where $\hat{\mathbf{x}}_l = \mathbf{Q}_l \mathbf{v}^o_l$. These coefficients are easily found by

$$\mathbf{v}^o_l = \mathbf{Q}_l^T \hat{\mathbf{x}}_l = \mathbf{Q}_l^T \mathbf{x}_l. \quad (8)$$

$\mathbf{v}^o_l$ is a constant for a known $\mathbf{\Phi}_l$. A new vector of *quantized* coefficients, $\tilde{\mathbf{v}}^o_l$, is connected to the reconstructed signal such that $\tilde{\mathbf{x}}_l = \mathbf{Q}_l \tilde{\mathbf{v}}^o_l$. The elements of $\tilde{\mathbf{v}}^o_l$ can take on values listed in the *value* codeword table. *These coefficients are our new decision variables*. In [8] we show that (7) can be written as

$$\|\mathbf{x}_l - \mathbf{Q}_l \mathbf{Q}_l^T \mathbf{x}_l\|^2 + \lambda \Big(\sum_{m=1}^{M} (R^{ind}_{l,m}) + R^{EOB} \Big) + \sum_{m=1}^{M} \min_{\tilde{v}^o_{l,m}} \big((v^o_{l,m} - \tilde{v}^o_{l,m})^2 + \lambda R^{val}_{l,m} \big). \quad (9)$$

This problem is much faster to solve, since only $JMM!$ comparisons are needed, compared to the J^M comparisons in the original problem. To find the optimal RD-tradeoff, we must solve (9) for $\binom{K}{M} M!$ combinations with $M = \{1, \ldots, M_{max}\}$.

3. THE SOLUTION TREE

In this section we organize all $\sum_{M=0}^{M_{max}} \binom{K}{M} M!$ possible ways of combining up to M_{max} frame vectors. We build a solution tree for each signal block, whose purpose is to make it possible to use techniques that give us additional complexity reduction of coding each signal vector optimally.

Consider a tree where each node represents a unique set of selected vectors from the frame $\mathbf{F}$, i.e., a unique $\mathbf{\Phi}_l$. The root node represents the selection of *zero* vectors ($M = 0$). It has K children nodes. Each of these nodes represents the selection of exactly *one* frame vector ($M = 1$). The edges that connect the root node to its children nodes are named “1”, “2”, ... ,“K”, to indicate the index of the frame vector that is selected. All nodes in level 1 have $K-1$ child nodes each, all nodes in level 2 have $K-2$ child nodes, and so on, until level M_{max} where the tree is bounded and no child nodes exist. For any node in the tree (except for the nodes at level M_{max}), the branches to the child nodes are named “1”, “2”, ... ,“K”, except for the names of the branches that leads from the root node down to the current node. This is why the number of children for each node at level M is equal to $K - M$. In other words, a frame vector can never be selected twice. It will have M_{max} generations, or levels. Level M will have $\binom{K}{M} M!$ nodes. The entire tree will represent all possible combinations of $0, 1, \ldots, M_{max}$ vectors selected, including all vector ordering permutations. For each node, we find the minimum rate-distortion solution by solving (9) for the given combination. To find the global optimum solution for the signal block, we need to search through all nodes in the tree.

3.1. Time reduction by depth-first-search

By choosing *depth-first-search* (DFS) [10] as the search strategy in this tree, we can build the QR-decomposition for each node recursively. There are two benefits by using DFS: There is less computation in order to find the QR-decomposition, and only one coefficient has to be found in order to find the best set of coefficients for the respective node.

Suppose that we know the minimum cost, $\min C_l$, for a parent node in level $M-1$,

$$\min C_{l,M-1} = \|\mathbf{e}_{l,M-1}\|^2 + \alpha_{l,M-1} + \beta_{l,M-1}, \qquad (10)$$

where

$$\begin{aligned}
\mathbf{e}_{l,M-1} &= \mathbf{x}_l - \mathbf{Q}_{l,M-1}\mathbf{Q}_{l,M-1}^T\mathbf{x}_l, \\
\alpha_{l,M-1} &= \lambda\Big(\sum_{m=1}^{M-1}(R_{l,m}^{ind}) + R^{EOB}\Big), \\
\beta_{l,M-1} &= \sum_{m=1}^{M-1}\min_{\tilde{v}_{l,m}^o}\big((v_{l,m}^o - \tilde{v}_{l,m}^o)^2 + \lambda R_{l,m}^{val}\big).
\end{aligned}$$

$\mathbf{Q}_{l,M-1}$ $(N \times M-1)$ is the orthonormal basis in the QR-decomposition of $\mathbf{\Phi}_{l,M-1}$, the set of selected frame vectors. When going from a parent node to a child node, we use the same set of selected frame vectors, only adding a new frame vector, $\phi_{l,M}$, as the last column in the set of selected frame vectors, $\mathbf{\Phi}_{l,M}$ $(N \times M)$, that is

$$\mathbf{\Phi}_{l,M} = \begin{bmatrix} \mathbf{\Phi}_{l,M-1} & \phi_{l,M} \end{bmatrix}. \qquad (11)$$

The new frame vector will always be added to the right end of the matrix, since it has a higher frame index value than all the previously selected vectors. When using the *Gram-Schmidt process* [9] to find the QR-decomposition, we know that column vector i in the orthonormal basis is only a function of column vector 1 to i in the original matrix. Thus, the first $M-1$ columns in the orthonormal basis of the QR-decomposition of $\mathbf{\Phi}_{l,M}$, $\mathbf{Q}_{l,M}$, will be equal to the parent's orthonormal basis, $\mathbf{Q}_{l,M-1}$. Since we know $\mathbf{Q}_{l,M-1}$, only the last vector, $\mathbf{q}_{l,M}$, has to be found by Gram-Schmidt to get the entire $\mathbf{Q}_{l,M}$. We show in [11] that the minimum solution for a child node in level M is found by

$$\min C_{l,M} = \|\mathbf{e}_{l,M}\|^2 + \alpha_{l,M} + \beta_{l,M}, \qquad (12)$$

where

$$\begin{aligned}
\mathbf{e}_{l,M} &= \mathbf{e}_{l,M-1} - \mathbf{q}_{l,M}\mathbf{q}_{l,M}^T\mathbf{x}_l, & (13)\\
\alpha_{l,M} &= \alpha_{l,M-1} + \lambda R_{l,M}^{ind}, & (14)\\
\beta_{l,M} &= \beta_{l,M-1} + \min_{\tilde{v}_{l,M}^o}\big((v_{l,M}^o - \tilde{v}_{l,M}^o)^2 + \lambda R_{l,M}^{val}\big). & (15)
\end{aligned}$$

Computing only $\mathbf{q}_{l,M}$ instead of $\mathbf{Q}_{l,M}$ will result in a reduction in the number of multiplications and additions equal to $(M+1)/2$ in situations where $N \gg 1$ [11]. The complexity reduction factor is larger when M is getting larger, i.e., using DFS strategy is more important for higher M_{max}.

Due to the knowledge of the parent node's local minimum, the number of comparisons needed to find the local minimum of the child node in level M is reduced. The first $M-1$ elements in the coefficient vector is the same as the parent's coefficients. Only the last coefficient must be found. In (15) the number of comparisons is reduced from MJ to J, a reduction factor equal to M.

3.2. Time reduction by introduction of lower bounds

In order to find the global optimum solution, all nodes in the solution tree have to be visited. But, finding the optimal solution in each node is not necessarily a requirement. Let C^* be the *so far* minimum cost. As an initial value, C^* could be set to the root node cost, $C_{l,0} = \|\mathbf{x}_l\|^2 + R^{EOB}$. An alternative is to use a fast heuristic to find a suboptimal solution. C^* is updated every time a node's solution is better than C^*. For a node at level M, we define the *lower bound* (LB) for the minimum node cost as

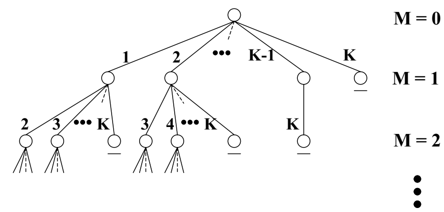

Fig. 1. The reduced solution tree.

$$LB = \|\mathbf{e}_{l,M}\|^2 + \alpha_{l,M} + \lambda M R_{min}^{val}, \qquad (16)$$

where R_{min}^{val} is the minimum number of bits for a *value* codeword. If $C^* < LB$ for a particular node, we know that the global optimal solution is not represented by the particular set of frame vectors, and further computation in order to find $\beta_{l,M}$ is not necessary. The DFS algorithm proceeds to the next node. In the worst case $C^* \geq LB$ in all nodes, and a full computation is necessary. But, in an average case many nodes do not need full computation, and the time saved by the introduction of lower bounds is significant. An example that supports this statement is shown in [11], where the time consumption is further reduced by a factor of 2. This is a significant contribution for speeding up the algorithm without loosing optimality.

3.3. Ordered vector selection and Run-length encoding

Let us add the constraint to our problem that the selected frame vectors, $\mathbf{\Phi}_l$, should only be in the same order as they appear in the frame $\mathbf{F}$. By doing this we reduce the problem complexity, since the $M!$ permutations of a given set of M vectors are avoided. In addition, it allows the use of *Run-length encoding* (RLE) as part of the entropy encoder, which is a bit saving technique when the number of zero elements in the coefficient vector is large. The RLE puts the nonzero coefficients, *values*, in one sequence and the number of zeros between each nonzero coefficient, *runs*, in another sequence. After the last nonzero coefficient in each block, we use an End Of Block (EOB) as we did when coding the *indices* directly. The *run* for the k-th coefficient is the number of zeros *in front* of this coefficient.

The number of solutions is reduced and we can draw a new *reduced* solution tree, shown in Figure 1. This tree is not balanced like the full solution tree. The reduced tree is equal to the full tree at levels 0 and 1, and the edges that connect the root node to its children nodes are named "1", "2", ... ,"K", indicating the vector selected. The node that represents the selection of frame vector 1, $\mathbf{f}_1$, has $K-1$ children nodes. These children represent the selection of *two* vectors ($M=2$), where the first vector of $\mathbf{\Phi}_l$ is $\mathbf{f}_1$ and the second is $\mathbf{f}_i$, where $i = \{2, 3, \ldots, K\}$, respectively. The node, however, representing $\mathbf{\Phi}_l = \mathbf{f}_2$ has $K-2$ children, the node with $\mathbf{\Phi}_l = \mathbf{f}_3$ has $K-3$ children, and so on. Thus, the node with $\mathbf{\Phi}_l = \mathbf{f}_{K-1}$ has only one child and the node representing $\mathbf{\Phi}_l = \mathbf{f}_K$ has none. The reduced tree will have M_{max} levels, as the full tree, but the number of nodes is reduced from $\sum_{M=0}^{M_{max}}\binom{K}{M}M!$ to $\sum_{M=0}^{M_{max}}\binom{K}{M}$. This is a considerable reduction in complexity. We do not have the same problem as when coding the *indices*, since the vector selection is now ordered. The introduction of RLE can clearly give us a more bit efficient representation and thus a better rate-distortion optimal solution.

An overview of the complexities of the cases described so far is given in Table 1. We can see the tremendous reduction in com-

		Complexity
a)	Original nonorthogonalized problem	$1+\sum_{M=1}^{M_{max}}\binom{K}{M}J^M$
b)	Orthogonalized problem	$1+\sum_{M=1}^{M_{max}}\binom{K}{M}JMM!$
c)	Orthogonalized problem using DFS	$1+\sum_{M=1}^{M_{max}}\binom{K}{M}JM$
d)	Orthogonalized problem using DFS and ordered vector selection	$1+\sum_{M=1}^{M_{max}}\binom{K}{M}J$

Table 1. Complexity of optimal coding of one signal block.

plexity between the original nonorthogonalized problem and the orthogonalized problem using depth-first-search and ordered vector selection. In addition, we have a complexity reduction caused by the lower bound technique presented in Section 3.2. For fixed values of K and J, the complexity reduction is becoming very significant for increasing M_{max}.

4. EXPERIMENTS

The optimality of the rate-distortion solution described in previous sections depends on the given frame and the VLC tables. The design of frames [1] is not considered in this work. The performance comparison using a trained frame, a Discrete Cosine Transform (DCT), and a randomly generated frame is presented in [11]. The VLC tables are optimized on a training signal, according to the approach in [8]. In the following experiment, the signal is a Gaussian AR(1) process with $\rho = 0.95$, the number of signal blocks, $L = 512$, and the number of samples per block, $N = 16$. We use a 16×32 frame trained for selecting 4 out of 32 vectors using OMP [1]. As in [11] we call this frame *fr4*. $M_{max} = 4$, and λ is varying from 0.00025 to 0.0020. Let us call the algorithm described in the previous sections the *Optimal Rate-Distortion Encoder* (ORDE). The algorithm using ordered selection is called *ORDE run*, while the non-ordered selection algorithm is called *ORDE ind*. We will compare these algorithms with Rate-Distortion Optimal (RDO) Matching Pursuit, proposed in [5]. There are two matching pursuit techniques presented in [5] that we want to compare with; RDO Basic (or standard) Matching Pursuit (RDO BMP) and RDO Fast (or fully-) Orthogonal Matching Pursuit (RDO FOMP). RDO BMP and RDO FOMP are both greedy vector selection algorithms. They are much faster than the ORDE algorithms, but they are suboptimal. In Figure 2 we see the result of this comparison. The proposed approach is around 15-31 % better than the RDO matching pursuit algorithms. The ORDE algorithm with Run-length coding has an RDO solution that is at the same level as the ORDE with index coding, even though only *ordered* vector selection is used. In other words, the tremendous complexity reduction achieved by ordering the vectors, does not results in a worse RD performance. This is partly because we are now able to use the more efficient RLE scheme.

5. CONCLUSION

An efficient method for rate-distortion optimal frame-based coding is presented in this paper. The efficiency is achieved by: a) using a QR-decomposition which results in a new set of independent decision variables; b) by choosing the right search strategy; and c) by using ordered vector selection that allows the use of Run-length encoding. The complexity overall is reduced by a factor of J^{M-1}, where J is the number of different *values* the representing coefficients can take on, and M is the number of nonzero coefficients per

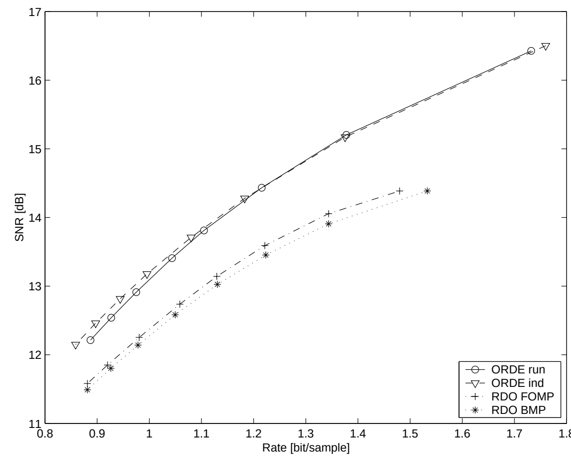

Fig. 2. Results for the ORDE and the Matching Pursuit algorithms.

signal block. This is a tremendous complexity reduction. Experiments show that this method outperforms Rate-Distortion Optimal (RDO) Basic Matching Pursuit and RDO Fast Orthogonal Matching Pursuit. The proposed approach is 15-31 % better than the matching pursuit techniques.

6. REFERENCES

[1] K. Engan, *Frame Based Signal Representation and Compression*, Ph.D. thesis, Norwegian University of Science and Technology/ Stavanger University College, Sept. 2000.

[2] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.

[3] G. Davis, *Adaptive Nonlinear Approximations*, Ph.D. thesis, New York University, Sept. 1994.

[4] M. Gharavi-Alkhansari and T. S. Huang, "A fast orthogonal matching pursuit algorithm," in *IEEE Proc. ICASSP '98*, Seattle, USA, May 1998, pp. 1389–1392.

[5] M. Gharavi-Alkhansari, "A model for entropy coding in matching pursuit," in *IEEE Proc. ICIP '98*, Chicago, USA, Nov. 1998, pp. 778–782.

[6] G. M. Schuster and A. K. Katsaggelos, *Rate-Distortion Based Video Compression*, Kluwer Academic Publishers, Boston, 1997.

[7] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Processing Magazine*, pp. 23–50, Nov 1998.

[8] T. Ryen, G. M. Schuster, and A. K. Katsaggelos, "A rate-distortion optimal coding alternative to matching pursuit," in *IEEE Proc. ICASSP '02*, Orlando, USA, May 2002, pp. 2177–2180.

[9] H. Anton, *Elementary Linear Algebra*, John Wiley and Sons, Inc., New York, 7th edition, 1994.

[10] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, London, 2nd edition, 2001.

[11] T. Ryen, G. M. Schuster, and A. K. Katsaggelos, "A frame-based rate-distortion optimal coding system using a lower bound depth-first-search strategy," in *Proc. of Nordic Signal Processing Symposium*, Tromsø, Oct. 2002.