ZERO REDUNDANCY CHANNEL CODING IN VECTOR QUANTISATION

Indexing terms: Codes and coding, Vector quantisation, Channel coding

Gray-coding of signal amplitudes can protect PCM transmission of analogue signals from the effects of channel bit errors. We generalise this to VQ, and provide a 'vector switching' algorithm that reduces the expected signal distortion by suitably assigning code words to the codevectors in a given codebook.

Introduction: Various methods of assigning binary code words to specify the amplitude samples of an analogue signal have been studied since the early days of PCM.[1] By choosing an appropriate code, such as a Gray code, the average distortion in reproducing the analogue signal due to channel noise can be reduced. Such codes have the property that code words with small Hamming distances from each other correspond to PCM amplitude levels that are a small distance apart from each other. In this manner, error protection against channel noise is provided without the use of any redundancy bits.

We extend this concept from the coding of scalar amplitude levels to the coding of a multidimensional set of amplitudes, or vectors. Such an extension is critically needed in vector quantisation, which is now an important technique in speech communication systems.[2,3] We have found that zero redundancy error control is particularly useful for low-bit-rate mobile radio channels.[3]

In vector quantisation (VQ), an analogue vector is encoded by assigning to it the best matching codevector from a predesigned set of codevectors stored in a codebook. The index, a binary word, that identifies the selected codevector is transmitted across a channel, received and decoded with the corresponding codevector from the codebook. Channel bit errors will cause an incorrectly received binary index. The resulting decoded codevector will no longer be the best match to the original input vector, introducing an increased distortion in the reproduced analogue signal.

We introduce the generalised Gray coding problem: to find the optimal assignment of a unique b-bit codeword to each of the 2^b codevectors in the codebook to minimise the average signal distortion. In general, any change in the assignment of indices to codevectors affects the magnitude of the average signal distortion resulting from channel errors. An assignment of indices to codevectors is equivalent to a permutation of the codevectors in a given codebook. Since there are only a finite number of such permutations, there exists a 'best' permutation which minimises the average signal distortion. The total number of permutations grows as the factorial of the size of the codebook, so that an exhaustive search of every codebook permutation is infeasible. For this reason, we present a locally optimal algorithm with the goal of effectively reducing the expected signal distortion. We call this a pseudo-Gray coding algorithm.

Definitions: Let $\boldsymbol{R}^k$ denote the k-dimensional Euclidean space. The input (analogue) signal vectors are assumed to be elements of this space and have some probability distribution (a joint distribution of the k components of the input vector). Let $\boldsymbol{W}_0 = \{w'_0, w'_1, \ldots, w'_{N-1}\}$ denote a predesigned codebook for vector quantisation of the input vectors, so that $\boldsymbol{W}_0$ is a finite set of $N = 2^b$ vectors in $\boldsymbol{R}^k$, whose subscripts (indices) correspond to an initial and perhaps arbitrary ordering of tne codewords. Let the set of $N = 2^b$ vector indices be denoted by $\boldsymbol{I} = \{0, 1, \ldots, 2^b - 1\}$. To specify the assignment of indices to codevectors, define a one-to-one mapping $f: \boldsymbol{I} \to \boldsymbol{W}_0$, called a permutation function. Then f uniquely assigns a codevector from $\boldsymbol{W}_0$ to each index in $\boldsymbol{I}$. In other words, f specifies a permutation (a rearrangement or reordering) of the initial codebook $\boldsymbol{W}_0$ that defines a newly ordered codebook $\boldsymbol{W} = \{w_0, w_1, \ldots, w_{N-1}\}$, whose codevectors are $\boldsymbol{w}_i = f(i)$.

We assume a memoryless binary symmetric channel with bit error probability ε. An error in one or more of the index bits will result in an incorrectly received index, and thus an incorrectly decoded codevector. For any two binary words $i, j \in \boldsymbol{I}$, let $H(i, j)$ denote the Hamming distance between i and j, i.e. the number of bit positions in which i and j differ.

For each binary index $i \in \boldsymbol{I}$ and each integer m with $0 \leq m \leq b$, define the mth neighbour set of i as $N^m(i) = \{j \in \boldsymbol{I} : H(i, j) = m\}$. If index j is transmitted, the probability that any particular index in $N^m(j)$ is received is given by $q_m = \varepsilon^m(1 - \varepsilon)^{b-m}$.

Let $d(x, y)$ be a real-valued distance function (metric) on $\boldsymbol{R}^k$, and let $p(w)$ be the probability mass function of the codevectors $\boldsymbol{w}$ in $\boldsymbol{W}$.

Distortion minimisation: Suppose an input vector $\boldsymbol{v}$ is approximated (vector-quantised) by the codevector $\boldsymbol{w}_i$ so that channel index i is transmitted, and suppose that possible channel errors cause index j to be received. The total distortion due to the combined effect of the quantisation and channel index errors can be written as

$$d(\boldsymbol{v}, \boldsymbol{w}_j) = d(\boldsymbol{v}, \boldsymbol{w}_i) + [d(\boldsymbol{v}, \boldsymbol{w}_j) - d(\boldsymbol{v}, \boldsymbol{w}_i)] \tag{1}$$

To optimise the performance of the coding system for a given codebook, we would like to minimise the expected value of this distortion over all possible permutations of the initial codebook $\boldsymbol{W}_0$. Thus, we take the expectation of eqn. 1 by averaging over both the distribution of the input vector $\boldsymbol{v}$ and the distribution of channel bit errors. We have

$$E[d(\boldsymbol{v}, \boldsymbol{w}_j)] \leq E[d(\boldsymbol{v}, \boldsymbol{w}_{i(v)})] + E[d(\boldsymbol{w}_{i(v)}, \boldsymbol{w}_j)] \tag{2}$$

by the triangle inequality where $i(\boldsymbol{v})$ denotes the index of the codevector selected by the encoder and is a function of the input $\boldsymbol{v}$, and where j denotes the received index that depends on both $i(\boldsymbol{v})$ and on the channel bit errors.

The right-hand side of the above inequality provides an upper bound for the minimum total expected distortion of the VQ system. This motivates us to seek to minimise this upper bound as a method of reducing the expected codebook distortion. This approach does not necessarily guarantee minimisation of the total distortion but is a heuristic solution that can contribute to its reduction.

The quantity $E_v[d(\boldsymbol{v}, \boldsymbol{w}_{i(v)})]$, the expected distortion due to the approximation of an input vector by a codevector, is independent of the assignment of indices to codevectors, and depends only on the original design of the codebook. It is thus a constant with respect to the minimisation problem. In order to minimise the upper bound, we must find a permutation which minimises the second term which we call the bit error distortion, $D \equiv E[d(\boldsymbol{w}_{i(v)}, \boldsymbol{w}_j)]$.

By using the fact that channel errors are independent of the source statistics, we first average over the input distribution to obtain

$$D = \sum_{r=0}^{2^b-1} p(\boldsymbol{w}_r) E\{d(\boldsymbol{w}_r, \boldsymbol{w}_j)\} \tag{3}$$

where the expectation remaining here is an average over the distribution of j, or, equivalently, over the distribution of the channel errors. Note that by minimising the upper bound rather than the exact average distortion, we have eliminated any need to consider the explicit distribution of the input vector $\boldsymbol{v}$ to find an optimal permutation. It is sufficient to know the codevector probabilities $p(\boldsymbol{w})$.

For any given value of the transmitted index r, the remaining expectation is evaluated using the probability q_m defined earlier, of having an error pattern with exactly $H(r, j) = m$ bit errors. Thus the expectation can be rewritten in terms of the neighbour sets as

$$D = \sum_{r=0}^{2^b-1} \sum_{m=0}^{b} \left\{ q_m \sum_{x \in N^m(r)} p(\boldsymbol{w}_r) d(\boldsymbol{w}_r, \boldsymbol{w}_x) \right\} \tag{4}$$

We define the mth cost $\boldsymbol{C}_m(\boldsymbol{w}_r)$ to be a value which measures the relative contribution to the overall expected bit error distortion of the codebook when exactly m bit errors occur and $\boldsymbol{w}_r$ is the codevector selected by the encoder, as follows:

$$\boldsymbol{C}_m(\boldsymbol{w}_r) = p(\boldsymbol{w}_r) \sum_{x \in N^m(r)} d(\boldsymbol{w}_r, \boldsymbol{w}_x) \tag{5}$$

The above formulation yields a means of determining the minimum upper bound on expected distortion. We define the total cost $C(\boldsymbol{w})$ of a given codevector $\boldsymbol{w}$ as

$$C(\boldsymbol{w}) = \sum_{m=0}^{b} q_m C_m(\boldsymbol{w}) \tag{6}$$

which measures the total contribution to bit error distortion due to possible channel errors when a particular codevector $\boldsymbol{w}$ is selected by the encoder. Using the permutation function f, if each channel codeword r represents the codeword $f(r)$, then we want to find

$$\min_f D = \min_f \left\{ \sum_{r=0}^{2^b-1} C(f(r)) \right\} \tag{7}$$

where the minimisation ranges over all possible permutation functions f. We aim to minimise this quantity in the hope of reducing the expected distortion introduced to the system from the combined effect of quantisation noise and channel errors.

Algorithm: We propose an algorithm whose flow chart is shown in Fig. 1, that rearranges a codebook (denoted by CB in Fig. 1) in a sequence of iterations that converges to a local minimum of the bit error distortion. The main idea of the algorithm is to iteratively switch (i.e. interchange) the positions of two codevectors, reducing the expected value of bit-error distortion at every switch. Thus, a monotonic decrease in distortion results as the algorithm progresses.

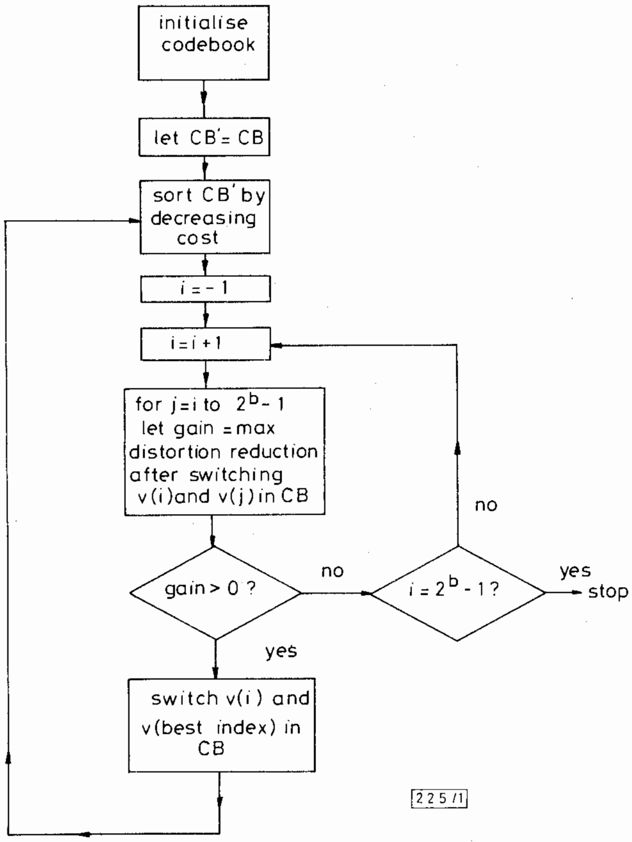

Fig. 1 *Locally optimal pseudo-Gray coding algorithm*

Each codevector $\boldsymbol{v}$ is assigned a cost, $C(\boldsymbol{v})$ as presented earlier. The codevectors are sorted in decreasing order of their cost values. The vector with the largest cost, say $\boldsymbol{v}_0$ is selected as a candidate to be switched first. That codevector which yields the greatest decrease in summed cost when switched with $\boldsymbol{v}_0$ (denoted by GAIN in the flowchart) is then switched with $\boldsymbol{v}_0$. The index corresponding to this codevector is denoted as 'Bestindex' in Fig. 1. If no such vector exists (an unsuccessful switch attempt), then the second highest cost vector is used to check for the most cost-efficient switch, and so on. If every codevector is such that when switched with every other vector an increase in summed cost results, then the algorithm halts in a locally optimal state.

The completion of each switch or unsuccessful switch attempt constitutes the end of an iteration. Following such a switch, the process repeats; the vectors are resorted based on cost, and new switches are attempted. The first block of the flowchart, specifying an 'initialise codebook', indicates that possible nonarbitrary initial codebooks can be chosen. This allows for heuristic preprocessing of a codebook permutation, such as forcing several high-probability codevectors to have a small Hamming distance from several of their close Euclidean neighbours. After the initialisation, the vector switching algorithm can be carried out to find a local minimum of bit error distortion.

Acknowledgment: This work was performed for the Jet Propulsion Laboratory, California Institute of Technology, sponsored by the National Aeronautics and Space Administration.

K. A. ZEGER
A. GERSHO

13th April 1987

Department of Electrical & Computer Engineering
University of California
Santa Barbara, CA 93106, USA

References

1 RYDBECK, N., and SUNDBERG, C. W.: 'Analysis of digital errors in nonlinear PCM', *IEEE Trans.*, 1976, **COM-24**, pp. 59–65

2 GERSHO, A., and CUPERMAN, V.: 'Vector quantization: a pattern matching technique for speech coding', *IEEE Commun. Mag.*, 1983, **21**, pp. 15–21

3 CHEN, J. H., DAVIDSON, G., GERSHO, A., and ZEGER, K.: 'Speech coding for the mobile satellite experiment'. Proceedings IEEE inter. conf. on communications, Seattle, June 1987