A QUADRATIC INDEX ASSIGNMENT ALGORITHM FOR VECTOR QUANTISATION OVER NOISY TRANSMISSION CHANNELS

N. L. C. Talbot (1) and G. C. Cawley (2)

(1) unaffiliated
(2) School of Information Systems, University of East Anglia, Norwich, Norfolk, NR4 7TJ, U.K.

ABSTRACT

Vector quantisation is a widely used technique in low-bit rate coding of speech signals, but is highly sensitive to errors in the transmitted codeword, due to noise in the transmission channel. This can degrade speech quality quite dramatically if the reference vector retrieved by the corrupted index is very different from the reference vector corresponding to the intended index. The index assignment (IA) process attempts to re-order the code book to minimise the effects of single bit errors by assigning similar reference vectors indices with similar binary patterns, generally at considerable computational expense. This paper demonstrates that the index assignment process can be expressed as a quadratic assignment problem, and solved using Hall's [1] method. This new method produces similarly well ordered code books to those obtained using a simple pairwise exchange algorithm with simulated annealing, but is around two orders of magnitude faster.

1. INTRODUCTION

Given p-dimensional unlabelled data $Y = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_m\} \subset I\!R^p$ representative of a data manifold $V \subseteq I\!R^p$, the process of vector quantisation [2] attempts to partition V into a number of sub-regions V_i using a finite set of reference or *code book* vectors, $W = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3, \ldots, \boldsymbol{w}_n\} \subset I\!R^p$. The values of the reference vectors are chosen so as to minimise the quantisation error, measured according to a distance metric, between a training vector $\boldsymbol{y}_i$ and the best matching reference vector $\boldsymbol{w}_{(\boldsymbol{y}_i)}$ over all vectors in Y. An incoming vector $\boldsymbol{v} \in V_i$ is mapped onto the most similar reference vector $\boldsymbol{w}_i$. The incoming vector can then be represented by the index i, resulting in a considerable reduction in the required bit rate.

Vector quantisation is however very sensitive to errors in the transmitted codeword, due to noise in the transmission channel, as the reference vectors recalled by the intended and corrupted indices might be very different. The index assignment (IA) process attempts to re-order the code book, such that similar reference vectors are recalled by indices with similar binary patterns [3]. This minimises the effect of an error in the transmitted codeword, as the reference vector corresponding to the corrupted index is made as similar as possible to the reference vector represented by the intended index.

2. QUADRATIC INDEX ASSIGNMENT

The index assignment process, where a code book is re-ordered such that similar reference vectors are represented by indices with similar binary patterns, is similar to the problem of placement in VLSI circuits, where modules, or cells, are placed on a two-dimensional surface so as to minimize the total wire length [4]. The aim of index assignment can be viewed as placing the $n = 2^r$ reference vector indices in an r-dimensional binary space, such that the indices of similar reference vectors are placed in similar locations. Like VLSI cell placement, the solution can be found using Hall's [1] solution to the quadratic assignment problem. A recursive sorting algorithm is then used to re-order the code book according to the resulting placement.

2.1 Quadratic Assignment

Given n nodes, and an $n \times n$ symmetric connection matrix $\boldsymbol{C}$ (with $c_{ii} = 0$) representing the connection strength between nodes i and j, the aim is to determine positions for each of the n nodes such that the weighted sum of squared distances is minimised [1]. If each element of the vector $\boldsymbol{x}_i$ represents the ith coordinate of each node, then to place the nodes in an r-dimensional space, the vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_r$ are required such that

$$\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(x_{i1} - x_{j1})^2 c_{ij} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(x_{i2} - x_{j2})^2 c_{ij} + \cdots + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(x_{ir} - x_{jr})^2 c_{ij} \quad (1)$$

is minimised. To avoid the trivial solution, $\boldsymbol{x}_i = \boldsymbol{0}$, the r constraints

$$\boldsymbol{x}_i^T \boldsymbol{x}_i = 1 \quad (2)$$

are imposed. Hall then defines the 'disconnection' matrix, $\boldsymbol{A}$, to be:

$$a_{ij} = \begin{cases} -c_{ij} & i \neq j \\ \sum_{k=1}^{n} c_{ik} & i = j \end{cases} \quad (3)$$

and shows that the objective function can be rewritten as:

$$\boldsymbol{x}_1^T \boldsymbol{A} \boldsymbol{x}_1 + \boldsymbol{x}_2^T \boldsymbol{A} \boldsymbol{x}_2 + \ldots + \boldsymbol{x}_r^T \boldsymbol{A} \boldsymbol{x}_r \quad (4)$$

It can be shown [1] that the disconnection matrix has n eigenvalues of the form $0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \cdots \leq \lambda_{n-1}$ and that the solution is given by the eigenvectors corresponding to the first r non-zero eigenvalues.

2.2 Index Assignment

The following procedure is used to reorder the code book using Hall's quadratic assignment algorithm: given $n = 2^r$ reference vectors, construct the disconnection matrix such that the connection strength c_{ij} is large if $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ are similar, and small if they are dis-similar. For initial experiments, the connection strength is calculated as follows:

$$c_{ij} = \begin{cases} \frac{1}{\|\boldsymbol{w}_i - \boldsymbol{w}_j\|} & i \neq j \\ 0 & i = j \end{cases} \quad (5)$$

The $(n \times r)$ solution matrix is formed from the eigenvectors corresponding to the first r non-zero eigenvalues of the disconnection matrix:

$$\boldsymbol{X} = [\boldsymbol{x}_1\ \boldsymbol{x}_2\ \cdots\ \boldsymbol{x}_r] = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1r} \\ x_{21} & x_{22} & \cdots & x_{2r} \\ \vdots & & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nr} \end{pmatrix} \tag{6}$$

An ordered code book is then produced according to a simple recursive algorithm (Algorithm 1):

1. Construct the $(n \times (r+p))$ augmented matrix $\boldsymbol{Z} = [\boldsymbol{X} \,|\, \boldsymbol{W}^T]$:

$$\boldsymbol{Z} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1r} & \boldsymbol{w}_1^T \\ x_{21} & x_{22} & \cdots & x_{2r} & \boldsymbol{w}_2^T \\ \vdots & & \ddots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nr} & \boldsymbol{w}_n^T \end{pmatrix}$$

2. For each $i = 1, \ldots, r$
   set $N = 2^{r-i+1}$
   3. For each $j = 1, \ldots, 2^{i-1}$
      sort rows $(j-1) \times N + 1$ to $j \times N$ of $\boldsymbol{Z}$ so that column i is in piecewise ascending order
4. Remove the first r columns of the matrix $\boldsymbol{Z}$

$\boldsymbol{Z}$ now represents the sorted code book.

Algorithm 1: Recursive sorting algorithm

3. RESULTS

For evaluation, a code book of 256 reference vectors, compiled from the data obtained from line spectral pair (LSP) analysis of speech from the DARPA/TIMIT speech corpus, was re-ordered using the quadratic assignment method and a conventional pair-wise exchange algorithm with simulated annealing [5]. The code books generated using the quadratic assignment and simulated annealing algorithms, are similarly well-ordered, as illustrated by the graph of mean error power against the Hamming distance of the codewords, over all reference vectors in the code book shown in Figure 1. The graph was created by simulating 1, 2, ..., 8 bit errors in every possible bit position for each reference vector, and noting the spectral distortion. As can be seen, the randomly ordered code book shows practically no change in performance as the Hamming distance increases, as expected, whereas the two ordered code books show significantly reduced mean error power for single bit errors. The quadratic assignment process produces an optimal placement in an r-dimensional continuous space, however the discretisation of this placement by the partitioning algorithm may not necessarily be optimal. As a result the simulated annealing algorithm can produce a better code book structure, although the difference is normally minimal. The quadratic assignment method however was found to be around two orders of magnitude faster than the conventional approach based on simulated annealing.

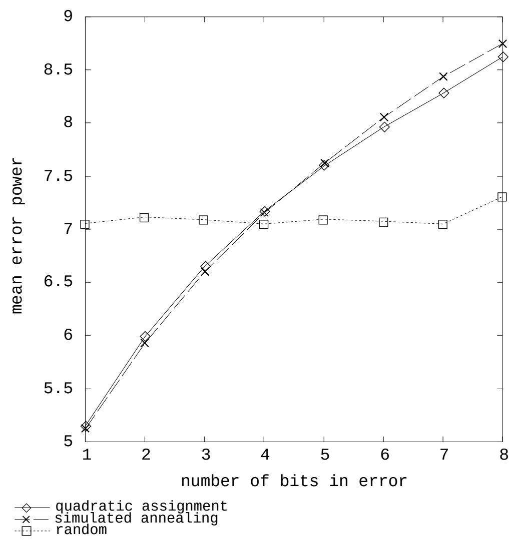

Figure 1: Graph of mean error power against Hamming distance, for an un-ordered code book, and code books ordered using simulated annealing and quadratic assignment algorithms.

4. CONCLUSIONS AND FURTHER RESEARCH

A novel index assignment algorithm is presented, based on Hall's solution to the quadratic assignment problem. The resulting code book ordering is of comparable quality to that generated using a conventional simulated annealing algorithm, but is obtained at a greatly reduced computational expense.

Further research is needed to determine the optimal strategy for converting the continuous valued vectors $\boldsymbol{x}_i$ into a sorted code book. In addition, distance metrics for connection strengths, such as $c_{ij} = -\|\boldsymbol{w}_i - \boldsymbol{w}_j\|$, are being considered to optimise the code book order for single bit errors, which are by far the most common.

5. REFERENCES

[1] K. HALL,
'An r-Dimensional Quadratic Placement Algorithm',
Management Science, 77(3), (November 1970).

[2] R. M. GRAY,
'Vector Quantization',

I.E.E.E. Acoustics, Speech and Signal Processing Magazine, pp 4–29, (April 1984).

[3] N. FARVARDIN & V. VAISHAMPAYAN,
'On the Performance and Complexity of Channel-Optimized Vector Quantizers',
IEEE Transactions on Information Theory, 37(1):155–160, (January 1991).

[4] N. L. C. TALBOT,
An Application Oriented Comparison of Optimization and Neural Network Based Design Techniques,
PhD thesis, University of Essex, Department of Electronic Systems Engineering, 1996.

[5] G. C. CAWLEY,
The Application of Neural Networks to Phonetic Modelling,
PhD thesis, University of Essex, Department of Electronic Systems Engineering, 1996.