On the reduction of fractal image compression encoding time

B E Wohlberg G de Jager
Digital Image Processing Laboratory, Faculty of Engineering, University of Cape Town,
Private Bag, Rondebosch 7700, SOUTH AFRICA
{brendt, gdj}@dip1.ee.uct.ac.za

***Abstract*— Lossy image coding by Partitioned Iterated Function Systems, popularly known as Fractal Image Compression, has recently become an active area of research. An image is coded as a set of contractive transformations in a complete metric space. As a result of a well known theorem in metric space theory, the set of contractive transformations (subject to a few constraints) is guaranteed to produce an approximation to the original image, when iteratively applied to *any* initial image. While rapid decompression algorithms exist, the compression process is extremely time consuming; an exhaustive search for the optimum mappings is $O(n^4)$ for an $n \times n$ image. The most common solution involves classification of domain and range blocks according to features such as the presence of edges, after which matches across class boundaries are excluded. We propose a geometric construction, allowing clustering, as well as providing upper and lower bounds for the best match between domain and range blocks, allowing blocks to be excluded from the computationally costly matching process.**

I. Introduction

A Partitioned Iterated Function System (PIFS) encoding of an image consists of a set of transforms on regions of the image. The set of regions from which the transform domains are chosen (the domain blocks) overlap, while the regions forming the ranges of the transformations are tiled (the range blocks)[1]. The simplest implementation has all tiled $b \times b$ blocks in the image as the set of range blocks, and all (overlapping) $2b \times 2b$ blocks in the image as the set of domain blocks [1, pg. 912]. The set of transformations consists of a spatial contraction (eg. averaging each 4 neighbouring blocks to construct a $b \times b$ block from a $2b \times 2b$ block), followed by one of the 8 square symmetry operations (4 rotations and 4 reflections), followed by a contractive affine transformation on the greyscale values (for a block with pixel values $(p_1, p_2, \ldots, p_n)$):

$$(T_{s,o})_i = sp_i + o$$

where $-1 < s < 1$ guarantees contractivity.

If such mappings are found for each range block, their union defines a mapping on the image as a whole. This mapping will be a contraction mapping in the metric space (with RMS distance measure) of greyscale images. This metric space is complete (ignoring the sampling and quantisation, which fix limits as to how close the sequence may approach the limit); Banach's fixed point theorem [2, pg 84] therefore implies that iterative application thereof to *any* initial image will generate a sequence converging to the unique fixed point of the transformation.

The image is thus encoded as a set of transformations, which have as their fixed point an image close to it in the sense of the distance metric used. The transformation co-efficients are then quantised and entropy coded. In order to minimise the distance between the image to be encoded and the fixed point of the transformation (ie. the lossiness of the encoding), a matching domain block must be found for every range block, with as small a distance as possible between them under the set of transformations discussed above.

A more effective scheme utilises a quadtree partition of range blocks, where a range block is subdivided into four smaller blocks if no domain can be found to match within an acceptable tolerance [1] [3]. A triangular range partition is also reported to be effective [4].

II. Computational Complexity

Consider an $n \times n$ image and $b \times b$ range blocks. The number of tiled range blocks is n^2/b^2, while the number of domain blocks is $(n-2b+1)^2$. The computation of best match between a range block and a domain block is $O(b^2)$. Considering b to be constant, the computational complexity of an exhaustive search is $O(n^4)$.

The computational requirements for an exhaustive search are prohibitive (in the region of 30 hours on a SUN sparc10 workstation for a 256×256 image). The most common approach to reducing computational demand is to classify the image blocks into a number of classes, and to avoid attempting matches across class boundaries [5] (eg. a smooth domain block is unlikely to match a range block containing an edge, under any affine transformation), thereby avoiding the costly matching process for these blocks. Saupe [6] has recently proposed utilising an invariant (under the set of transformations applied to domain blocks) representation of image blocks, followed by a fast nearest neighbour search in the space of these representations. Expected encoding complexity based on this algorithm is $O(n^2 \log n)$.

We propose a geometric view of the minimisation problem, which allows us to derive upper and lower bounds of distances between blocks in terms of distances already computed. Computational load for this scheme *(for block to block matching,*

[1]The range/domain labels are occasionally reversed in the literature, where the terminology is instead based on the transformations used in image reconstruction.

not for the the minimisation algorithm as a whole) is independent of block size, unlike direct computation, which is $O(b^2)$ for $b \times b$ blocks.

III. MINIMISATION PROBLEM

Consider an image I, together with the set of $b \times b$ range blocks, and $2b \times 2b$ domain blocks. Construct the set of range vectors $R \subset \mathbb{R}^n$ (where $n = b^2$), by taking the pixel values in each range block in scan-line order. The set of domain vectors $D \subset \mathbb{R}^n$ is constructed by subsampling the domain blocks by averaging, followed by the procedure applied to the range vectors.

Define $S_i : \mathbb{R}^n \mapsto \mathbb{R}^n, \quad 1 \leq i \leq 8$ as the symmetry operations on the square, and $T_{a,b} : \mathbb{R}^n \mapsto \mathbb{R}^n, \quad a, b \in \mathbb{R}$ where $(T_{a,b}\boldsymbol{u})_i = a\boldsymbol{u}_i + b$. If we utilise the Euclidean distance measure, the metric space $(\mathbb{R}^n, d_E)$ is also an inner product space, with the norm and distance defined in the usual way:

$$\|\boldsymbol{u}\| = \sqrt{\boldsymbol{u} \cdot \boldsymbol{u}}$$

$$d_E(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|$$

The minimisation problem is then to find the $\boldsymbol{u} \in D$ such that $min_{a,b,c}\|T_{a,b}S_c\boldsymbol{u} - \boldsymbol{v}\|$ is a minimum, for each $\boldsymbol{v} \in R$. Ignoring symmetry operations, which we have yet to properly address within this framework (one may view S_c as generating an expanded domain set), we consider finding $min_{a,b}\|T_{a,b}\boldsymbol{u} - \boldsymbol{v}\|$.

IV. GEOMETRIC CONSTRUCTION

In this section we shall introduce a geometric view leading to our proposed solution to the previously described minimisation problem. If the angle between vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ is ϕ

$$\cos\phi = \frac{\boldsymbol{u} \cdot \boldsymbol{v}}{\|\boldsymbol{u}\| \; \|\boldsymbol{v}\|}$$

We show later that the minimisation problem may be restated in terms of angular distances between vectors, and provide rapidly computable upper and lower bounds on unknown angles as a function of known angles.

Define $\mathbf{1}_n = (1, 1, \ldots, 1) \in \mathbb{R}^n$, where we shall refer to $\mathbf{1}$ where n is obvious from context. Note that $\|\mathbf{1}_n\|^2 = n$. The transformations $T_{a,b}$ may then be written as $T_{a,b}\boldsymbol{u} = a\boldsymbol{u} + b\mathbf{1}$, and the minimum distance between a range vector $\boldsymbol{v}$ and a domain vector $\boldsymbol{u}$ as $\min_{a,b}\|T_{a,b}\boldsymbol{u} - \boldsymbol{v}\|$. Each domain vector, under the set of transformations $T_{a,b}$, generates a plane in $\mathbb{R}^n$, containing span$\{\mathbf{1}\}$ (see figure 2).

Geometrically, if we define $\boldsymbol{p}$ to be the closest vector in the plane span$\{\boldsymbol{u}, \mathbf{1}\}$ to $\boldsymbol{v}$ (see figure 1), and $\boldsymbol{n} = \boldsymbol{v} - \boldsymbol{p}$, the required minimum distance is $\|\boldsymbol{n}\|$. From a geometric perspective, the minimisation requirement is to find the closest domain "plane" to each range vector. We shall show that knowledge of the angles between the planes, and the angle between a range vector and one of the planes provides information regarding the range vector angles with the other planes, making direct computation of angular distances unnecessary for certain vectors.

Since $\boldsymbol{p}$ (as defined above) lies in span$\{\boldsymbol{u}, \mathbf{1}\}$, we have that $\boldsymbol{p} = a^*\boldsymbol{u} + b^*\mathbf{1}$ for some a^*, b^*, and $\boldsymbol{n} = \boldsymbol{v} - \boldsymbol{p}$. Since $\boldsymbol{n}$ is

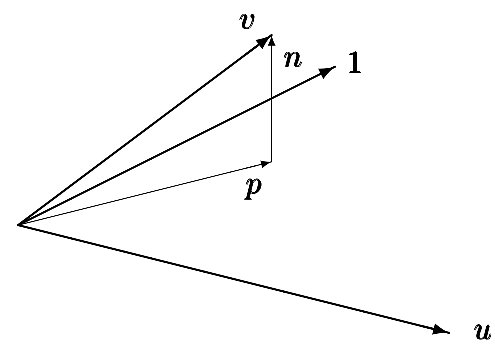

Fig. 1. Geometry of perpendicular from span$\{\boldsymbol{u}, \mathbf{1}\}$ to $\boldsymbol{v}$

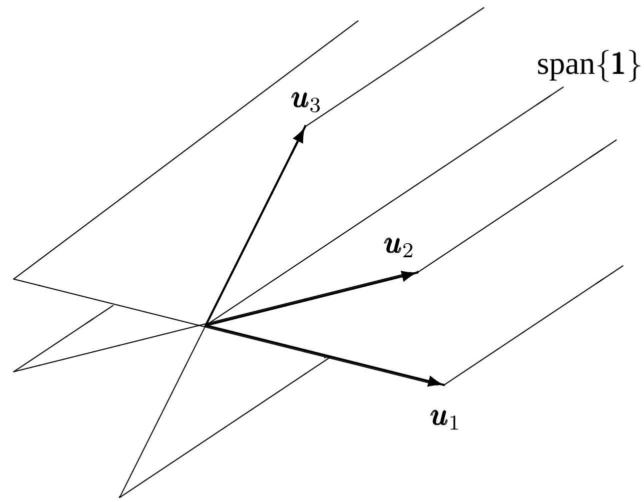

Fig. 2. Geometry of planes generated by domains under operator $T_{a,b}$

orthogonal to span$\{\boldsymbol{u}, \mathbf{1}\}$, we have that $\boldsymbol{n} \cdot \boldsymbol{u} = 0$ and $\boldsymbol{n} \cdot \mathbf{1} = 0$. Solving for a^*, b^*, we obtain

$$a^* = \frac{\|\mathbf{1}\|^2\boldsymbol{u} \cdot \boldsymbol{v} - (\boldsymbol{u} \cdot \mathbf{1})(\boldsymbol{v} \cdot \mathbf{1})}{\|\mathbf{1}\|^2\|\boldsymbol{u}\|^2 - (\boldsymbol{u} \cdot \mathbf{1})^2}$$

$$b^* = \frac{\|\boldsymbol{u}\|^2\boldsymbol{v} \cdot \mathbf{1} - (\boldsymbol{u} \cdot \mathbf{1})(\boldsymbol{v} \cdot \boldsymbol{u})}{\|\mathbf{1}\|^2\|\boldsymbol{u}\|^2 - (\boldsymbol{u} \cdot \mathbf{1})^2}$$

$$\boldsymbol{n} = \boldsymbol{v} - a^*\boldsymbol{u} - b^*\mathbf{1}$$

where $\|\boldsymbol{n}\|$ is the required minimum distance.

V. PROJECTION OPERATOR

Since the planes generated by the domain vectors are embedded in $\mathbb{R}^n$ where $n > 3$, the planes do not have a unique normal. In order to define an angle between planes, we find for each plane a vector in that plane and perpendicular to $\mathbf{1}$, and define angles between planes in terms of angles between these vectors. In addition, we would like inequalities involving the angles between a range vector and two range "planes". Since the range vector, the two domain vectors, and $\mathbf{1}$ span a 4-dimensional subspace of $\mathbb{R}^n$, we would like to project these vectors into the 3-dimensional subspace perpendicular to $\mathbf{1}$

(and containing the origin), thereby allowing the application of results from solid geometry.

The following projection operator[2] satisfies both requirements.

Definition 1:

$$P_{\boldsymbol{n}}\boldsymbol{u} = \boldsymbol{u} - (\frac{\boldsymbol{u}\cdot\boldsymbol{n}}{\|\boldsymbol{n}\|^2})\boldsymbol{n}$$

$P_{\boldsymbol{n}}\boldsymbol{u}$ projects $\boldsymbol{u}$ parallel to $\boldsymbol{n}$ into the space perpendicular to $\boldsymbol{n}$, and containing the origin (see figure 3).

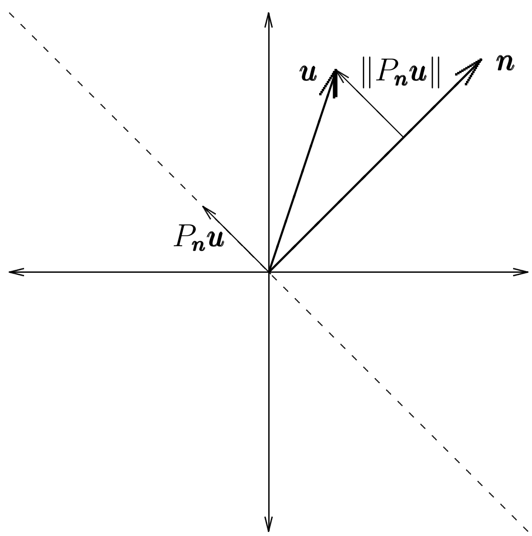

Fig. 3. Projection operator

Since $P_{\boldsymbol{1}}\boldsymbol{u} = \boldsymbol{u} - (\frac{\boldsymbol{u}\cdot\boldsymbol{1}}{\|\boldsymbol{1}\|^2})\boldsymbol{1}$, we have that $P_{\boldsymbol{1}}\boldsymbol{u} \in \text{span}\{\boldsymbol{u},\boldsymbol{1}\}$, and $P_{\boldsymbol{1}}\boldsymbol{u} \perp \boldsymbol{1}$. Therefore we may define the angle between the planes $\text{span}\{\boldsymbol{u},\boldsymbol{1}\}$ and $\text{span}\{\boldsymbol{v},\boldsymbol{1}\}$ as the angle between $P_{\boldsymbol{1}}\boldsymbol{u}$ and $P_{\boldsymbol{1}}\boldsymbol{v}$.

The following results are required:

Lemma 1: If $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3 \in \mathbb{R}^3$, and the angles between $\boldsymbol{v}_1$, $\boldsymbol{v}_2$ and $\boldsymbol{v}_2$, $\boldsymbol{v}_3$ and $\boldsymbol{v}_1$,$\boldsymbol{v}_3$ are α_{12}, α_{23} and α_{13} respectively, then

$$|\alpha_{12} - \alpha_{23}| \le \alpha_{13} \le \alpha_{12} + \alpha_{23}$$

Proof: $\alpha_{13} \le \alpha_{12} + \alpha_{23}$ and its cyclic permutations are a standard result of solid geometry (see [7, pg. 106] and figure 4). This implies $\alpha_{23} - \alpha_{12} \le \alpha_{13}$ and $\alpha_{12} - \alpha_{23} \le \alpha_{13}$. The result follows immediately.

For brevity we introduce the notation $\boldsymbol{v}' = P_{\boldsymbol{1}}\boldsymbol{v}$ for any vector $\boldsymbol{v}$.

Theorem 1: Given range vector $\boldsymbol{v}$ and domain vectors $\boldsymbol{u}_1$,$\boldsymbol{u}_2$, and defining $\alpha_1 = \angle(\boldsymbol{v}', \boldsymbol{u}_1')$ and $\alpha_2 = \angle(\boldsymbol{v}', \boldsymbol{u}_2')$, if $0 \le \alpha_1, \alpha_2 \le \pi/2$, then $\alpha_1 < \alpha_2 \;\Rightarrow$ the minimum distance from $\boldsymbol{v}$ to $\text{span}\{\boldsymbol{u}_1, \boldsymbol{1}\}$ is less than the minimum distance to $\text{span}\{\boldsymbol{u}_2, \boldsymbol{1}\}$

Proof: Since $\text{span}\{\boldsymbol{u}_i, \boldsymbol{1}\} = \text{span}\{\boldsymbol{u}_i', \boldsymbol{1}\}$ we may write the minimum distance vector $\boldsymbol{n}_i$ in terms of $\boldsymbol{u}_i'$ and $\boldsymbol{1}$ rather than $\boldsymbol{u}_i$ and $\boldsymbol{1}$, resulting in considerable simplification, as $\boldsymbol{u}_i' \cdot \boldsymbol{1} = 0$.

$$\boldsymbol{n}_i = \boldsymbol{v} - \tilde{a}\boldsymbol{u}' - \tilde{b}\boldsymbol{1}$$

[2] It is interesting to note that Saupe [6] utilises an identical operator (apart from normalisation) for his divergent approach.

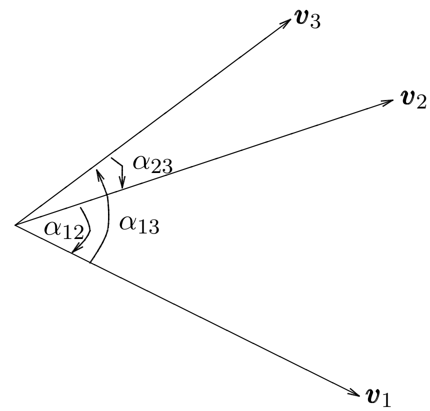

Fig. 4. Angular inequalities in $\mathbb{R}^3$

where

$$\tilde{a} = \frac{\boldsymbol{u}_i' \cdot \boldsymbol{v}}{\|\boldsymbol{u}_i'\|^2} \quad \tilde{b} = \frac{\boldsymbol{v}\cdot\boldsymbol{1}}{\|\boldsymbol{1}\|^2}$$

giving

$$\|\boldsymbol{n}_i\|^2 = \boldsymbol{n}_i \cdot \boldsymbol{n}_i = \|\boldsymbol{v}\|^2 - \frac{(\boldsymbol{u}_i' \cdot \boldsymbol{v})^2}{\|\boldsymbol{u}_i'\|^2} - \frac{(\boldsymbol{v}\cdot\boldsymbol{1})^2}{\|\boldsymbol{1}\|^2}$$

If $\alpha \in [0, \pi/2]$ we have that $\cos\alpha$ is positive and strictly decreasing[3].

$$\begin{aligned} & \angle(\boldsymbol{v}', \boldsymbol{u}_1') < \angle(\boldsymbol{v}', \boldsymbol{u}_2') \\ \Rightarrow \quad & \frac{\boldsymbol{u}_1' \cdot \boldsymbol{v}'}{\|\boldsymbol{u}_1'\| \; \|\boldsymbol{v}'\|} > \frac{\boldsymbol{u}_2' \cdot \boldsymbol{v}'}{\|\boldsymbol{u}_2'\| \; \|\boldsymbol{v}'\|} \\ \Rightarrow \quad & \frac{(\boldsymbol{u}_1' \cdot \boldsymbol{v}')^2}{\|\boldsymbol{u}_1'\|^2} > \frac{(\boldsymbol{u}_1' \cdot \boldsymbol{v}')^2}{\|\boldsymbol{u}_1'\|^2} \end{aligned}$$

since $\boldsymbol{v}' \cdot \boldsymbol{u}_i' = \boldsymbol{v} \cdot \boldsymbol{u}_i'$ and we are considering positive quantities only. Result follows by inspection of $\|\boldsymbol{n}_i\|^2$.

In the following section we shall show how these results may be applied to the optimisation problem.

VI. Distance bounds

Consider the set of domain vectors $\boldsymbol{u}_1 \ldots \boldsymbol{u}_n$ and range vectors $\boldsymbol{v}_1 \ldots \boldsymbol{v}_m$. Define the angle between $\text{span}\{\boldsymbol{u}_i, \boldsymbol{1}\}$ and $\text{span}\{\boldsymbol{u}_j, \boldsymbol{1}\}$ by

$$\cos\alpha_{ij} = \frac{\boldsymbol{u}_i' \cdot \boldsymbol{u}_j'}{\|\boldsymbol{u}_i'\| \; \|\boldsymbol{u}_j'\|}$$

Once α_{kj} have been computed for some k and all j, lemma 1 gives bounds

$$|\alpha_{ik} - \alpha_{kj}| \le \alpha_{ij} \le \alpha_{ik} + \alpha_{kj}$$

on α_{ij} since $\text{span}\{\boldsymbol{u}_i, \boldsymbol{u}_j, \boldsymbol{u}_k, \boldsymbol{1}\} \in \mathbb{R}^4$, embedded in $\mathbb{R}^n$, and is projected into $\mathbb{R}^3$ by operator $P_{\boldsymbol{1}}$.

We propose clustering the $\boldsymbol{u}_i'$ according to angles α, utilising lemma 1 to speed up the process. Choose an initial vector, $\boldsymbol{u}_1'$ for example, and compute $\alpha_{1i}\ \forall i$. Define all $\boldsymbol{u}_i'$ such that α_{1i} is less than a predetermined threshold to be within a cluster

[3] Since we admit negative values of a^*, the maximum possible angular distance is $\pi/2$, and this condition is therefore not problematic

centred at $\boldsymbol{u}'_1$. Pick a $\boldsymbol{u}'_j$ a large angular distance from $\boldsymbol{u}'_1$ and repeat the process, utilising lemma 1 to exclude unnecessary computation where possible. The process should terminate with a set of k clusters[4], within each of which the angular distances from one vector to another are small.
Define the angle between $\boldsymbol{v}'_i$ and $\boldsymbol{u}'_j$ by

$$\cos \beta_{ij} = \frac{\boldsymbol{v}'_i \cdot \boldsymbol{u}'_j}{\|\boldsymbol{v}'_i\| \, \|\boldsymbol{u}'_j\|}$$

Theorem 1 guarantees that we may find the best matching domain vector for range $\boldsymbol{v}'_i$ by minimising β_{ij} over all j. Compute the β_{ij} to all k cluster centres. Since angles are small within the clusters, lemma 1 may be expected to give useful bounds on the angular distance between $\boldsymbol{v}'_i$ and the cluster members. In addition, all members of a cluster may be excluded if the angular distance to the cluster centre is large enough. In addition, many important quantities may be rapidly computed from these angles, eg.

$$a^* = \frac{\|\boldsymbol{v}'_i\|}{\|\boldsymbol{u}'_j\|} \cos \beta_{ij}$$

allowing the contractivity condition to be checked.

VII. Conclusions

The proposed process reduces computational requirements in two separate ways. First, the clustering made possible by theorem 1 makes distance computation unnecessary for members of all clusters with sufficiently large angular distance from the cluster centre to the range vector in question. Second, lemma 1 allows exclusion from direct computation of distances to cluster members, by utilising the angular distances from cluster centres to cluster members, and from the range vector in question to cluster centres, which are already known at this stage in the process.
While these results have yet to be tested empirically, we expect a considerable reduction in computational requirements. Many aspects remain unexplored, such as the possibility of exploiting the continuity property of images, which results in spatially neighbouring blocks in an image having small mutual angular distances where edges are not present. In addition, the design of an optimal clustering algorithm for this application requires attention.

References

[1] Y. Fisher, "A discussion of fractal image compression," in *Chaos and Fractals: New Frontiers of Science* (H.-O. Peitgen, H. Jürgens, and D. Saupe, eds.), ch. Appendix A, pp. 903–919, New York: Springer-Verlag, 1992.

[2] C. Pitts, *Introduction to Metric Spaces*. Edinburgh: Oliver & Boyd, 1972.

[3] G. Lu and T.-L. Yew, "Image compression using partitioned iterated function systems," in *Image and Video Compression*, vol. SPIE Vol. 2186, pp. 122–133, The International Society of Optical Engineers, 1994.

[4] F. Davoine, E. Bertin, and J.-M. Chassery, "From rigidity to adaptive tesselations for fractal image compression: Comparative studies," in *IEEE 8th Workshop on Image and Multidimensional Signal Processing*, Sept. 1993.

[4] Assuming that only the most promising cluster is searched the optimum number of clusters is $k \approx \sqrt{n}$ where n is the number of domain vectors

[5] A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Transactions on Image Processing*, vol. 1, pp. 18–30, Jan. 1992.

[6] D. Saupe, "Breaking the time complexity of fractal image compression," tech. rep., Institut für Informatik, Universität Freiburg, 1994.

[7] K. L. Nielsen and J. H. Vanlonkhuyzen, *An outline of plane and spherical trigonometry*. New York: Barnes & Noble, 1944.

Presented at COMSIG94:

B. Wohlberg and G. de Jager, "On the reduction of fractal image compression encoding time," in *1994 IEEE South African Symposium on Communications and Signal Processing (COMSIG '94)*, (University of Stellenbosch), pp. 158–161, IEEE, Oct. 1994. IEEE Cat No 94TH0665-0.