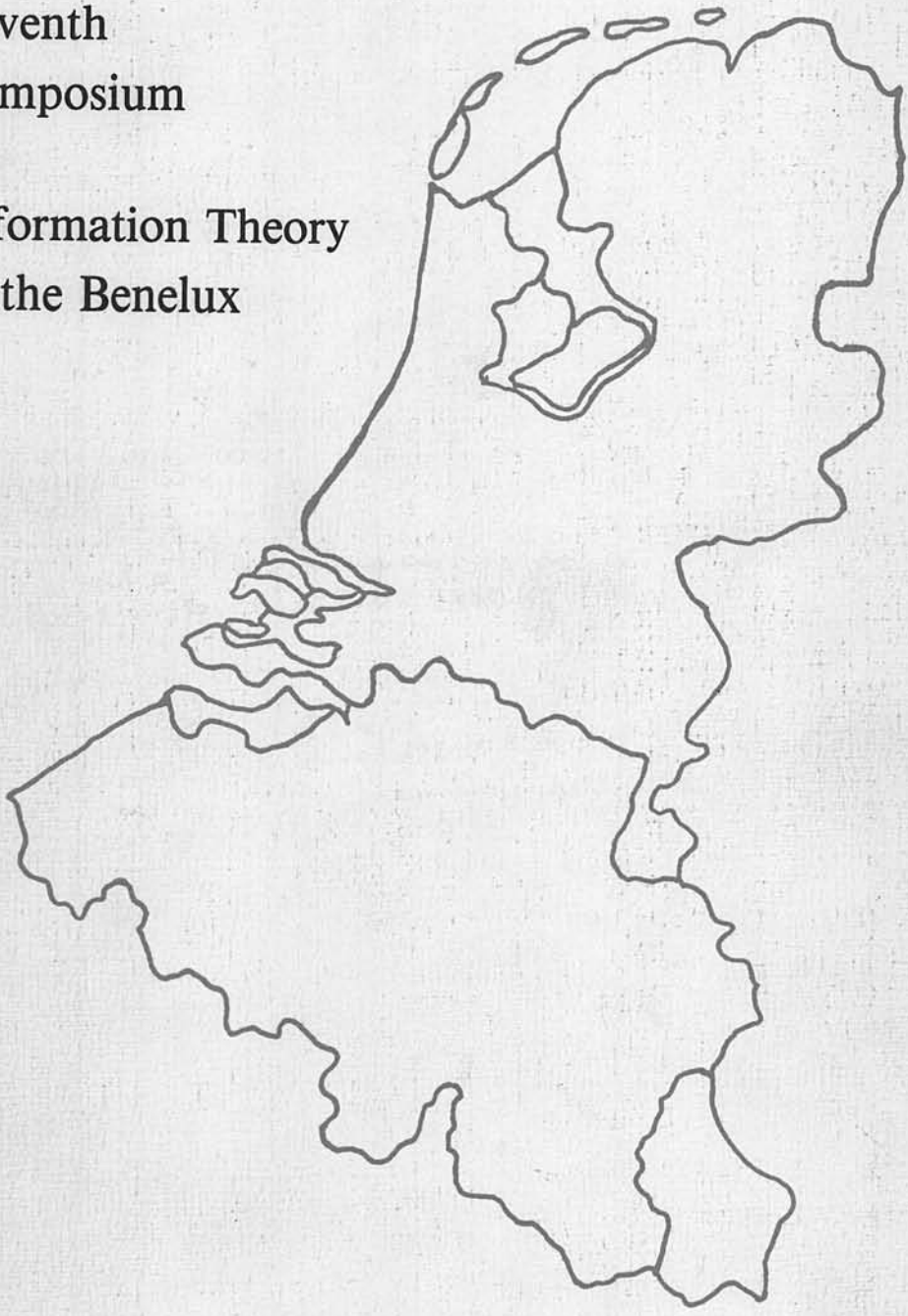
Seventh Symposium on Information Theory in the Benelux



Noordwijkerhout - the Netherlands
May 22-23, 1986

2074 6116

INFORMATION THEORY IN THE BENELUX

INFORMATION THEORY
IN THE BENELUX

Proceedings of the Seventh Symposium on Information Theory in the Benelux

May 22-23, 1986
Noordwijkerhout
the Netherlands

Edited by
D.E. Boekee

Organized by
Information Theory Group
Department of Electrical Engineering
Delft University of Technology
Delft, the Netherlands

Werkgemeenschap Informatie- en Communicatietheorie
Enschede, 1986

Delft University Press/1986

Published and distributed by

Delft University Press
Stevinweg 1
2628 CN Delft
the Netherlands
Telephone: (0) 15 783254

Printed in the Netherlands

Under the auspices of:

Werkgemeenschap Informatie- en Communicatietheorie,
Afdeling der Elektrotechniek,
Technische Hogeschool Twente,
Postbus 217, 7500 AE Enschede, The Netherlands

Previous Symposia:

1st: 1980 (Zoetermeer)
2nd: 1981 (Zoetermeer)
3rd: 1982 (Zoetermeer)
4th: 1983 (Haasrode) ISBN 90-334-0690-X
5th: 1984 (Aalten) ISBN 90-71048-01-2
6th: 1985 (Mierlo) ISBN 90-71048-02-0

CIP-gegevens Koninklijke Bibliotheek, Den Haag

Proceedings of the Seventh Symposium on Information Theory in the Benelux, held in Noordwijkerhout, the Netherlands, May 22-23, 1986

Ed. by D.E. Boekee

Delft: Delft University Press

Fig., tab.
Met lit.opg., reg.

ISBN 90-6275-272-1

Key word: Information Theory

TABLE OF CONTENTS

3. *Source coding*

4. *Image processing*

5. *Picture coding*

6. *Detection and estimation*

7. *Multi-user theory/cryptograph*

PREFACE

The sequence of yearly Benelux Symposia on Information Theory, of which the seventh one is held this year, has started in 1980. It is the purpose of the symposia to offer an opportunity to researchers in the field of information theory within the Benelux to present recent results of their work. The steadily increasing number of presentations and attendees clearly demonstrates the strong interest within the Benelux in information theory and its applications. In this respect I mention an increasing number of presentations and attendees from industrial research centers, emphasizing the growing mutual research interests of universities, institutes and industries in the Benelux.

Much research in our field is presently related to image processing. It is therefore a privilege to us that Prof. J.W. Woods accepted an invitation to be the 1986 guest lecturer at the symposium. Prof. Woods is a well-known expert in the field of two-dimensional signal processing, in particular image restoration and image coding.

The organizing committee of this symposium was formed by Profs.E.W. Grõneveld, E.C. van der Meulen, J.P.M. Schalkwijk and D.E. Boekee.

Finally, I would like to express my thanks and appreciation to Mrs. Y. Smits, who skillfully assisted in the organization of the symposium and to Mrs. M. van Velzen and Mrs. A. Bosch for their typing and secretarial support.

Dick E. Boekee

May, 1986.

PREDICTIVE VECTOR QUANTIZATION OF IMAGES*

John W. Woods and Hsueh-Ming Hang

This paper presents two techniques for the unification of predictive tree encoding and vector quantization. We refer to such approaches as predictive vector quantization (PVQ). The unification is achieved by imposing a tree structure on the VQ table with the branch symbols progressively specifying the quantizer outputs. A modification of the LBG design algorithm can then be made, incorporating an (M,L) tree search, to optimize the PVQ encoding. Experimental results show a marked improvement over tree encoding alone.

INTRODUCTION

By *predictive vector quantization* (PVQ) we mean a predictive tree encoding in which the ordinary scalar quantizer is replaced by a vector quantizer (VQ). Because typical images have high correlation over neighboring pixels, they can be compressed by employing a predictive model such as DPCM and tree codes [1], [2]. However, since a real image is locally nonstationary, a scalar quantizer together with a fixed structure coding filter can only condense pictures to a certain extent. Vector quantizers help improve the coding performance because they quantize a whole block of data and, thus, can match local image statistics better. The purpose of this paper is to review new image coding schemes based on the PVQ concept [3].

Rate distortion theory indicates that a well-defined signal source can be compressed closely to the rate distortion bound, provided that the coding block length is large enough [4]. From this viewpoint, conventional DPCM has the drawback that its predictor only uses the past information to remove redundancy and its quantizer only operates

---

* This is a shortened version of a paper based on the doctoral thesis of H.-M. Hang and published in the *IEEE Transactions on Communications* in November 1985.

J.W. Woods is with ECSE Department, R.P.I., Troy, NY 12181, on sabbatical leave at Delft University of Technology.
H.-M. Hang is with AT and T Bell Laboratories, Holmdel, NJ 07733.

on a single pixel. A predictive tree code is thus introduced by adding a delayed decision feature which makes use of the nearby future data [5], [6]. The tree code is then further improved by replacing the scalar quantizer with a vector quantizer, resulting in a predictive vector quantizer.

Image encoding using PVQ [7] is not a straightforward extension of the ordinary vector quantization. A special implementation of 1-D PVQ has appeared for speech coding in Stewart et al. [8]. But the full potential of the general PVQ approach, especially its application to images, had not been explored. In order to construct a code tree on a compact 2-D region, we devised a 2-D decision order which provides an appropriate encoding sequence for 2-D tree codes. The details of this ordering can be found in [9], [10].

PREDICTIVE VECTOR QUANTIZATION

The basic idea of predictive vector quantization (PVQ) is to use a predictive filter to remove the predictable redundancy in the data and then use a VQ to encode the prediction error. We will review two implementations of PVQ namely, *sliding block PVQ* and *block tree PVQ*.

Sliding Block PVQ

Fig. 1 represents an ordinary sliding block decoder in which the u_i's are the inputs to the shift register, the q_i's are the outputs of the decoder, and $\mathbf{F}$ is a time-invariant mapping which specifies the output value q_i. Suppose the shift register is binary with length J; then the total number of possible states of this machine is 2^J, i.e. the mapping $\mathbf{F}$ has 2^J entries. This mapping $\mathbf{F}$ can thus be viewed as a lookup table, with the shift register acting as an address selector which picks entries in the table to form the outputs. In this way, the current output q_i is determined by the vector $\mathbf{U}_i=(u_i,u_{i-1},\ldots,u_{i-J+1})$ which is the state of the shift register, where u_i is the current input and $u_{i-1},u_{i-2},\ldots,u_{i-J+1}$ are the J-1 previous inputs. Hence, the information contained in the previous data can be utilized to select the best current output value q_i.

We can view $\mathbf{U}_i$ as an index to the vector quantizer. At the time J, this index corresponds to the representation vector $\mathbf{Q}_J=(q_J,q_{J-1},\ldots,q_1)$. For $i > J$, we simply slide the block to the right; hence the name "*sliding block*" for this type of VQ. To quantize a sampled waveform, the source signal is compared against all the quantization levels specified by the shift register of which the latest input has two possible values; the one with least distortion is then selected. The ordinary scalar quantizer can be viewed as the special case of this machine which only contains one element in the shift register. Therefore, if we choose the mapping $\mathbf{F}$ properly, the performance of VQ will always be better than that of a scalar quantizer.

We can also adopt a sliding block structure to implement PVQ, which we call *sliding block PVQ* (SBPVQ). The block diagram of a 2-D SBPVQ decoder is shown in Fig. 2. The encoding filter in this decoder is a recursive difference equation,

$$\hat{s}(m,n) = \sum_{i,j} c_{ij} \cdot \hat{s}(m-i,n-j) + q(m,n)$$

$$\equiv c * \hat{s}_{old} + q(m,n),$$

i.e. the reproduced signals $\{\hat{s}(\cdot,\cdot)\}$ are filter outputs driven by the selected PVQ levels. One of the problems in applying this scheme to a 2-D image is the selection of a register support for the mapping $\mathbf{F}$. Since an image pixel is highly correlated with its neighbors, naturally we would choose a compact region around the current point to be our register support. For example, the causal region of Fig. 3 could be the support of the register in Fig. 2. As we slide the region of Fig. 3 horizontally across the image, the current quantization level (input to the filter) is determined by the contents of the register, i.e. the previous and current path map symbols. The encoding filter then uses this quantization level to generate a reproducing pixel, $\hat{s}(m,n)$. Essentially, an SBPVQ requires about the same amount of computation as a tree code but needs an extra register and a VQ table.

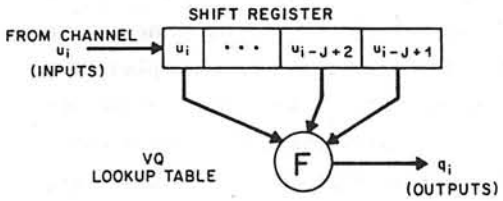


Fig. 1. A sliding block channel decoder.

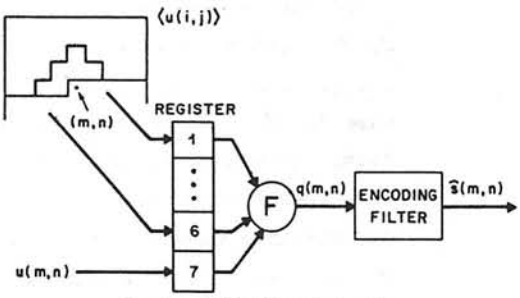


Fig. 2. A 2-D SBPVQ decoder.

In order to describe our design algorithm, we need to define two more terms. In the encoding process, releasing a data pixel is equivalent to selecting an entry in the VQ table for that pixel. The index of the selected entry will be called the *partition index associated with this pixel.* Also the unquantized prediction error (i.e., $e(m,n) = s(m,n) - c * \hat{s}_{old}$) will be called the *prediction error associated with the released pixel.* The SBPVQ design algorithm can then be described.

SBPVQ Design Algorithm:

Step 1. Initialization: Start with some initial value for **F**. For example, use the scalar quantization levels derived from a predictive tree code (so-called product VQ codes in [11]).

Step 2. Coding: Apply the above encoding procedure to the training data, i.e., introduce a minimum distortion partition $\{P_1,\ldots,P_N\}$ on the test image. Store the prediction error $e(m,n)$ and the partition index of each pixel. The partition index associated with a data point is, equivalently, the contents of the register used to encode that pixel.

Step 3. Updating **F**: Since the squared-error is used, the new quantization level of index j is the average of all the prediction errors of partition index j, i.e.,

$$q_j = \frac{1}{|P_j|} \sum_{(m,n)\in P_j} e(m,n)$$

where $|P_j|$ denotes the number of training vectors in partition P_j.

Step 4. Compute the distortion and compare it to the previous distortion. Stop if the distortion decrement is less than a prespecified value. Otherwise, go to Step 2.

Block Tree PVQ

The block tree implementation of PVQ is easy to appreciate in concept. A test image is first partitioned into small blocks, and then predictive tree coding is performed on each block. The difference between block tree PVQ and a tree code is that the quantization levels in the former are vectors.

Initially, we considered ideal block PVQ (full-searched PVQ) which has a full size VQ table and requires an exhaustive search. Due to computational consideration, this scheme was deemed impractical. Then we imposed a tree structure on the VQ table, calling the new algorithm *block tree PVQ*. The idea of tree-structured vector quantizers was first proposed by Buzo et al. for linear predictive coding (LPC) of speech [12]. However, the tree search technique used here is different. The tree-searched VQ table in [12] is a list of vectors organized by a tree-like framework, and the search is basically an address locating procedure. A node in that tree is the representative for all the nodes (or branches) extending from that node. Only the ultimate leaves (nodes or branches without successors) of the tree are used as code vectors. On the other hand, we follow the traditional sequential tree coding approach to construct the VQ table. Every tree branch is a part of a code vector. A complete code vector is formed by concatenating the branch symbols along any path in the tree.

The structure of an ideal block PVQ decoder is shown in Fig. 4. The path map **u** from the channel is a vector containing an address in the VQ lookup table. An entry in the VQ lookup table is another vector which is a sequence of quantization levels used to drive the en-

coding filter.

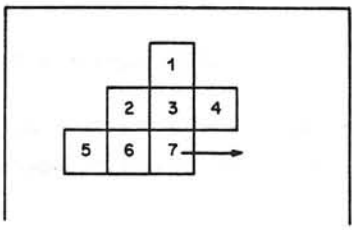


Fig. 3. A causal region for the register in SBPVQ.

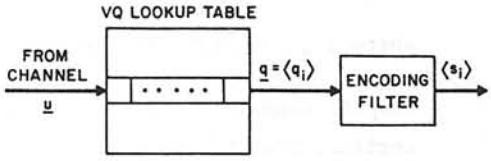


Fig. 4 An ideal block PVQ decoder.

As a simple decoding example, consider the test image of Fig. 5. At the receiver, the quantization vector $\mathbf{q} = \{q_1, q_2, \ldots, q_9\}$ of a 3x3 block is selected by the path map symbol **u**. Then, each element q_i passes through the encoding filter and yields the reconstructed signals $\hat{s}_i$ sequentially.

Since the block of this PVQ is a compact 2-D region, the search order of elements inside a block should follow the 2-D search ordering defined in [9] and [10], there called decision ordering. Indeed, the 1-D like search yields a less satisfactory result [3]. The importance of search order becomes apparent when a full-searched PVQ is replaced by a tree-searched PVQ. The 2-D search region also limits the geometric shape of the encoding filter so that the decoder is causally realizable. For instance, a nonsymmetric half-plane filter cannot be used with a rectangular block search region.

The computational problem of an ideal block PVQ can be greatly eased by imposing a tree structure on the VQ table, as mentioned above, and applying the (M,L) search algorithm to the code tree. We call this new scheme *block tree PVQ* (BTPVQ). As illustrated by Fig. 6, the VQ lookup table now has a tree structure and a path in the tree is the quantization vector identified by the path map symbol **u**. If we apply an (M,L) search with M=8 on the test image of Fig. 5, the encoder only conducts 2x8 or fewer decoder operations per pixel, which is much smaller than the 512 operations of the ideal block PVQ.

Additionally, the encoder does not have to make a decision immedia-

tely at the end of a block. Instead, it can delay its decision-making and thus take advantage of the dependence between successive blocks. For example, the code tree in the first block of Fig. 5 can be extended to the second block, and the encoder would then release the first block after reaching the end of the second block. In other words, the tree structure inside one block would act as a substitute for a full-searched table, and the delayed decision feature can be brought in by allowing the tree to grow continuously over several blocks.

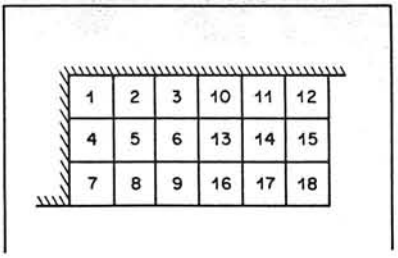


Fig. 5 A test image for BTPVQ.

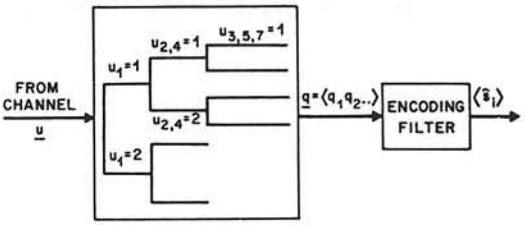


Fig. 6 The structure of BTPVQ.

EXPERIMENTAL RESULTS

We present results on a man's face image of size 256x256 with 8 bits/pixel grey level. A zero-mean version of this image was coded in the image density domain. The SNR results quoted are defined in terms of peak-to-peak signal (255) to rms noise, as is standard in the image processing field. The bit rate is 1 bit/pel.

The man's face image was coded with DPCM, tree coding using the (M,L) algorithm, and with BTPVQ. A closeup of the results is shown in Fig. 7. Image A is the original. Image B is the DPCM result with SNR=26.9 dB. Image C is the result of tree coding with M=8 and L=20. The SNR is 30.4 dB which is 3.5 dB more than DPCM. Image D is the result of BTPVQ with M=8 and L=2x3x3, i.e. two 3x3 blocks. The SNR is 32.5 dB which is 2.1 dB more than tree coding. Subjectively we see a marked improvement in the coded result with respect to tree coding.

The above is perhaps an unfair comparison because the DPCM and tree coding parameters were determined from the man's face image while the BTPVQ parameters were obtained from another image (Lady). Within the

training set the BTPVQ resulting SNR was 33.7 dB which is a bit more than a doubling in performance with respect to tree coding at the rate of 1 bit/pel.

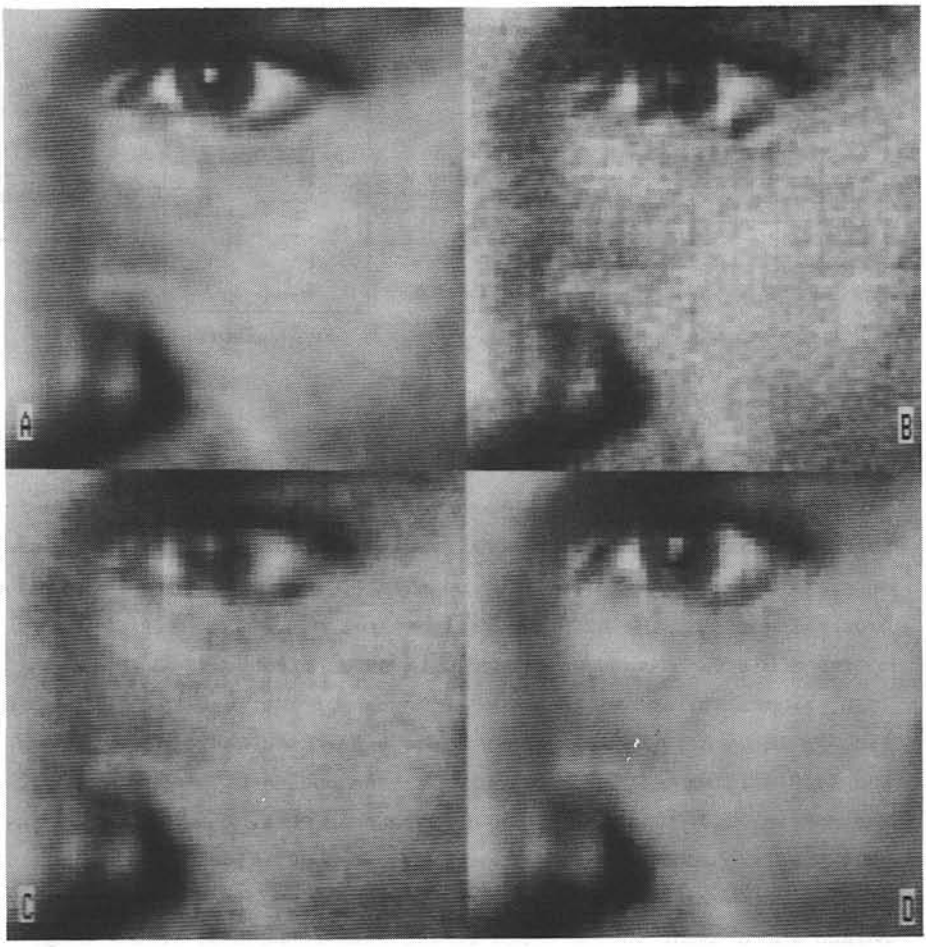


Figure 7

REFERENCES

[1] J.B. O'Neal, Jr., "Predictive quantizing systems (differential pulse code modulation) for the transmission of television signals", *Bell Syst. Techn. J.*, vol. 45, pp. 689-721, May-June 1966.

[2] J.W. Modestino, V. Bhaskaran and J.B. Anderson, "Tree encoding of images in the presence of channel errors", *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 677-697, Nov. 1981.

[3] H.-M. Hang and J.W. Woods, "Predictive vector quantization of images", *IEEE Trans. Commun.*, vol. COM-33, p. 1209-1219, Nov. 1985.

[4] T. Berger, *Rate Distortion Theory: A Mathematical Basis for Data Compression*. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[5] J.B. Anderson and J.B. Bodie, "Tree Encoding of Speech", *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 379-387, July 1975.

[6] J.B. Anderson and C.W. Law, "Real-number convolutional codes for speech-like quasi-stationary source", *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 778-782, Nov. 1977.

[7] H.-M. Hang and J.W. Woods, "Predictive vector quantization of images", presented at 18th Annu. Conf. Inform. Sci. Syst., Princeton, NJ, Mar. 1984.

[8] L.C. Stewart, R.M. Gray and Y. Linde, "The design of trellis waveform coders", *IEEE Trans. Commun.*, vol. COM-30, pp. 702-710, Apr. 1982.

[9] H.-M. Hang, "Two-dimensional seqeuntial decision ordering", submitted for publication.

[10] H.-M. Hang, "Predictive coding of images", Doctoral thesis, Dep. Elec., Comput., Syst. Eng., Rensselaer Polytech. Inst.,Troy, NY, July 1984.

[11] R.M. Gray, "Vector Quantization", *IEEE ASSP Mag.*, pp. 4-29, Apr. 1984.

[12] A. Buzo, A. Gray, Jr., R.M. Gray, and J.D. Markel, "Speech coding based upon vector quantization", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 562-574, Oct. 1980.

DOUBLY STOCHASTIC GAUSSIAN RANDOM FIELD MODELS FOR IMAGE ESTIMATION

John W. Woods

ABSTRACT

The two-dimensional (2-D) doubly stochastic Gaussian (DSG) model was introduced by one of the authors to provide a complete model for spatial filters which adapt to the local structure in an image signal. Here we present the optimal estimator and 2-D fixed-lag smoother for this DSG model extending earlier work of Ackerson and Fu. As the optimal estimator has an exponentially growing state space, we investigate a suboptimal estimator using an M-algorithm tree searching approach.

INTRODUCTION

For some time it has been apparent that linear shift-invariant (LSI) filtering is of limited utility in many image processing problems. The main difficulty is that the constraint of shift-invariance leads to blurring of the edges in images. This effect has motivated the introduction of many adaptive procedures, e.g. [1,2,3] which attempt to track the apparent spatial inhomogeneity (nonstationarity) in images. Some of these filters have obtained better mean square error (MSE) and most have offered better subjective improvement than the LSI filters designed with a linear minimum MSE (LMMSE) criterion.

In this paper we regard the image random field as globally homogeneous but possesing a local structure created by a hidden 2-D Markov chain. The coefficients of a conditionally Gaussian, autoregressive model are switched by the Markov chain to generate the required local structure. The resulting non-Gaussian random field, termed doubly stochastic Gaussian (DSG), has apparent inhomogeneity on a local scale as well as homogeneity on a global scale. The estimators designed from this model have shown both good subjective and MSE improvement [4,11] unlike the LSI case where only good numerical improvement is obtained.

J.W. Woods is with ECSE Department, R.P.I. Troy, NY 12181, on sabbatical leave at Delft University of Technology. Research supported by the U.S. National Science Foundation under Grant ECS-8313889 and the Netherlands Organization for the Advancement of Pure Research (ZWO).

The MSE error criterion is believed to be more subjectively relevant for the new model because of the DSG model's incorporated local structure.

DSG RANDOM FIELDS

We generalize the conventional nonsymmetric half-plane (NSHP) autoregressive Gaussian model [6] by allowing the model parameters to be a function of a discrete valued structure field $\ell(n_1,n_2)$,

$$s(n_1,n_2) = \sum_{m_1,m_2} c_{m_1,m_2}^{\ell(n_1,n_2)} s(n_1-m_1,n_2-m_2) + \sigma_{\ell(n_1,n_2)} w(n_1,n_2) \tag{1}$$

where $w(n_1,n_2)$ is a white Gaussian noise field with zero mean and unit variance and σ_ℓ is the rms value of the prediction error in model state ℓ. If we take the structure field ℓ to be a 2-D Markov Chain, we get an overall Markov model for the joint field $\{s(n_1,n_2),\ell(n_1,n_2)\}$ only the first component of which is observable. The idea of a 2-D Markov chain was used in [5] to model facsimile images and was generalized in [7] to model image structure. The composite field manifested in (1) is termed DSG in analogy to the doubly stochastic Poisson terminology for the compound Poisson process. The DSG model has been employed in [7] to perform image estimation and in [8] to improve adaptive prediction DPCM image coding.

We assume that the 2-D Markov chain is chosen to be homogeneous. This then implies the DSG field would also be asymptotically homogeneous given the BIBO stability of the elemental NSHP models

$$\{c_{m_1,m_2}^{\ell},\sigma_\ell\} \qquad \text{for } \ell = 1,\ldots,L. \tag{2}$$

If we choose these causal models to approximate predominant correlation directions, then we introduce a structure which appears locally to be inhomogeneous thus matching this observed quality in images. On the other hand, we have a global homogeneity which permits the estimation of the DSG model parameters and estimation errors in the ergodic case. This combination is potentially very advantageous for image pro-

cessing applications. Typically we choose 4 correlation directions as an appropriate compromise for a local prediction model which is predicting just one pixel ahead.

OPTIMAL ESTIMATOR

This signal is observed in white Gaussian noise according to the observation equation,

$$r(n_1,n_2) = s(n_1,n_2) + v(n_1,n_2), \quad (n_1,n_2) \in [0,N_1-1]x[0,N_2-1], \tag{3}$$

where the observation noise $v(n_1,n_2)$ is independent of $w(n_1,n_2)$. The object is to find for fixed $k_1 \geq 0$ and $k_2 \geq 0$, the MMSE estimate of $s(n_1-k_1,n_2-k_2)$ given the causal set of observations up to pixel (n_1,n_2) denoted:

$$\underline{R}(n_1,n_2) \triangleq \{r(0,0),r(1,0),\ldots,r(N_1-1,0);r(0,1),\ldots,r(N_1-1,1);\ldots;$$
$$r(0,n_2-1),\ldots,r(N_1-1,n_2-1);r(0,n_2),\ldots,r(n_1,n_2)\}.$$

Introducing the vector notation:

$$\underline{L}_j(n_1,n_2) \triangleq [\ell_j(0,0),\ldots,\ell_j(N_1-1,0);\ldots;\ell_j(0,n_2),\ldots,\ell_j(n_1,n_2)],$$

for a Markov chain path from pixel (0,0) up to and including (n_1,n_2) and using Bayes' rule and the preceding definitions, the optimal fixed-lag estimate is given for each pixel (n_1,n_2) by

$$\hat{s}(n_1-k_1,n_2-k_2|n_1,n_2) \triangleq \sum_{j=1}^{L} \hat{s}_j(n_1-k_1,n_2-k_2|n_1,n_2) \cdot$$
$$P[\underline{L}_j(n_1,n_2)|\underline{R}(n_1,n_2)], \tag{4}$$

where $L = L^{(n_2-1)N_1+n_1}$ and

$$\hat{s}_j(n_1-k_1,n_2-k_2|n_1,n_2) \triangleq E[s(n_1-k_1,n_2-k_2)|\underline{R}(n_1,n_2),\underline{L}_j(n_1,n_2)].$$

The following recursive expression may then be derived for the condi-

tional path probability, analogously to the 1-D case [9],

$$P[\underline{L}_j(n_1,n_2)\,|\underline{R}(n_1,n_2)] = k\; P[\underline{\ell}_j(n_1,n_2)\,|\underline{\ell}_j(n_1-1,n_2)] \cdot$$

$$p[r(n_1,n_2)\,|\underline{R}(n_1-1,n_2),\underline{L}_j(n_1,n_2)] \cdot P[\underline{L}_j(n_1-1,n_2)\,|\underline{R}(n_1-1,n_2)]. \tag{5}$$

Since the set of random variables $\underline{R}(n_1,n_2)$ is conditionally jointly Gaussian distributed we may write,

$$p[r(n_1,n_2)\,|\underline{R}(n_1-1,n_2),\underline{L}_j(n_1,n_2)] \sim N[\hat{s}_j(n_1,n_2|n_1-1,n_2)\ ,$$

$$\sigma_j^2(n_1,n_2) + \sigma_v^2], \tag{6}$$

where $\sigma_j^2(n_1,n_2)$ is the *a priori* error variance of the 2-D Kalman filter with model sequence $\underline{L}_j(n_1,n_2)$. The *a posteriori* probabilities are calculated using (5) and (6). The MMSE optimal estimate of $s(n_1-k_1, n_2-k_2)$ is then calculated from (4).

Unfortunately, the number of paths which must be considered in evaluating (4) is generally exponential in n_1 and n_2 and hence this optimal estimator is non-implementable.

SUBOPTIMAL ESTIMATOR

We attempt to overcome this problem of exponential growth in the required number of filters by extending the approach of Tugnait and Haddad [10]. Our objective is to restrict the number of filters to a reasonable number. Instead of propagating filters matched to all possible sequences $\underline{L}_j(n_1,n_2)$ for all j, we discard some of the unlikely model sequences.

M Algorithm

In this suboptimal estimator, we limit the number of filters to a maximum allowable number M. Instead of carrying along the *a posteriori* probabilities $P[\underline{L}_j(n_1,n_2|\underline{R}(n_1,n_2)]$ for all j, we now keep only the M most probable sequences and discard the rest. Suppose at pixel (n_1-1,n_2)

there are M sequences. Each of the M sequences is extended by L models at (n_1,n_2). Therefore there will be ML extensions at pixel (n_1,n_2). The *a posteriori* probabilities for these ML extensions are formed according to (5) where now $j=1,\ldots,ML$. These *a posteriori* probabilities are then arranged in descending order and the model sequences $\underline{L}_j(n_1,n_2)$ corresponding to the first M probabilities are chosen as the M sequences to be carried forward to the next pixel. The approximate estimator equation is then given by

$$\hat{s}_M(n_1-k_1,n_2-k_2|n_1,n_2) = \sum_{j=1}^{M} \hat{s}_j(n_1-k_1,n_2-k_2|n_1,n_2) \cdot P[\underline{L}_j(n_1,n_2)|\underline{R}(n_1,n_2)], \quad (7)$$

where the *a posteriori* probability is given by (5) with a reordered index j now satisfying $1 \leq j \leq M$.

Clearly, as M approach $L^{(n_2-1)N_1-n_1}$, the suboptimal estimator performance will approach that of the optimal estimator (4). In a practical case we would of course hope to use a much smaller value of M. For a given value of M, (7) will require running M 2-D Kalman filters with correspondingly M separate global states [6].

Merging of Sequences

It is possible that two model sequences $\underline{L}_j(n_1,n_2)$ and $\underline{L}_k(n_1,n_2)$ may haye the same recent models and differ only in early models and because of this the predictions $\hat{s}_j(n_1,n_2|n_1-1,n_2)$ and $\hat{s}_k(n_1,n_2|n_1-1,n_2)$ are very 'close'. In this situation it is useful to 'merge' the two sequences, i.e., to absorb the probability of one sequence into the other and discard the first [10]. Our decision to merge the sequences is based on the Bhattacharya distance between the two conditional probability densities,

$$p[r(n_1,n_2)|\underline{R}(n_1-1,n_2),\underline{L}_j(n_1,n_2)] \text{ and}$$

$$p[r(n_1,n_2)|\underline{R}(n_1-1,n_2),\underline{L}_k(n_1,n_2)].$$

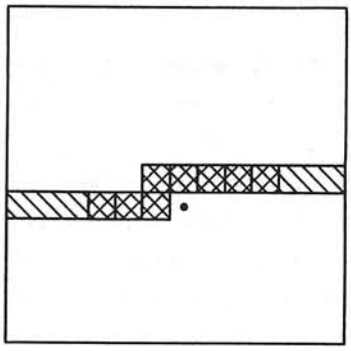
The B-distance measure is given by

$$d[\underline{L}_k, \underline{L}_j] = \frac{(m_k - m_j)^2}{4(\sigma_k^2 + \sigma_j^2)} + \tfrac{1}{2} \ell n \left[\frac{\sigma_k^2 + \sigma_j^2}{2\, \sigma_j\, \sigma_j} \right]$$

where m_i, σ_i^2 are the respective mean and variances of the above conditional Gaussian densities. If this distance is less than a threshold, say ε, then the two sequences are merged into one. Typical useful values of ε range from 10^{-2} to 10^{-5}. Apart from eliminating the need to carry two sequences which are very close, this procedure permits carrying forward a sequence that would otherwise have been discarded.

Further Approximations

Following the approach of [6], we approximate the 2-D Kalman filter by a reduced update Kalman filter (RUKF). This constrained filter optimizes its update over a local update region $U_{\oplus+}$ at each observation pixel (n_1, n_2). This is illustrated in Fig. 1 below which also shows the global state support $S_{\oplus+}$.



global state region $\mathcal{S}_{\oplus+}(n_1, n_2)$

update region $\mathcal{U}_{\oplus+}(n_1, n_2)$

Fig. 1.

For each of the M space-variant RUKF's, error covariances must be stored and computed at each pixel. In order to avoid such complexity we calculate steady-state gains for each of the models and use these gains whenever the model appears in a particular model sequence $\underline{L}_j(n_1,n_2)$. The resulting composite filter is still space-variant because the gains switch from the steady state value of one model to that of the next as the scan progresses. Such an approximation is justifiable if model transitions occur far apart. In the case of edge models this basically means that the edge regions are long since the appropriate edge model will predict along the edge. In that case the gains reach their steady-state values before the model switches and increased error then occurs only in the transient portion.

EXPERIMENTAL RESULTS

We processed a noisy 256x256 pixel image with a relatively high SNR_i=12 dB. The original image is shown in Fig. 2a and is called *Lady*. It has been pre-smoothed to minimize the effects of scanner noise. The noisy image, shown in Fig. 2b, was processed by an RUKF and by the M-algorithm with M=5.

Our DSG model included 4 directional edge predictors and one 'isotropic' predictor as in the DSG random field model. Four prediction directions were judged adequate for predicting just one pixel ahead with low (1x1)-order models. The DSG model parameters were identified from the original noise-free image. The AR model for RUKF was also identified from the noise-free original. Closeups of the resulting output images are shown in Fig. 2c (RUKF) and 2d (M-algorithm). We note that the M-algorithm has produced a subjectively much better result. The mid-frequency background noise in the RUKF output has been suppressed and the edges are sharper in the M-algorithm estimate. The SNR improvement is 4.6 dB for the RUKF and 5.8 dB for the M-algorithm.

A detailed analysis reveals that the M-algorithm suppressed the noise in the 'isotropic' regions by 3 dB more than RUKF but that there was increased signal distortion. The net processing gain balanced out to 1.4 dB. In the edge regions the noise suppression and signal distortion are

Fig. 2.

approximately comparable with a slight 0.3 dB advantage for the M-algorithm in total error. Since most of the image is in the non-edge or 'isotropic' category, the overall net processing gain 1.2 dB is closer to 1.4 dB than to 0.3 dB.

ACKNOWLEDGEMENT

This paper is based on the Ph.D thesis of Subra Dravida. A longer version of this paper has been submitted for publication elsewhere [11].

REFERENCES

[1] R. Wallis,"An Approach to the Space Variant Restoration and Enhancement of Images", *Proceedings Image Science Math. Sympos.*, pp. 107-111, November 1976, Monterey, CA.

[2] J.S. Lim, "Image Restoration by Short Space Spectral Subtraction", *IEEE Trans. Acoust., Speech and Signal Process.*, Vol. ASSP-28, pp. 191-197, April 1980.

[3] J.S. Lee, "Refined Filtering of Image Noise Using Local Statistics", *Computer Graphics and Image Process.*, Vol. 15, 1981, pp. 380-389.

[4] V.K. Ingle and J.W. Woods, "Multiple Model Recursive Estimation of Images", *Proceedings ICASSP'79*, Washington D.C., pp. 642-645, April 1979.

[5] D. Preuss, "Two-Dimensional Facsimile Source Coding Based on a Markov Model", *NTZ*, Vol. 28, pp. 358-363, October 1975. See also *Proceedings ICC*. pp. 7/12-7/16, 1975.

[6] J.W. Woods and V.K. Ingle, "Kalman Filtering in Two Dimensions: Further Results", *IEEE Trans. Acoustics, Speech and Signal Process.*, Vol. ASSP-29, pp. 188-197, April 1981.

[7] J.W. Woods, "Two-Dimensional Kalman Filtering", in T.S. Huang (ed.) *Two-Dimensional Transforms and Filters*, Springer Verlag, Berlin, Chap. 7, pp. 155-205, 1981.

[8] J.W. Woods and I. Paul, "Adaptive Prediction DPCM Coding of Images", *Proceedings ICC-80*, pp. 31.8.1-.5, June 1980.

[9] G.A. Ackerson and K.S. Fu, "On State Estimation in Switching Environments", *IEEE Trans. Automatic Control*, Vol. AC-15, pp. 10-16, February 1970.

[10] J.K. Tugnait and A.H. Haddad, "A Detection Estimation Scheme for State Estimation in Switching Environments", *Automatica*, Vol. 15, pp. 477-481, 1979.

[11] J.W. Woods, S. Dravida and R. Mediavilla, "Image Estimation Using Doubly Stochastic Gaussian Random Field Models", submitted to *IEEE Trans. Pattern Anal. and Machine Intell.*

A CLASS OF BURST CORRECTING CODES

Mario Blaum*, Patrick G. Farrell** and Henk C.A. van Tilborg***

Abstract: The binary, linear code C, consisting of all $(k_1 + 1) \times (k_2 + 1)$ *binary matrices with even row and column sums, has length* $(k_1 + 1)(k_2 + 1)$, *dimension* $k_1 k_2$ *and minimum distance 4. So the code C is only one-error correcting. However if the bits are read out diagonally, the code can correct longer bursts. More precisely, assume (without loss of generality) that* $k_1 \geq k_2$. *Then C can not correct all bursts of length* $k_2 + 1$. *The code C can correct all bursts of length up to* k_2 *iff* $k_1 \geq 2(k_2 - 1)$. *An efficient decoding algorithm is presented for the* k_2*-burst correcting codes.*

1. INTRODUCTION

Figure 1 shows a simple "array code". It consists of all binary, $(k_1 + 1) \times (k_2 + 1)$ rectangles, with the property that every row sum and every column sum has even parity.

This code C is a binary, linear code of length $(k_1 + 1)(k_2 + 1)$ and dimension $k_1 k_2$. The last row and column can be considered as the places where the parity check bits are located.

The code C is capable of correcting a single random error, but can not correct two errors in the same row (or column). So the minimum distance of C is 4.

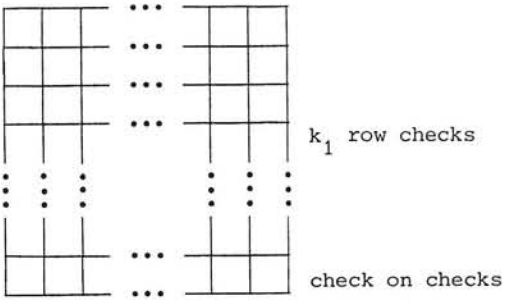


Figure 1:
A $(k_1 + 1) \times (k_2 + 1)$ two-dimensional code.

* IBM Almaden Research Center, San Jose, CA 95120-6099, USA.
** University of Manchester, Manchester, England.
*** Eindhoven University of Technology, Eindhoven, The Netherlands.

Without loss of generality we may assume that $k_1 \geq k_2$. It is known that array codes can correct bursts, if the digits are read out diagonally (see Figure 2).

0	17	14	11
4	1	18	15
8	5	2	19
12	9	6	3
16	13	10	7

Figure 2: Diagonal read-out 0,1,2,3,4,...,19.

On the other hand it is easy to see that an array code can not correct all burst patterns of length $k_2 + 1$. Indeed the burst 10...01 of total length $k_2 + 1$ starting at the position 1 has the same syndrome as the same burst starting at position $1 + (k_2 + 1)$ or at position $1 + 2(2k_2 + 1)$, etc..

It was conjectured [1] that an array code can correct any burst of length up to k_2, if and only if $k_1 \geq 2(k_2 - 1)$. We shall prove this conjecture, by means of a very efficient decoding algorithm.

2. RESULTS

First we have to say a little bit more about our notation. There are two ways of denoting a codeword in C. One is the array notation $(C_{i,j})_{0 \leq i \leq k_1, 0 \leq j \leq k_2}$. The second way is the vector notation $(c_0, c_1, \ldots, c_{n-1})$, $n = (k_1 + 1)(k_2 + 1)$. It reflects the diagonal read out. In the sequel $i \bmod n$ denotes the unique integer j, $0 \leq j < n$, satisfying $i \equiv j \bmod n$.

Lemma 1: $C_{i,j} = c_{f(i,j)}$, $0 \leq i \leq k_1$, $0 \leq j \leq k_2$, where

$$f(i,j) = (i - j)(k_2 + 1) + j \bmod n .$$

Proof: Working modulo n with the subscripts, the lemma easily follows from the following observations

$$C_{0,0} = c_0 ,$$

$$C_{i,j} = c_t \Rightarrow C_{i+1,j} = c_{t+k_2+1} ,$$

$$C_{i,j} = c_t \Rightarrow C_{i,j-1} = c_{t+k_2} .$$

□

The code C with the diagonal readout will be able to correct cyclic bursts. So we shall regard coordinates 0 and $n-1$ as neighbours. Often we need to know, how many coordinate positions $C_{(i,j)}$ and $C_{(i',j')}$ are apart in the corresponding codeword $\underline{c}$. The answer will be denoted by $\|(i,j) - (i',j')\|$ and will be called the distance between coordinates (i,j) and (i',j').

Corollary 2: Let $0 \leqslant i,\ i' \leqslant k_1 + 1$ and $0 \leqslant j,\ j' \leqslant k_2 + 1$. Then

$$\|(i,j) - (i',j')\| =$$

$$= \min\{(f(i,j) - f(i',j')) \bmod n\ ,\ (f(i',j') - f(i,j)) \bmod n\} .$$

Proof: This is a direct consequence of Lemma 1. □

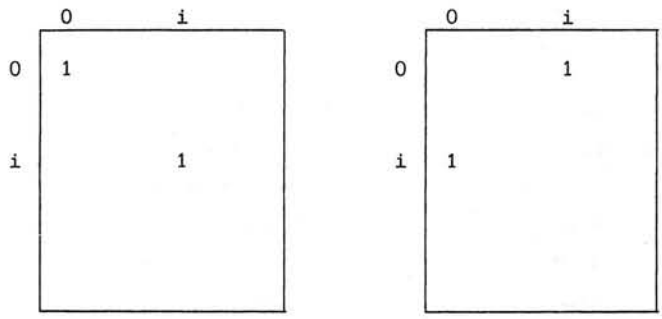
Lemma 3: A burst of length k_2 will never contain two positions in the same row or column.

Proof: Elements in the same column have a distance divisible by $k_2 + 1$. Elements in the same row have a distance $\min\{jk_2, n - jk_2\}$ for some j, $1 \leqslant j \leqslant k_2$. Since $k_1 \geqslant k_2$, it follows that $n - jk_2 \geqslant n - k_2^2 =$
$= (k_1 + 1)(k_2 + 1) - k_2^2 = (k_1 - k_2)k_2 + k_1 + k_2 + 1 > k_2$. □

Let h_i, $0 \leqslant i \leqslant k_1$, be the syndrome of the i-th row. So h_i is the modulo-2 sum of the elements in row i. It follows from Lemma 3 that we can replace the modulo-2 sum in the computation of the syndrome of a burst of length $\leqslant k_2$, by a summation over the integers. For the syndromes v_j, $0 \leqslant j \leqslant k_2$, the same holds. In other words "cancelation" of ones does not occur in these computations.

Theorem 4: If a $(k_1 + 1) \times (k_2 + 1)$ array code C, $k_1 \geq k_2$, can correct all possible bursts of length up to k_2, then $k_1 \geq 2(k_2 - 1)$.

Proof: Assume that $k_1 \leq 2(k_2 + 1)$. Consider the following two arrays of weight 2:



where $1 \leq i \leq k_2 - 1$. Clearly both arrays have the same syndrome. Also the first array is a burst of length $i + 1 \leq k_2$. With Corollary 2 one can deduce from the assumption $k_1 < 2(k_2 - 1)$, that for some value of i, $1 \leq i \leq k_2 - 1$, also the second array will be a burst of length $\leq k_2$ (see [2]). □

Theorem 5: Let $k_1 \geq 2(k_2 - 1)$. Then C can correct all bursts of length up to k_2.

We refer the reader to [2] for the proof and for a complete description of the decoding algorithm. Here we shall only demonstrate the algorithm for a "typical" example.

Example 6: Let $k_1 = 10$ and $k_2 = 6$ (so $n = 77$). Then $k_1 \geq 2(k_2 - 1)$. So this array code can correct bursts of length up to 6. Consider the syndrome depicted below.

	column 0	1	2	3	4	5	6	
row 0	0	71	65	59	53	47	41	0
1	7	1	72	66	60	54	48	0
2	14	8	2	73	67	61	55	1
3	21	15	9	3	74	68	62	1
4	28	22	16	10	4	75	69	1
5	35	29	23	17	11	5	76	0
6	42	36	30	24	18	12	6	0
7	49	43	37	31	25	19	13	1
8	56	50	44	38	32	26	20	0
9	63	57	51	45	39	33	27	0
10	70	64	58	52	46	40	34	0
vertical syndrome:	1	0	1	1	0	0	1	

horizontal syndrome

five consecutive zeros

If we regard the horizontal syndrome cyclically, we see a non-extendable sequence (called gap) of at least $k_1 - k_2 + 1 = 5$ consecutive zeros. This gap is unique because of inequality $k_1 \geq 2(k_2 - 1)$. Since there is no cancelation of ones, all the ones in the burst lie in rows 2-7. Row 2 is the first of these rows (if the gap were in the rows 3-9, then row 10 would have been the first). The left most column with syndrome 1, is column 0. We now claim that the burst with the syndrome above has a one in position (2,0), i.e. in coordinate 14. If this were not the case, row 2 would have a one in exactly one of the other columns and similarly column 0 would have a one in exactly one of the rows 3-7. But all these positions have distance at least 6, as can be easily seen from the figure above (this can of course also be proved formally). So no two of these positions lie in a burst of length 6. Hence we have proved that the burst has a one in position (2,0). In exactly the same way one finds the three other places, where the burst

has a one: (3,2), (4,3) and (7,6). The corresponding coordinate places are 14, 9, 10 and 13. So the burst starts at position 9 and has pattern 110011.

REFERENCES

[1] P.G. FARRELL & S.J. HOPKINS, Burst-error-correcting codes, The Radio and Electronic Engineer, 52 (1982) 182-192.

[2] M. BLAUM, P.G. FARRELL & H.C.A. VAN TILBORG, A class of burst-error correcting array codes, to appear in IEEE Trans. Info. Theory.

AN ERROR-CONTROL CODING SYSTEM FOR STORAGE OF 16-BIT WORDS IN MEMORY ARRAYS COMPOSED OF THREE 9-BIT WIDE UNITS

Wil J. van Gils*

ABSTRACT: Error-correcting codes are widely used to improve the reliability of computer memories. The shift of VLSI technology towards higher levels of integration has resulted in multiple-bit-per-card and multiple-bit-per-chip memory structures. This paper describes codes for storing 16-bit words in a memory array consisting of three 9-bit wide memory units, a unit being a single card or a single chip. These codes are able to correct single bit errors, to detect up to four bit errors and to detect the failure of a complete memory unit. The codes have an elegant structure which makes fast decoding possible by simple means.

1. INTRODUCTION

Single-error-correcting, double-error-detecting (SEC-DED) binary codes are widely used to increase the reliability of computer memories having a one-bit-per-chip or one-bit-per-card structure. However, the shift of VLSI technology towards higher levels of integration has resulted in multiple-bit-per-card and multiple-bit-per-chip memory structures. Frequently occurring error events in such memory arrays are single cell failures due to impingement of atomic alpha particles. These cause transient single bit errors. Less frequent are permanent errors due to single cell, row, column, row-column or complete chip failures. These can produce single bit errors, but may also cause multiple bit errors in a single chip output. Codes are therefore needed which correct/detect not only bit errors, but also errors caused by the failure of a complete chip or card.

*Philips Research Laboratories, P.O. Box 80.000,
5600 JA Eindhoven, The Netherlands.

This paper is concerned with the use of 9-bit wide memory chips in large memory arrays. Usually, such a chip is used to store bytes together with their corresponding parity bits. We describe the construction and use of a class of [27,16] binary linear codes that encode 16 data bits into 27 code bits, which are stored in three 9-bit wide memory units. In [3], a similar code is described. It can correct single bit errors, detect double bit errors, and detect the failure of a complete chip. However, this code is not optimal and its lack of structure requires a rather complex decoder.

We have constructed a class of [27,16] codes which can correct single bit errors, detect up to four bit errors and detect single memory chip failures. The codes constructed are optimal in the sense that there does not exist any [27,16] code having better correction-/detection properties. Our coding schemes also include simpler decoders using less hardware than the one described in [3].

In Section II we describe the construction and the properties of the codes. The decoders are described in Section III.

II CONSTRUCTION AND PROPERTIES OF THE CODES

Let α be a root of the primitive polynomial $x^8+x^4+x^3+x^2+1$. Hence, α is a primitive element of the Galois field $GF(2^8)$. Define β to be equal to α^{85}, $\beta := \alpha^{85}$. The finite field $GF(2^8)$ has sixteen normal bases, namely

$$N_b := \{ \alpha^{b2^i} \mid i=0,1,\ldots,7 \}$$

for $b \in B:=\{5,9,11,15,21,29,39,43,47,53,55,61,63,87,91,95\}$. For each of these normal bases N_b, we define the 8 by 8 binary matrix $M_b = \{ m_{ij}^{(b)} \}_{i=0\ j=0}^{7\ \ 7}$ by

$$\beta\alpha^{b2^i} = \sum_{j=0}^{7} m_{ij}^{(b)} \alpha^{b2^j} \qquad i=0,1,\ldots,7.$$

This means that the i^{th} row of M_b is the binary representation of $\beta\alpha^{b2^i}$ with respect to the basis N_b. The matrix M_b is a primitive element of the field GF(4), so that $M_b^3 = I$ and $I+M_b+M_b^2 = 0$, where I denotes the identity matrix and 0 denotes the all-zero matrix. Furthermore, it can be readily seen that the row (i+1) mod 7 of M_b^2 is equal to the i^{th} row of M_b (i=0,1,...,7). In [2] these matrices M_b were used to construct codes for the generalized Triple Modular Redundancy scheme. Here we shall use them to construct [3*9,16] codes.

Let $\underline{p}^T(A)$ for a binary matrix A denote the column vector of row parities of A, i.e. $p(A)_i = \sum_j a_{ij}$. Define C_b, $b \in B$ to be the binary linear [3*9,16] code with generator matrix

$$G_b := \begin{bmatrix} I & \underline{p}^T(I) & 0 & \underline{p}^T(0) & M_b & \underline{p}^T(M_b) \\ 0 & \underline{p}^T(0) & I & \underline{p}^T(I) & M_b^2 & \underline{p}^T(M_b^2) \end{bmatrix}.$$

We consider all codewords $\underline{c}$ in such a code to be composed of three symbols of nine bits: $\underline{c} = (\underline{c}_1, \underline{c}_2, \underline{c}_3)$, where $\underline{c}_1$, $\underline{c}_2$ and $\underline{c}_3$ all have length nine.

In terms of [1,4], the constructed codes have minimum (compound) distance profile (6,2,0). This guarantees correction of single bit errors, detection of single (9-bit) symbol errors and detection of up to four bit errors [2].

III ENCODER AND DECODER IMPLEMENTATION

The elegant structure of the codes makes fast decoding possible by simple means. A hardware realization of the encoder and the decoder will be presented [2].

REFERENCES

[1] W.J. van Gils, "A Triple Modular Redundancy Technique Providing Multiple Bit Error Protection Without Using Extra Redundancy", to appear in IEEE Trans. on Computers, 1986.

[2] W.J. van Gils, "An Error-control Coding System for Storage of 16-bit Words in Memory Arrays Composed of Three 9-bit Wide Units", to appear in Philips Journal of Research, 1986.

[3] IBM, European Patent Application publication no. 0100825.

[4] P. Piret, "Binary Codes for Compound Channels", IEEE Trans. on Information Theory, vol. IT-31, no. 3, pp. 436-440, May 1985.

ON POWERS OF THE DEFECT CHANNEL AND THEIR EQUIVALENCE TO NOISY CHANNELS WITH FEEDBACK

J.P.M.Schalkwijk*

Using Shannon's results on channels with side information at the transmitter, we will show that if the channel defects are known to the sender it is possible to replace the defect channel by an equivalent noisy channel with feedback. Feedback strategies for these noisy channels can now be translated into optimal codes for the original channel with defects. For the binary defect channel we can thus reliably transmit information at rates up to the channel capacity $C_\infty = 1-p$, where p is the expected fraction of defects.

INTRODUCTION

Consider a process that yields integrated circuit (IC) memory chips. This process is not perfect, i.e. individual memory cells have probability p of being defective. Fig. 1 gives a schematic representation of the generic memory cell, i.e. of the binary defect channel (BDC).

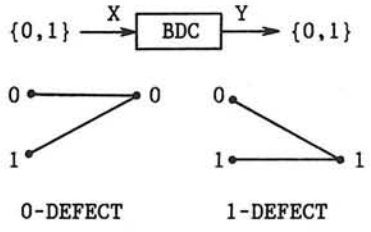


Fig. 1. Cell with unknown defect

*J.P.M.Schalkwijk is with the Eindhoven University of Technology, Department of Electrical Engineering, P.O.Box 513, 5600 MB Eindhoven, The Netherlands.

A binary random variable X is stored into the cell during the writing cycle. In the reading cycle we obtain the binary random variable Y, which in the ideal (p=0) situation always equals X. We distinguish, see Fig. 1, between 0-defects and 1-defects, i.e. between defective cells that always produce a "0" or a "1", respectively, when being read.

If the binary random variable X takes on the values 0 and 1 with equal probability, then the probability of a read error equals p/2, i.e. the memory behaves as a binary symmetric channel (BSC) with transition probability p/2. From Shannon's channel coding theorem we know that there exist codes that allow essentially error free transmission at rates up to the channel capacity

$$C_1 = 1-h(\tfrac{p}{2}) \text{ bits per memory cell,} \qquad (1)$$

where $h(x)=-x \log x-(1-x)\log_2(1-x)$ is the binary entropy function. Note that for $p=\frac{1}{2}$ we can store at most

$$1-h(\tfrac{1}{4}) = .18872 \text{ bits per memory cell.}$$

The remaining fraction, $\frac{1}{2}-.18872 = .31128$, of expected nondefective memory space is necessary to inform the reader about the location of the defects.

Up to now there was no loss in just treating the memory as a BSC with crossover probability p/2. The situation, however, becomes entirely different if we assume the locations and the values of the defects to be known to the (writer) sender. Instead of existance results concerning good codes, one can now use constructive feedback strategies [1], [2] to obtain reliable storage at efficiences up to C_1 bits per memory cell. For example, if $p=(3-\sqrt{5})/2$, one can use the optimal triple repetition code of [2]. However, there is more. In 1974 Kutznetsov and Tsybakov [3] obtained the remarkable result that now, with the sender knowing the defects, we can reliably store up to

$$C_\infty = 1-p \text{ bits per memory cell.} \qquad (2)$$

That is, asymptotically for large memory chips no good memory space has to be wasted in order to inform the reader about the defect locations! However, the Kutznetsov and Tsybakov result is, just like Shannon's channel coding theorem, an existance proof. In the present paper we will actually construct codes that yield reliable storage up to C_∞ bits per memory cell. Note that the feedback strategies [1], [2] mentioned above are constructive, but they only achieve storage efficiencies up to C_1. This is because these feedback strategies are non-anticipatory! They take into account what happens to the digit just being stored, but they do not take into account what will happen to the digits yet to be stored up to N=2,3,... time units into the future. To achieve C_∞ we have to anticipate into the future! For this we need Shannon's results [4] on channels with side information at the transmitter. These results of Shannon's will be described in the next section.

SHANNON STRATEGIES

Consider a finite collection $\{K_t = (A,[p_{ti}(j)],B) | t = 1,2,\ldots,h\}$ of channels. The generic K_t has inputs $i\varepsilon A = \{1,2,\ldots,a\}$, outputs $j\varepsilon B = \{1,2,\ldots,b\}$, and transition probabilities $p_{ti}(j)$, $t=1,2,\ldots,h$. On each successive transmission nature chooses one of these channels K_t independently at random with probability g_t, $t=1,2,\ldots,h$. One can distinguish three cases. In the first case neither the sender nor the receiver are aware of nature's choice K_t, $t=1,2,\ldots,h$. This amounts to having an equivalent channel $\bar{K} = (A,[\bar{p}_i(j)],B)$, with

$$\bar{p}_i(j) = \sum_{t=1}^{h} g_t p_{ti}(j),$$

connecting sender and receiver. In the second case both the sender and the receiver are aware of nature's choice K_t, $t=1,2,\ldots,h$. One is now able to reliably send information from sender to receiver at rates up to

$$\bar{C} = \sum_{t=1}^{h} g_t C_t,$$

where C_t is the capacity of channel K_t, t=1,2,...,h. We are interested in the third case, where the sender is aware of nature's choice K_t, t=1,2,...,h, but the receiver is not. This intermediate case is refered to as a channel $K = \{(K_t, g_t) | t=1,2,...,h\}$ with side information at the transmitter.

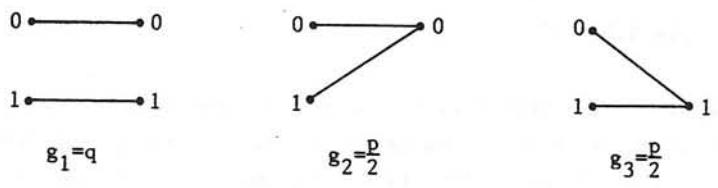
Shannon [4] now proves the existence of a derived channel $K' = (A^h, [r_X(y)], B)$,

$$r_X(y) = r_{x_1,x_2,...,x_h}(y) = \sum_{t=1}^{h} g_t p_{tx_t}(y), \qquad (3)$$

that has the following two properties. First, the capacity C' of K' gives the highest rate at which one can reliably transmit information over the original channel K with side information at the transmitter. Second, an optimum code for the derived channel K' can be translated into an optimum coding strategy for K, in that each input $X = (x_1, x_2, ..., x_h)$ of K' defines a function (strategy) from t to i for K. Further note that K' has a^h inputs and b outputs, but that only b inputs of K' are needed to achieve capacity. In the next section we apply the results of Shannon's to the BDC with known (to the writer) defects.

KNOWN DEFECTS

Consider the BDC with defects known at the (writer) sender as the channel K with side information of the previous section, see Fig. 2. The equivalent channel K' has $a^h = 2^3 = 8$ inputs and b=2 outputs. Only



two inputs, for example X_1=000 and X_2=111, of K' are required to achieve capacity. According to (3) we obtain for the crossover probability

$$r_{X_1}(1) = r_{000}(1) = \sum_{t=1}^{3} g_t P_{t0}(1) = (1-p).0 + \frac{p}{2}.0 + \frac{p}{2}.1 = \frac{p}{2}$$

Likewise $r_{X_2}(0) = r_{111}(0) = \frac{p}{2}$ and, hence, for K=BDC the derived channel K' is a BSC with crossover probability p/2. As the BDC is deterministic K' can be considered a BSC with noiseless feedback. Thus, the feedback strategies of [2] can be used to achieve capacity. In particular, if the probability of a defect equals $p=(3-\sqrt{5})/2$ one can use triple repetition coding.

Note that with K=BDC one obtains a maximum rate $C_1 = 1-h(\frac{p}{2})$ that could also be attained in the case of unknown defects. Now let us take advantage of the fact that we can anticipate on future defects, i.e. let $K=BDC^2$. As far as the defect locations are concerned we have four possibilities, to wit cc, cd, dc and dd, where c stands for "correct" and d for "defect". As each defect can be either 0 or 1, we have a total of h=9 component channels K_t, t=1,2,...,9, whose probabilities g_t are listed in the following Table. The equivalent channel

Table: component channel probabilities.

index	defects	probability
t=1	cc	$g_1 = q^2$
t=2	c0	$g_2 = \frac{1}{2}qp$
t=3	c1	$g_3 = \frac{1}{2}qp$
t=4	0c	$g_4 = \frac{1}{2}pq$
t=5	1c	$g_5 = \frac{1}{2}pq$
t=6	00	$g_6 = \frac{1}{4}p^2$
t=7	01	$g_2 = \frac{1}{4}p^2$
t=8	10	$g_8 = \frac{1}{4}p^2$
t=9	11	$g_9 = \frac{1}{4}p^2$

K' has thus $a^h = 4^9 = 262144$ inputs and b=4 outputs. However, only four inputs are required to achieve capacity and it is not that difficult to find a capacity achieving set $\{X_1, X_2, X_3, X_4\}$ of inputs for K'. Fig. 3 gives an input X_1 that mainly projects onto the output y=00.

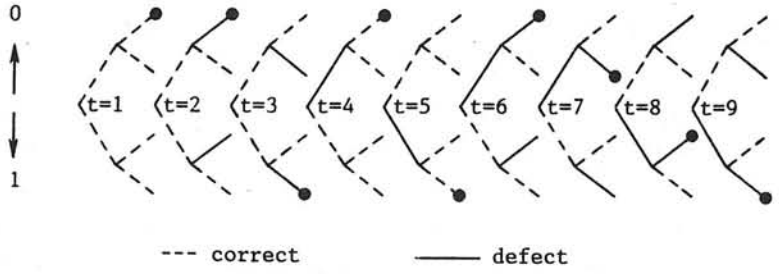


Fig. 3. The input $X_1=x_{11},x_{12},\ldots,x_{19}$ of K' that mainly projects onto y=00.

For each component channel K_t, i.e. for each defect pattern of the BDC², the input x_{1t}, t=1,2,...,9, that corresponds to X_1 is given by the fat leave of the corresponding t-tree, where upward branches correspond to a 0, downward branches to a 1, and solid branches to a defect. Note that X_1 minimizes $H(y|X_1)$, and thus for a symmetrical channel K' with a uniform input distribution the input X_1 maximizes $I(X_1;y)$ as it should at capacity. As the derived channel K' is, in fact, symmetrical one does not have to find X_2,X_3, and X_4 in order to compute the capacity C_2 of BDC². The transition probabilities leading away from input X_1 of K' are (from inspection of Fig. 3):

$$\begin{aligned} r_{x_1}(00) &= q^2+qp+\tfrac{1}{4}p^2 \\ r_{x_1}(01) &= \tfrac{1}{4}p^2, \\ r_{x_1}(10) &= \tfrac{1}{4}p^2, \\ r_{x_1}(11) &= pq+\tfrac{1}{4}p^2. \end{aligned} \qquad (4)$$

Using (4) we find for the capacity C_2 of BDC² in bit per transmission

$$C_2 = \tfrac{1}{2}\left(2-\{h(1-\tfrac{p^2}{2}) + (1-\tfrac{p^2}{2})\, h\left[\frac{(1-\frac{p}{2})^2}{1-\frac{p}{2}}\right] + \frac{p}{2}^2\}\right). \qquad (5)$$

Fig. 4 is a plot of C_1,C_2, and C_∞ versus the defect probability p.

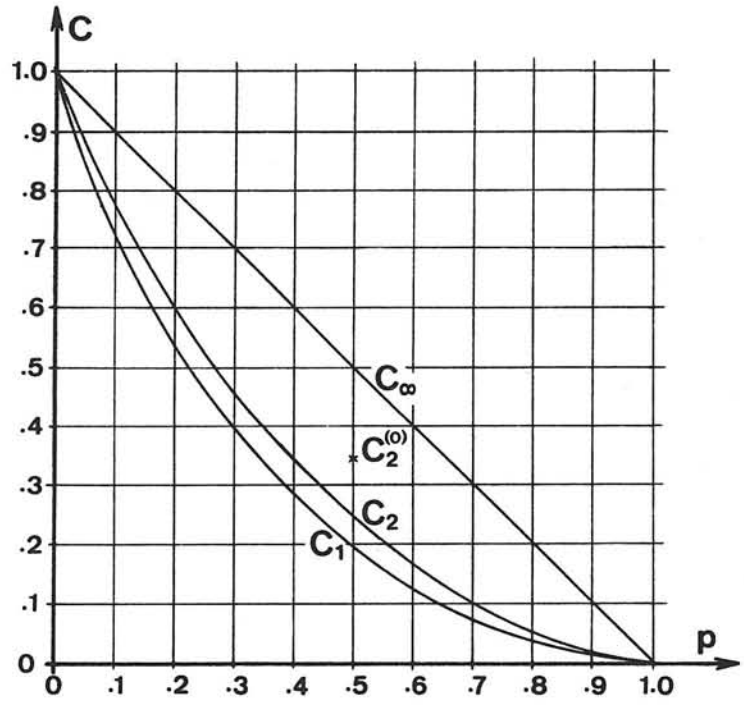


Fig. 4. Capacity versus defect probability.

For $p=\frac{1}{2}$ we find

$C_1 = .18872$ $\quad C_1^{(0)} = .32193$

$C_2 = .25434$

$C_3 = .27042$ $\quad C_2^{(0)} = .34150$

. .

. .

$C_\infty = .50000.$

Surprisingly, it is not hard to show that the capacity C_n in bit per transmission of BDC^n approaches C_∞ as $n\to\infty$. Also indicated in Fig. 4 is the capacity $C_2^{(0)} = .34150$ for $p=\frac{1}{2}$ of BDC^2 in the case where we have only 0-defects. The capacity $C_n^{(0)}$ for 0-defects only approaches $C_\infty=1-p$ somewhat faster as does the capacity C_n for equiprobable 0- and

1-defects. The capacity $C_n^{(0)}$, n=2,3,... , is somewhat harder to calculate as the resulting derived channel K' is not symmetric. In all cases the equivalent channel K' can be considered a discrete memoryless channel with noiseless feedback, where a multiple repetition feedback strategy as discussed in [5] can be used to achieve capacity. It is thus possible to find easily decodable optimal codes for the defect channel with know defects in a systematic manner!

CONCLUSIONS

Using Shannon strategies [4] we found easily decodable optimal codes for the discrete memoryless defect channel with known defects. In a similar way we can find codes for the bursty defect channel with known defects. The only effect of the bursty character of the defects is a change in the probabilities g_t, t=1,2,..., h, of the component channels K_t of the channel K with side information at the transmitter.

ACKNOWLEDGEMENT

The author wants to thank A.V.Kutznetsov and A.J.Vinck for telling him about the defect channel, and F.M.J.Willems for pointing out the significance of Shannon's side information paper. Thanks are also due to Ch.M.Bijl-Wind and H.M.Creemers for their help in preparing this manuscript.

REFERENCES

[1] M.Horstein, "Sequential transmission using noiseless feedback," IEEE Trans.Inform.Theory, vol.IT-9, July 1963, pp.136-143.

[2] J.P.M.Schalkwijk, "A class of simple and optimal strategies for block coding on the binary symmetric channel with noiseless feedback," IEEE Trans.Inform.Theory, vol.IT-17, May 1971, pp.283-287.

[3] A.V.Kutznetsov & B.S.Tsybakov, "Coding for memories with defective cells," Problemy Peredachi Informatsii, vol.10-2, pp.52-60, 1974.

[4] C.E.Shannon, "Channels with side information at the transmitter," IBM J.Res.Develop, vol.2, pp.289-293, Oct.1958. Reprinted in *Key Papers in the Development of Information Theory*, D.Slepian, Ed. New York: IEEE, 1974, pp.339-372.

[5] D.W.Becker, "Multiple-repetition feedback coding," Ph.D.Thesis, Dept.of Inform. and Comp.Science, Univ.of Cal.at San Diego, 1973.

REPEATED RECORDING FOR AN OPTICAL DISC

F.M.J. Willems* and A.J. Vinck*

We describe the repeated recording model for optical discs and design three codes that can be used in order to store information in a reliable way. The third code has a rate of 0.517 bit per spot.

I. INTRODUCTION

On an optical disc we can record information if we use a laser and an electromagnet. The laser heats up a spot on the disc and on this spot either a 0 or a 1 is stored, depending on the orientation of the magnetic field generated by the electromagnet. We can record binary information on such a disc if we properly reverse the current through the coil of the magnet. An optical effect makes it possible to read the 0-'s and 1-'s on the disc, and in this way the recorded information can be reproduced.

If we want to store on the disc in a short time a huge amount of information, the inductivity of the coil of the electromagnet will prevent us from reversing the current. Therefore we propose the following strategy : A "new" disc contains only 0-'s. If we store information on the disc for the first time, we write only 1-'s and thus it is not necessary to reverse the current direction. Before storing information on the disc for the second time we reverse the current and now we write only 0-'s. The third time we write only 1-'s etc. For reasons of simplicity we assume that each time that we record, we rewrite the entire disc.

The interesting feature of the above strategy is that the states of the spots of (a part of) the disc are known to the writer, before

---

* F.M.J. Willems and A.J. Vinck are with the Department of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven.

he records new information on this (part of the) disc. This state information can be used by the writer. The reader has no information about previous states of the spots on the disc.

Our problem now is to find "good" codes for the above model. With good codes much information can be stored reliably on the disc. Remember that during odd cycles only 1-'s may be written and during even cycles only 0-'s.

II. SOME CODES

In this section we describe some simple codes. These codes all have the property that with probability 1 the reproduced information is equal to the stored information.

A. The blocklength of the first code is 2 spots. During the odd cycles 01, 10, or 11 is written, during the even cycles 00, 01, or 10. The information to be stored can take on 2 values, A and B. The tables below contain what is written, as a function of the previous states of the 2 spots and the information to be stored.

	00	01	10
A	01	01	10
B	11	11	11

odd cycles

	01	10	11
A	01	10	01
B	00	00	00

even cycles

If during an odd cycle an A has to be stored, 01 is written if the previous states were 00 or 01, and 10 is written if the previous states were 10. During an odd cycle we always write 11 if a B has to be stored. Reproducing the information contained in the 2 spots is simple, 01 and 10 corresponds to an A, 11 to a B. Note that during an odd cycle it is impossible to write a 0 if the previous state was 1, during an even cycle it is not possible to change a 0 into a 1. We remark that what we write during the odd cycles are the previous states for the even cycles etc.

Per 2 spots we can store 1 bit of information with the code described above. Therefore the rate of this code is 0.5 bit per spot.

B. The code described under A reads the previous states and writes an odd-weight codeword if an A has to be stored and an even-weight codeword if a B has to be stored. Note that the number of 10 combinations on the disc can never increase. Because we have assumed that a new disc contains only 0-'s, the combination 10 will never occur. Therefore we obtain the more simple tables below.

	00	01
A	01	01
B	11	11

odd cycles

	01	11
A	01	01
B	00	00

even cycles

It will be clear that this code does not need to read the previous states anymore. This makes the implementation a lot simpler. In fact this code writes during the odd cycles a 0 or 1 (depending on A or B) on spot 1 and always a 1 on spot 2, and during the even cycles always a 0 on spot 1 and a 0 or 1 (depending on A or B) on spot 2. From inspecting this code we see that it does not only write information but it also prepares spots for the next cycle. We therefore can call this code a time-share code. One can easily determine the spots that contain information and the spots that are prepared.

The rate of the code above is again 0.5 bit per spot, it is still the same code as described under A.

The question now arises whether or not such time-share codes are optimal. Under C we will describe a code that demonstrates that time-share codes are not optimal. In this code storage and preparation are present in a diffuse form.

C. The blocklength of the third code is 5 spots. During the odd cycles we write codewords with weight 3, 4, or 5, during the even cycles we write codewords with weight 0, 1, or 2. We describe only the code for

the odd cycles. Note that the codewords on the disc before writing, the previous states of the spots, have weight 0, 1, or 2.

01111	A set of codewords
10011	(a basic set).
11100	

Consider the above set of 3 codewords. All 3 codewords have weight not less than 3. We now want to know whether or not it is possible to write at least one of these 3 codewords when the previous codeword has weight not more than 2. More precisely, can we always choose one of these 3 codewords such that we do not have to change a 1 into a 0 (this is impossible during odd cycles)? The answer to this question is yes.

If the previous codeword contains a 1 on spot 1 the first word in the (basic) set (01111) can not be written since it has a 0 on spot 1. If the previous codeword contains a 1 on spot 2 or spot 3 we can not write the second word in the basic set. A previous codeword with a 1 on spot 4 or 5 eliminates the third codeword. Since a previous codeword has weight not more than 2, at most 2 codewords from the basic set are eliminated. The set contains 3 codewords and therefore at least 1 of these codewords can be written without changing a 1 into a 0.

We can say that the above set of three codewords "covers" all possible (weight not more than 2) previous codewords. If we now partition all words of weight not less than 3 in sets that cover all weight not more than 2 codewords, we obtain a code. As first set we take the basic set (01111,10011,11100). By permuting the columns of this basic set, we find 4 more sets of codewords that cover all previous codewords. Then only the word 11111 remains. This codeword however covers, on its own, all previous words and therefore forms a sixth set. With these 6 sets we can now store one out of six information symbols (A, B, C, D, E, and F) using 5 spots (see table on next page).

The rate of this code clearly is $\frac{1}{5}\log_2 6 = 0.517$ bit per spot.

A-set:	B-set:	C-set:
01111 (1)	10111 (2)	11011 (3)
10011 (23)	01101 (14)	01110 (15)
11100 (45)	11010 (35)	10101 (24)

D-set:	E-set:	F-set:
11101 (4)	11110 (5)	11111 (x)
01011 (13)	00111 (12)	
10110 (25)	11001 (34)	odd cycles

By inverting the code for the odd cycles we obtain the code for the even cycles. Note that the above code is rather good in the sense that the basic set and its 4 permutations exactly partition the set of words with weight 3 or 4. It is unknown whether or not there exist basic sets for higher blocklengths with the same property. Presently Erik Kwast is investigating this.

III. REMARKS

Using Shannon-theoretic arguments it is possible to show that rates higher than $\log_2((1+\sqrt{5})/2) = 0.694$ bits per spot can not be realized with reliable codes.

Furthermore it can be shown that rates arbitrarily close to 0.694 can be achieved with codes that have an arbitrily small but positive error probability. Recently John van Breemen has designed some codes of this type.

It should be noted that there is a close relationship between the configuration studied here and the Blackwell broadcast channel studied by multi-user information theorists.

ACKNOWLEDGEMENT

We thank Stan Baggen for proposing the problem investigated here to us. It is because of Krista that this contribution does not contain the proof of the fact that rates higher than 0.694 can not be realized.

THE UNCERTAINTY PRODUCT VERSUS THE SUM OF ENTROPIES UNCERTAINTY PRINCIPLE

C. Kamminga

The well-known form of the uncertainty relation as introduced by Gabor states that if time duration Δt of a signal and the frequency width of its Fourier transform are defined by their variances in the time and in frequency domains, then $\Delta t.\Delta f \geq \mu$. An extension of this uncertainty principle was obtained by Leipnik where the sum of entropies of two distributions are related as the absolute squares of a Fourier transform pair. In this presentation a few computational examples illustrating the relationship between the two uncertainty principles are given for different types of real signals.

1. INTRODUCTION

Perusing the amount of literature that nowadays exists on the classical Heisenberg/Weyl uncertainty relations and its extensions, we are faced with detailed mathematical insights, but surprisingly few applications in the field of signal processing, other than measurement uncertainties.

In this paper, two forms of the uncertainty principle and their relationship will be given for the case of a practical situation, the so-called echolocating signal of dolphins. The dolphin sonar signal turns out to be a natural optimal signal, i.e. it approaches the theoretical lower bound of the uncertainty in both the time and frequency domains. The result is further enhanced by the suggestion that the receiver - in this case the cochlea - is not primarily designed to handle sinusoids optimally, but rather processes them non-optimally. From the point of view of the cochlea, following BARRET (1978), the optimum rôle is filled by an elementary signal that is bounded by the relation $\Delta t.\Delta f$ is minimal, where Δt represents the signal duration and Δf the signal bandwidth. The concepts of duration in time and frequency need to be

---

C. Kamminga is with the Delft University of Technology, Department of Electrical Engineering, Information Theory Group, P.O. Box 5031, 2600 GA Delft, the Netherlands

specified, but at present the quantification is done empirically. The equation of Heisenberg and Weyl introduces the variance as a measure of the uncertainty. This approach was also followed by GABOR (1946), who introduced the uncertainty relation in communication theory, using the formalism of quantum mechanics. ZAKAI (1960) introduced a generalisation of the definition of 'duration' which includes the Shannon information measure. Although this approach extends over several definitions of durations, the uncertainty relationship that follows from it has limited physical application and clarity.

A more complete result for an uncertainty relation based on Shannon's information measure was given by Leipnik (1959). In particular, we note here an uncertainty relation comprising of a summation of information quantities, linked to the pair of Fourier transforms of the original function. This is in contrast to Zakai's product relationship for time and frequency duration. It is not surprising that the optimal function minimising both relations is the same: the Gaussian function.

2. GABOR'S UNCERTAINTY RELATION

Although the frequency-time uncertainty principle has been known since the early thirties, it was GABOR (1946) who gave it a general interpretation. We characterized the frequency-time uncertainty product in the context of structural information theory. The following definition of the duration of a time-limited function s(t) is used:

$$\Delta t^2 = \frac{\int_T (t-t_0)^2 \; s^2(t)\,dt}{E} \quad \text{with } t_0 = \frac{\int_T t s^2(t)\,dt}{E}$$

$$\Delta f^2 = \frac{\int_\Omega (f-f_0)^2 \; S(f)^2\,df}{E} \quad \text{with } f_0 = \frac{\int_\Omega f s(f)^2\,df}{E}$$

Without loss of generality normalisation is done by setting E=1. From these quantities it is possible to obtain the GABOR uncertainty relation in the rigorous form

$$\Delta t.\Delta f \geq \mu.$$

A problem that arises quite naturally is concerned with what shape of

s(t) the inequality turns into an equality. The smallest possible uncertainty product is obtained for a Gaussian function

$$s(t) = e^{-\alpha^2 (t-t_o)^2} \; e^{2\pi j f_0 t},$$

which is a harmonic function, modulated by a probability envelope.
If we set $\Delta t . \Delta f = 1$
then $t = \frac{\sqrt{\pi}}{\alpha}$.

The function that satisfies the equality is called an elementary signal, and it covers the minimum effective area in the time-frequency plane.

3. LEIPNIK'S UNCERTAINTY RELATION

With regard to the formulation by Zakai, a more complete result using Shannon's information measure in an uncertainty relationship was given by Leipnik. From the definition

$$H_t = - \int_T s^2(t) \; \ell n \; s^2(t) \; dt$$

and its analogous form H_f, it can be shown that H_t and H_f are related by the following summation:

$$H_t + H_f \geq \ell n \frac{e}{2} .$$

The interested reader is referred to the original article for the complete mathematical treatment.

As expected, the minimising waveform is again a Gaussian, as in Gabor's version of the uncertainty principle:

$$s(t) = \sqrt[4]{2} \; e^{- t^2} .$$

A closer examination of the 'information' uncertainty relation leaves us with a question regarding a relationship between Gabor's product and the sum of information quantities from Leipnik.
Not unexpectedly, the latter equation reveals the additive nature of

information.

An interesting point arises if we calculate both uncertainties and look at the corresponding behaviour of time functions which approach the lower bound of the inequality.

4. EXAMPLES

We refer to figure 1 which shows the relationship between $\Delta t.\Delta f$ and H_t+H_f for three different types of signals:

I A sinusoid increasing in time duration, starting with a one-period signal ℓ1.
A certain compression with increasing number of cycles is noticed in the H_t+H_f values.

II A frequency modulated sinusoid with increasing number of cycles starting with the one-period signal f1.

III Gaussian signals
g_1 - Gaussian pulse
g_2 - a sinusoid with Gaussian envelope
g_3 - a frequency-modulated sinusoid with Gaussian envelope.
Note the difference between the points g_1 and g_2; they posses the same $\Delta t.\Delta f$ value but show a different behaviour when the information sum is used.

After examining these test signals and their responses, an interesting point arises if we input near-optimal signals such as the echolocation sonar signals of the dolphin. We refer to figure 2 for a comparison of several dolphin signals 1 to 6. The cluster of data points 2, 3, 4 and 6 does suggest an investigation of the behaviour of the lower values of the uncertainty relation. At first sight the difference between g_1 and g_2 could be used. A Gaussian modulated sine wave does fit more adequately into the description then the sole Gaussian pulse. Therefore, there might be a slight preference for the Leipnik measure of uncertainty on the basis of intuition. If we consider the original formulae of Gabor and Leipnik, we note that Leipniks measure uses the complete time function, while the Gabor uncertainty is based on the

second moments description of duration. Despite the quite complete description of a dolphin signal in terms of structural information, we are still left with the following interesting question: how close is this signal to the lower bound, as for example in the case of the Irrawadi dolphin, who's signal is characterised by a $\Delta t.\Delta f = 1.09$? (KAMMINGA et al. 1983).

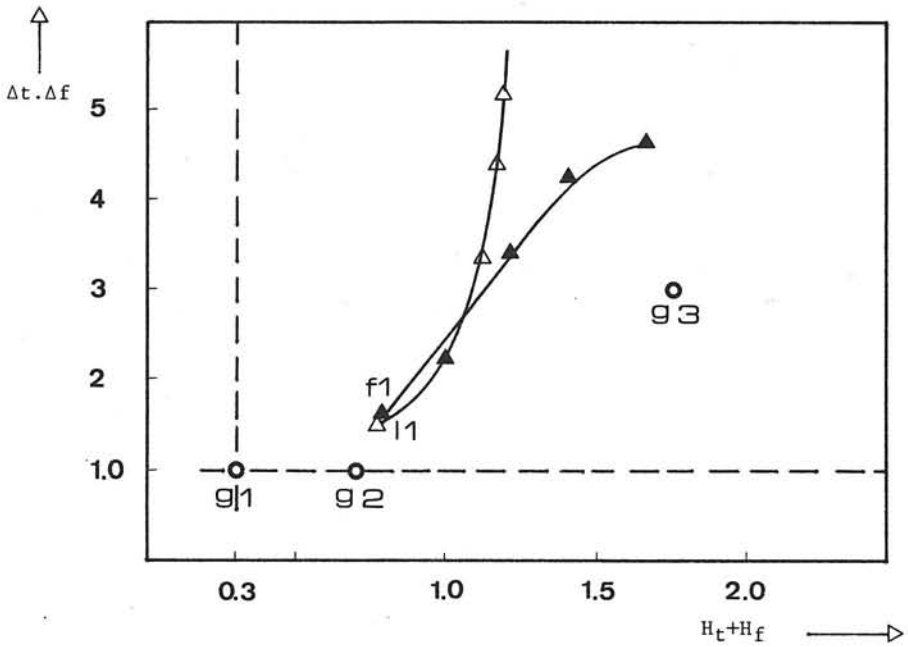


Figure 1. The relationship between $\Delta t.\Delta f$ and H_t+H_f for the different types of test signals I, II and III together with the Gaussian signals.

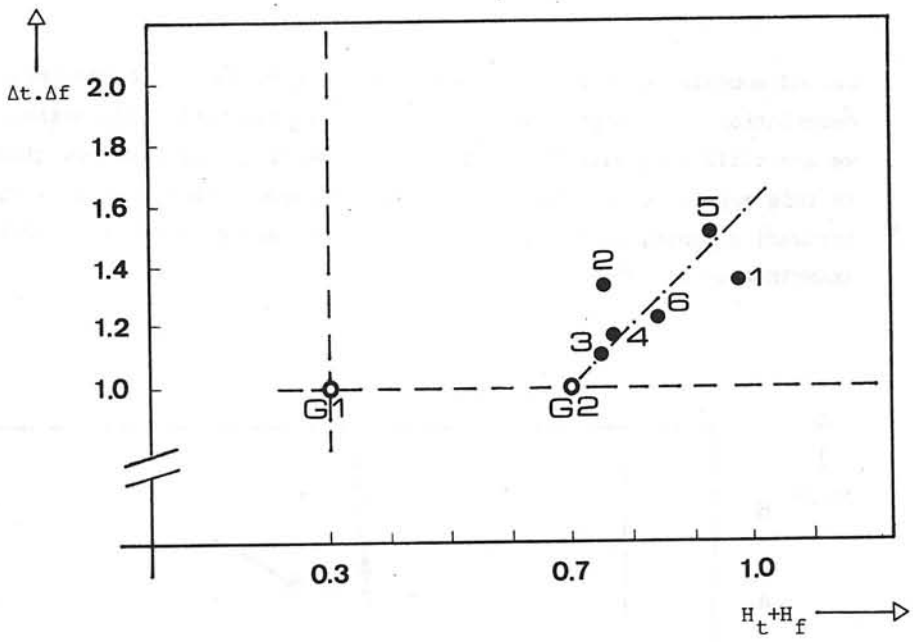


Figure 2. The relationship between $\Delta t.\Delta f$ and H_t+H_f for different dolphin signals. Dotted lines indicate lower bounds for the uncertainty product as well as the sum of entropies.

REFERENCES

[1] HEISENBERG, W. (1927), The actual content of quantum-theoretic kinematics and dynamics, Z. Physik 43, 172-198.

[2] WEYL, H. (1928), *Theory of Groups and quantum mechanics*, 77, 393-394. Dutton, New York.

[3] BARRET, T.W. (1978), The cochlea as Laplace analyzer for elementary signals, Acustica 39, 155-173.

[4] GABOR, D. (1946), Theory of Communication, J. Inst. Elec. Eng. 93, Part III, 429-439.

[5] ZAKAI, M. (1960), A class of definitions of duration, Information & Control, 3, 101-115.

[6] LEIPNIK, R. (1959), Entropy and the uncertainty principle, Information & Control, 2, 64-79.

[7] KAMMINGA, C., H. WIERSMA, W.H. DUDOK VAN HEEL and TAS'AN (1983), Investigations on cetacean sonar VI. Sonar bounds in Orcaella brevirostris, first descriptions of acoustic behaviour. Aq. Mamm. 10, (3), 83-95.

INVERSE SYNTHETIC APERTURE RADAR (ISAR)

Gerard A. van der Spek*

A radar system which observes a moving air target will provide the user with limited information. Normally this information concerns position versus time and echo strength, which offer little to characterize the target. The resolution of the radar system is determined by the bandwidth of the transmitted signal (range) and by the beamwidth of the antenna combined with the radial distance to the target (cross-range). If the resolution can be reduced, in one or more dimensions, to a fraction of the size of the observed object it will be possible to obtain an "image".

A one dimensional image can be obtained by the ISAR-technique. An aircraft is tracked by a coherent radar with a pencil beam and echoes are obtained during several seconds at a sufficiently high repetition rate.
The radial speed of different aircraft parts will depend on path and speed of the aircraft. Corresponding doppler shifts will be observed in the Fourier spectrum of the echo series, which can be interpreted as a projection of the reflectivity of the aircraft on an axis which is perpendicular to the line of sight and to the (apparent) rotation axis of the aircraft.

The ISAR technique will be discussed and results will be presented which are obtained for several civil aircraft.

*FEL-TNO, P.O. Box 96864, 2509 JG The Hague, The Netherlands

ERROR SENSITIVITY OF COMPRESSED IMAGE DATA ON SATELLITE COMMUNICATION LINKS*

H.J. Simons**

This paper describes the effect of transmission errors present on satellite communication links, on compressed facsimile data and on uncompressed and compressed earth observation image data. The error statistics of the ECS/SMS channel are determined and the effect of these errors on the users data is evaluated.

1. INTRODUCTION

With the availability of wideband satellite links, it becomes possible to transmit high volumes of data to remote users very fast. The satellite transmission system of APOLLO [1] is an example of a high speed digital transmission system, specially designed to handle long data messages such as: page facsimile and earth observation image data.

The APOLLO system is a switched digital network based on the ECS part of the Satellite Multiservice System (ECS/SMS) of EUTELSAT. Although the ECS/SMS system is designed for continuous mode of operation, the APOLLO system operates in burst transmission mode, where different earth stations share the same carrier in time division, using a demand-assigned sequential satellite access technique. Each earth station can be shared by different data stations.
The earth stations assemble incoming data into blocks, according to the High-level Data Link Control (HDLC) framing structure, as shown in figure 1, with a maximum length of 32 kbits.

* This investigation was carried out under ESTEC Contract no. 6235/85/NL/JS
** NATIONAL AEROSPACE LABORATORY (NLR) Informatics Division, Anthony Fokkerweg 2, 1059 CM Amsterdam, The Netherlands

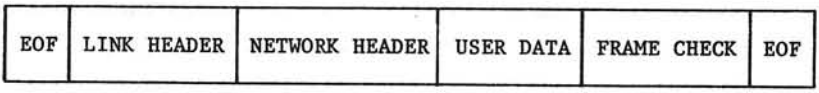


Fig. 1 APOLLO data frame structure

The beginning and end of each data frame are indicated by the unique sequence 01111110 called the End-of-Frame (EOF) sequence. To prevent that this sequence occurs somewhere in the data, the HDLC transmit controller inserts a 0 every time 5 consecutive 1's appear in the data. The HDLC receive controller removes all 0's after 5 consecutive 1's. To several data frames a synchronization preamble and a proper postamble are added, to form a transmit data burst.
To a data burst, framing and signalling bytes are added, according to the ECS/SMS frame/multiframe structure [2]. Then all data except the framing bytes are scrambled and the total data stream is protected by a forward error correcting (FEC) rate ½, constraint length 7, convolutional code, with the input differentially encoded. The encoded data stream is QPSK modulated. The receiver has to perform QPSK demodulation, carrier recovery, bit timing, soft decision Viterbi decoding and code synchronization. After frame and multiframe synchronization have been achieved, the message bytes can be descrambled and the framing bytes can be removed.
Two modes of operation are foreseen. The one is a connection oriented mode in which erroneously received data frames can be retransmitted, resulting in a virtually error free data transmission. The other is a connection less mode in which the user receives data including errors. In this paper the effect of transmission errors in connection less mode, on the users data will be identified. For this purpose the data types chosen are compressed facsimile data and uncompressed and compressed Thematic Mapper (TM) image data.

2. LINK ERROR CHARACTERISTICS

At the output of the receiver, several types of error may be present. The most relevant are errors due to loss of carrier and bit timing synchronization, loss of Viterbi decoder node synchronization, FEC decoder decoding errors and loss of frame or multiframe synchronization.

The ECS/SMS performance requirements specify a bit error rate (BER) of less than 10^{-6} for 99 % of the time, at an E_b/N_0 of 6.1 dB at the input of the demodulator. Under these normal conditions the loss of bit timing, decoder or frame synchronization are negligibly small. Therefore only the effects due to FEC decoder decoding errors will be considered.

It is well known that after Viterbi and differential decoding, the error sequence consists of error bursts separated by error-free guard-spaces. In this context a guard-space is defined as a sequence of at least 6 consecutive correct bits. This length is chosen because an ideal constraint length 7 Viterbi decoder is known to be on a correct path again after 6 consecutive correct bits. A burst is a sequence of consecutive bits not containing a guard-space and preceded and succeeded by a guard-space.

From simulations and measurements with the Viterbi decoder, bit error probabilities are known as function of E_b/N_0 [3] and also average burst statistics are known as function of E_b/N_0 [4], [5]. Here E_b/N_0 is the baseband signal energy to noise ratio at the input of the Viterbi decoder.

Because the data also is differentially decoded, the error statistics changes again. A differential decoder introduces one extra error for each isolated error at its input. However, to an error burst only one error is added at the end and the distribution of errors inside a burst will be changed.

Table 1 gives the error burst statistics after Viterbi and differential decoding for different bit error rates at the output of the differential decoder. For more details refer to [6].

TABLE 1

Error burst statistics after Viterbi and differential decoding

Bit error rate	P	10^{-5}	10^{-6}	10^{-7}
average burst length	$\bar{\ell}$	7.7	6.9	6.6
Burst error probability	P_B	$2.1*10^{-6}$	$2.3*10^{-7}$	$2.3*10^{-8}$

With these statistics the average number of error bursts in a sequence of M bits, with M very large, can be approximated by $M.P_B$ and the probability that a sequence of λ bits, with λ small ($\lambda.P_B << 1$), is hit by an error burst can be approximated by $(\bar{\ell}+\lambda-1)P_B$.

When an error burst corrupts part of the transmitted data, such a burst can have different effects on the users data, dependent on the position where it occurs and the sequence that results. Four different error types can be distinguished which are:

- Substitution errors, when transmitted bits are incorrectly received.
- Bit insertion errors, when a pattern of five consecutive bits is corrupted, such that the HDLC receive controller does not remove the inserted bit.
- Bit deletion errors, when any data pattern is corrupted in such a way that five consecutive 1s preceded and followed by a 0 result, such that the HDLC receive controller wrongly will delete the final 0.
- End-of-Frame (EOF) simulation errors, when any data pattern is corrupted in such a way that six or seven consecutive 1s result, such that the current HDLC frame will be terminated.

It can be shown [7] that, assuming random user data, the HDCL transmit controller inserts a 0 on average after 62 user bits. A bit insertion error occurs if either one of the preceding 1s is hit by an error burst. Therefore $Pr\{\text{bit insertion error}\} = (\bar{\ell}+4)P_B/62$.

For any error burst, there are $(\bar{\ell}+4)$ different 7 bit sequences which can be corrupted into 0111110 and for each of these possible

candidates there is only one out of the 128 7 bit sequences which realy can generate this sequence. Therefore Pr{bit deletion error} = $(\bar{\ell}+4)P_B/128$.

For any error burst, there are $(\bar{\ell}+5)$ different 8 bit sequences which can be corrupted into 6 or 7 consecutive 1s, and only 4 of the 256 possible 8 bit sequences realy result in the mentioned sequence. Therefore Pr{EOF simulation error} = $(\bar{\ell}+5)P_B/64$.
Table 2 shows the different error type probabilities at a BER of 10^{-6}.

TABLE 2
Statistic of different error types

P_b	P_B	Pr(insertion)	Pr(deletion)	Pr(EOF simulation)
10^{-6}	$2.3*10^{-7}$	$4.0*10^{-8}$	$2.0*10^{-8}$	$4.3*10^{-8}$

3. ERROR SENSITIVITY OF COMPRESSED FACSIMILE DATA

For the compression of ISO A4 pages containing only black and white information, the CCITT has recommended a one-dimensional (1-D) modified Huffman coding scheme and a two-dimensional (2-D) modified READ coding scheme [8].
An uncompressed high resolution (HR) A4 page (7.7 lines/mm x 8 pels/mm) consists of 2376 lines x 1728 pels/line or approximately 4 M bits. From measurements on several reference documents [9] it was shown that the average compression factor for 1-D compressed pages was 7.7, resulting in a datacontent of $5.3*10^5$ bits for a HR page. The average compression factor for 2-D compressed pages was 11.5 resulting in a datacontent of $3.6*10^5$ bits.

When an error burst corrupts a part of the transmitted data this can have different effects on the users data, dependent on whether the HDLC frame header or user data is corrupted. When the compressed user data is corrupted by a substitution, bit insertion or de-

letion error, codeword synchronization will be lost. Each codeline is terminated by a unique End-of-Line (EOL) codeword, which is constructed in such a way that it also can be recovered after loss of codeword synchronization. Therefore in a 1-D compressed page only one line will be incorrect due to an error burst. In a 2-D compressed page also all following lines will be decompressed incorrectly, until a 1-D compressed line is recovered. To prevent error propagation, the CCITT has recommended to insert a 1-D compressed line once every 4 lines.

The average number of error burst per page, at a BER of 10^{-6}, is once every 8 pages for a 1-D compressed page and only once every 12 pages for a 2-D compressed page.

Most of the error bursts only lead to the loss of up to a few lines, which in general is not very catastrophic. However, when an EOF simulation error occurs, an average half a 32 kbits frame will be lost, which corresponds to approximately 75 lines in a 1-D compressed page and to 110 lines in a 2-D compressed page. Synchronization will again be recovered, but the loss in general is unacceptable. At a BER pf 10^{-6}, EOF simulation errors occur, an average, once every 44 respectively 65 pages for 1-D respectively 2-D compressed pages.

When an error burst hits the HDLC frame header, which has a length up to 300 bits, the total frame will be rejected, which also is catastrophic to the page. At a BER of 10^{-6} this occurs only once every 1000 pages on average and thus is negligible compared with EOF simulation errors.

4. ERROR SENSITIVITY OF UNCOMPRESSED AND COMPRESSED IMAGE DATA

An uncompressed Thematic Mapper (TM) 1/4 scene consists of 2880 x 3460 8 bits pels, or approximately 80 Mbits. The data is arranged in lines, starting with a lineheader for identification and synchronization.

At a BER of 10^{-6}, on average 18 error bursts corrupt the image data. Of these, on average 4 or 5 are bit insertions or deletions

resulting in the loss of codeword synchronization. When the line-header contains a robust synchronization pattern, line sunchronization will be recovered and only one line will be lost. On average 3 or 4 of the errors in a 1/4 scene are EOF simulation errors resulting in a loss of on average half a frame, or approximately 2000 pixels, which lead to the loss of 1 or 2 lines. It may be obvious that an exhaustive synchronization procedure is required to recover after such errors. The remaining 10 substitution errors only lead to several incorrect pixel values.

When the data is compressed with a DPCM compression scheme, in general only moderate compression factors of 1.5 will be achieved on average. At a BER of 10^{-6}, there will be 12 error bursts in such an image. The data is arranged in lines, starting with a line-header, analogous to the uncompressed data format. Of the error-bursts, 10 are substitution, bit insertion or deletion errors, all resulting in the loss of codeword and line synchronization, which can be recovered from the lineheader again. In 1-D DPCM only 1 line is decompressed incorrectly, but in 2-D DPCM, the errors propagate to the following lines until a 1-D compressed line is recovered.
Approximately 2 of the error bursts are EOF simulation errors, leading to the loss of 1 or 2 lines in a 1-D compressed image. In a 2-D compressed image the errors again propagate to the following lines. To limit this propagation, it is recommended to insert a 1-D compressed line once every 10 lines.
It can be seen that, although less errors do occur in a DPCM coded image, more of these errors are visible than in an uncompressed image.

When larger compression factors are required, and when small reconstruction errors can be tolerated, a transform coding algorithm like the Chaturvedi algorithm [10] can be used. The data is arranged in compressed 8 x 8 blocks, where each block is assigned a header which contains a block number and a block length indicator. Furthermore, 256 (or less) blocks form a segment, which is indicated by a segment header, containing a synchronization sequence and segment identifications.
The obtainable compression factor depends very much on the image

entropy and on the tolerable reconstruction error. In the following a compression factor of 4 will be assumed.

At a BER of 10^{-6} the transmitted data for one image contains 5 error bursts on average, of which 4 fall in the compressed data and 1 hits a block header. Of the error bursts in the data, 2 are substitution errors, resulting in an incorrectly decompressed 8 x 8 block. However, from the block header, block boundaries are known, such that there is no error propagation over block boundaries. 1 of the error bursts is either a bit insertion or a deletion error, resulting in an incorrectly decompressed block and a loss of block boundaries since the next block is 1 bit out of position. This results in incorrectly decompressed blocks until the end of the current segment, where synchronization will be recovered again. The error burst which hits the block header has the same effect of loosing block boundaries until the end of the segment, which is visible as a beam in the image of 8 pixels wide and on average 1000 pixels long. The remaining error burst in the data is an EOF simulation error, which leads to the non-acceptance of approximately 100 blocks on average. Blocks will be decoded incorrectly until the first segment header is recovered. Up to 2 segments may be affected, which is visible as a beam in the image of 8 pixels wide and up to 4000 pixels long, which is only a limited area considering the size of the image.

5. CONCLUSIONS

In this paper the effect of transmission errors on compressed facsimile data and on uncompressed and compressed image data have been evaluated. The calculated error statistics show that the burstyness of the errors is an advantage, since it reduces the number of error events.

For the transmission of facsimile pages it was shown that catastrophic errors occur only once every 44 or 65 pages for 1-D respectively 2-D compression, at a BER of 10^{-6}.

For the transmission of image data it was shown that the number of

errors in each image can be largely reduced by introducing compression, however, the effect of errors is larger in a compressed image. Furthermore it was shown that certain errors lead to the loss of synchronization, which can only be recovered if the data has its own synchronization structure.
At a BER of 10^{-6} an uncompressed image contains 8 line errors, a 1-D DPCM compressed image contains 12 line errors and a transform coded image (CF = 4) contains 2 incorrect 8 x 8 blocks and 3 beams of 8 pixels wide of incorrect blocks, all on average.

It is believed that for transmission of relatively short data messages, such as in facsimile, a BER as high as 10^{-6} can be acceptable. For longer data messages the images may be acceptable, dependent on the requirements of the user, but in general a smaller BER will be required.

It should be noted that the calculations at a BER of 10^{-6} are worst case situations, which occur only in bad weather conditions. For most of the time the BER will be 10^{-7} or even 10^{-8}, resulting in a reduction of the number of errors by a factor of 10 respectively 100 and decompression errors than occur only very infrequently.

6. REFERENCES

[1] APOLLO, System Requirements Specification, ESA-SP-1068, August 1984.

[2] ECS Multiservice System Specification, EUTELSAT, Document ECS/C21-20, Rev2E, May 1983.

[3] J.J. SPILKER Jr, Digital Communications by Satellite, Englewood Cliffs: Prentice Hall, 1977.

[4] H.F.A. ROEFS & M.R. BEST, *Concatenated Coding on a Spacecraft-to-Ground Telemetry Channel: Performance*, International Conference on Communications, Denver, Colorado, 14-18 June 1981.

[5] S.J. CURRY & W.D. HARMON, *A Bound on Viterbi Decoder Error Burst Length*, Proceedings of the International Telemetering Conference, Los Angeles, California, September 1976.

[6] H.J. SIMONS, Vulnerability of Low Redundancy Image Data and Text Transmission over Satellite Communication Links, Report TR 85100 L, National Aerospace Laboratory, The Netherlands, July 1985.

[7] J.S. MA, *On the Impact of HDLC Zero Insertion and Deletion on Link Utilization and Reliability*, IEEE Tr. on Communications, Vol. COM-30, No. 2, February 1982.

[8] CCITT Recommendation T4, Fascile VII. 2, Geneva 1980.

[9] R. HUNTER & A.H. ROBINSON, *International Digital Facsimile Coding Standards*, Proceedings of the IEEE, Vol. 68, No. 7, July 1980.

[10] W.C. HUISMAN, *Three Image Compression Algorithms for CADISS*, 5th Symposium on Information Theory in the Benelux, Aalten, 24-25 May 1984.

$$R \triangleq E(\lg(c_k^*))/L, \tag{4}$$

where the enteger k is arbitrarily. The expectation in (4) is evaluated using the statistics of the source being compressed.

In section IV we describe and analyse an encoder-decoder pair and we are able to prove that for each binary stationary source

$$RL \leq H(U_0,U_1,\ldots,U_{L-1}) + \text{ceil}(\log(L+1)). \tag{5}$$

All logarithms in this manuscript unless stated otherwise are assumed to have base 2. For the size of the buffers we find that it suffices to take

$$M = 2^L - 1. \tag{6}$$

Note that essentially our coding strategy is of the fixed-to-variable type.

It is well known (see Gallager [1], par. 3.3) that in our situation when the source is stationary

$$RL \geq H(U_0,U_1,\ldots,U_{L-1}|U_{-M},U_{1-M},\ldots,U_{-1}). \tag{7}$$

Also it is clear that (see again Gallager [1], par. 3.5), because of (5) and (7),

$$\lim_{L\to\infty} R = H_\infty(U), \tag{8}$$

where $H_\infty(U)$ is the entropy of the (stationary) source. It is (8) that makes our universal method optimal.

A crucial point in our argumentation is a result on repetition times. The next section is devoted to this subject.

III. REPETITION TIMES

A source generates $\ldots,x_{-2},x_{-1},x_0,x^1,x^2,\ldots$ with $x_t \varepsilon A_x$, a finite alphabet. We assume that this source is stationary.

Let A_x^+ be the subset of A_x that contains all x with $P(X_0=x)> 0$. Now for $m = 1,2,3,\ldots$ and $x \varepsilon A_x^+$ we define

$$Q_m(x) \triangleq P(X_{-m}=x, X_{1-m} \neq x, X_{2-m} \neq x, \ldots, X_{-1} \neq x | X_0=x). \tag{9}$$

and

$$T(x) = \sum_{m=1,\infty} mQ_m(x), \tag{10}$$

where it is understood that

$$\sum_{n=1,\infty} a_n \triangleq \lim_{N\to\infty} \sum_{n=1,N} a_n. \tag{11}$$

In this section we state the following theorem.

THEOREM: For a discrete stationary source, for $x \varepsilon A_x^+$

$$\sum_{m=1,\infty} Q_m(x) = 1, \text{ and} \tag{a}$$

$$P(X_0=x)T(x) = 1 - \lim_{N\to\infty} P(X_0 \neq x, X_1 \neq x, \ldots, X_N \neq x). \tag{b}$$

PROOF: The proof of this theorem is not given here.

IV. THE ALGORITHM

We start the description of our algorithm by introducing the following concept.

The L-th ordered derived source of the source $\{u_t\}_{t=-\infty}^{\infty}$ is defined as the source that generates $\{v_t\}_{t=-\infty}^{\infty}$ with $v_t \triangleq (u_{t-L}, u_{t-L+1}, \ldots, u_{t-1})$. Without proof we give the following lemma.

LEMMA: The L-th order derived source of a stationary source is stationary. (end)

It is important to note that this lemma implies that the theorem in section III holds for the L-th order derived source. We will now describe the encoding process of our universal algorithm.

Let t=kL, hence $v_t = u_k^L = (u_{t-L}, u_{t-L+1}, \ldots, u_{t-1})$ is being encoded. The buffer now contains $\phi_k = (u_{t-L-M}, u_{t-L-M+1}, \ldots, u_{t-L-1})$ with $M = 2^L-1$.

Note that using ϕ_k and u_k^L the encoder can form (has access to) v_{t-m} with $1 \leq m \leq M$. With these L-vectors the encoder determines the integer m_k. This m_k is set equal to the smallest m, $1 \leq m \leq 2^L-1$, for which

$$v_{t-m} = v_t. \tag{12}$$

If such an m_k can not be found set $m_k = M+1 = 2^L$. From the above it follows that $m_k \in S \triangleq \{1,2,\ldots,2^L\}$.

We now assume that S is partitioned in L+1 subsets. These subsets S_p, p=0,1,2,...,L, are defined as follows

$$S_p \triangleq \{2^p, 2^p+1, \ldots, 2^{p+1}-1\}, \text{ for } p=0,1,2,\ldots,L-1 \text{ and}$$
$$S_L \triangleq \{2^L\}. \tag{13}$$

Note that S_p for p=0,1,2,...,L-1 contains 2^p elements.
Next suppose that $m_k \neq 2^L$. Then, using the subsets of S, it is possible to assign to each m_k a subset number p which indicates that $m_k \in S_p$, and a member index q which is defined as

$$q \triangleq m - 2^p. \tag{14}$$

After having determined m_k, the encoder constructs a codeword $c_k^*(m_k)$.

If $m_k \neq 2^L$ the codeword c_k^* is obtained by concatenating the subset number p and the member index q of m_k, both in radix-2 notation. For the subset number ceil(log(L+1)) binary digits are needed, for the member index p binary digits. Hence if $m_k \neq 2^L$ (this means that u_k^L appears somewhere in the buffer),

$$lg(c_k^*) = bot(log(m_k)) + ceil(log(L+1)). \tag{15}$$

If $m_k = 2^L$ (this corresponds to the situation where no match for u_k^L is found in the buffer), the codeword c_k^* is obtained by concatenating the subset number L and the source word u_k^L, the subset number in radix-2 notation. Now for the subset number again ceil(log(L+1)) binary digits are needed and for the source word L digits. Hence for $m_k = 2^L$,

$$lg(c_k^*) = L + ceil(log(L+1)). \tag{16}$$

One easily verifies that the decoder after having received c_k^* can reconstruct u_k^L. Also note that the codewords emitted by the encoder satisfy the prefix condition.

We will now analyse the described algorithm. Suppose that $v_t(=u_k^L)=v$ is the codeword being encoded at t=kL. Now what is the average length L(v) of the codeword assigned to it? We extrapolate the notation of (9) somewhat and obtain

$$\begin{aligned} L(v) &= \sum_{m=1,2^L-1} Q_m(v)[bot(log(m)) + ceil(log(L+1))] \\ &\quad + \sum_{m=2^L,\infty} Q_m(v)[L + ceil(log(L+1))] \\ &\leq \sum_{m=1,\infty} Q_m(v)[log(m) + ceil(log(L+1))] \end{aligned}$$

(a)

$$= \sum_{m=1,\infty} Q_m(v)\log(m) + \text{ceil}(\log(L+1))$$

(b)

$$\leq \log\Big[\sum_{m=1,\infty} mQ_m(v)\Big] + \text{ceil}(\log(L+1))$$

$$= \log(T(v)) + \text{ceil}(\log(L+1))$$

(c)

$$\leq -\log(P(v_t=v)) + \text{ceil}(\log(L+1)). \qquad (17)$$

Here (b) follows from the (a)-part of the theorem in section III, (b) from the convexity of the log function and (c) from the (b)-part of this theorem. Note that throughout the derivation (17) we have used the fact that $P(v_t) > 0$. Fortunately only those v appear in the source output stream as v_t $(=u_k^L)$.

Using (17) we can now upperbound the efficiency of our system:

$$RL = \sum_{v:P(v_t=v)>0} P(v_t=v)L(v)$$

$$\leq \sum_{v:P(v_t=v)>0} P(v_t=v)[-\log(P(v_t=v) + \text{ceil}(\log(L+1))]$$

$$= H(V) + \text{ceil}(\log(L+1))$$

$$= H(U_0,U_1,\ldots,U_{L-1}) + \text{ceil}(\log(L+1)), \qquad (18)$$

where we have obeyed the convention that $0\log(0) = 0$. This concludes the proof of the result announced in section II.

V. CONCLUSION AND REMARKS

We conclude that our algorithm is easy to implement and that its minimax redundancy with respect to $H(U_0,U_1,\ldots,U_{L-1})$ instead of $LH_\infty(U)$ is acceptable for stationary sources.

The algorithm can be generalized to arbitrary source and code alphabet sizes.

The author was motivated by a number of very interesting papers in the field of universal source coding. These papers are well known and need not be referred to here.

REFERENCE

[1] R.G.GALLAGER, *Information Theory and Reliable Communication*, New York: Wiley, 1968.

CONSTRUCTING ARITHMETIC SOURCE CODES

Tjalling J. Tjalkens*

In this paper we discuss the more practical aspects of arithmetic source codes. We describe two types of codes and for both we give algorithms for constructing the code parameters and relate the code inefficiency to the complexity (space).
Hereafter we discuss the decoding of these codes, showing another tradeoff between redundancy and complexity (time).

INTRODUCTION

Last year [1] we described a class of arithmetic codes based on the Elias source coding algorithm [2]. We will briefly summarize the results.

Let X be a finite alphabet source with <u>alphabet</u> $AX \triangleq \{0,1,..,c-1\}$ and probabilities $P(\underline{x}^n)$, $\underline{x}^n \triangleq x_1,x_2,..x_n$. The finite <u>code alphabet</u> AY consist of the integers 0,1,..,d-1. We will assume d=2. An arithmetic code converts a source string $\underline{x}^n$ into a number $B(\underline{x}^n)\varepsilon[0,1)$. The <u>code string</u> $\underline{y}^m$ is the binary representation of $B(\underline{x}^n)$, or $B(\underline{x}^n) = \Sigma_{i=i}^{n}\, y_i 2^{-i}$.

The <u>rate</u> R(n) of the code is defined as:

$$R(n) \triangleq \frac{E\{L(x^n)\}}{n} \qquad (1)$$

where $L(\underline{x}^n) = m$ is the <u>length</u> of the representation $\underline{y}^m$ of $B(\underline{x}^n)$.

An arithmetic code uses an <u>exponential table</u> A[i]. This table is defined by two positive integers k and N and is given as: (i and j are integers)

$$A[iN+j] = \begin{cases} \lceil 2^{-j/N} \rceil_k & i=0,\ 0\le j<N \\ 2^{-i}A[j] & i\neq 0,\ 0\le j<N \end{cases} \qquad (2)$$

* Tj. Tjalkens is with the Eindhoven University of Technology, Department of Electrical Engineering, P.O.Box 513, 5600 MB Eindhoven, The Netherlands.

By $\lceil a \rceil_k$ we denote the smallest real number not less than a and expressable in at most k significant digits. Observe that the multiplication by 2^{-i} is a simple shift over i places.
The storage needed for this table is N(k-1) bits. As said in [1] the design of the code consists of selecting the integer stepsizes $s(x_n|\underline{x}^{n-1})$. A good design results in stepsizes such that $A[s(x_n|\underline{x}^{n-1})] \approx P(x_n|\underline{x}^{n-1})$. The recursive computation of $B(\underline{x}^n)$ is done by:

$$B(\underline{x}^0) \triangleq 0 \tag{3a}$$

$$S(\underline{x}^0) \triangleq 0 \tag{3b}$$

$$B(\underline{x}^{n+1}) = B(\underline{x}^n) + \sum_{x<x_{n+1}} A[S(\underline{x}^n)+s(x|\underline{x}^n)] \tag{3c}$$

$$S(\underline{x}^{n+1}) = S(\underline{x}^n) + s(x_{n+1}|\underline{x}^n) \tag{3d}$$

Here $\underline{x}^0$ is the empty string. For the meaning of S(.) see below.

DECODABILITY

The code is decodable if the stepsizes are such that for all $\underline{x}^* \varepsilon AX^*$ (the set of all finite strings over AX) [1]:

$$\sum_{x\varepsilon AX} A[S(\underline{x}^*)+s(x|\underline{x}^*)] \leq A[S(\underline{x}^*)] \tag{4}$$

Remark: Now we see the function of $S(\underline{x}^n)$. $A[S(\underline{x}^n)]$ gives the maximum augend to $B(\underline{x}^n)$ due to the extra source symbols $x_{n+1}x_{n+2}\cdots$. For a given $x^*\varepsilon AX^*$ (4) is called the local test at $S(\underline{x}^*)$.

We may remove the dependency on $S(\underline{x}^*)$ and obtain a weaker, global test which is a sufficient condition for decodability. The global test is:

$$\forall \underline{x}^* \varepsilon AX^* : \sum_{x\varepsilon AX} \lambda^{-s(x|\underline{x}^*)} \leq 1/\beta \tag{5}$$

with

$$\lambda = d^{1/N};\ \beta = 1+d^{1-k} \tag{6}$$

CODE DESIGN AND PERFORMANCE

First we give a global design (i.e. a fixed selection of the stepsizes independent of $S(\underline{x}^*)$).

$$\forall \underline{x}^* \epsilon AX^*,\ x \epsilon AX : s(x|\underline{x}^*) = \lceil \log_\lambda \beta - \log_\lambda P(x|\underline{x}^*) \rceil \tag{7}$$

This selection satisfies (5) so the code is decodable.

It can be shown that $L(\underline{x}^n) \leq \frac{S(x^n)}{N} + K$, for some small constant K. So

$$E\{L(\underline{X}^n)\} \leq H(\underline{X}^n) + \sum_{\underline{x}^n \epsilon AX^n} P(\underline{x}^n) \log_d \frac{P(x^n)}{\prod_{i=1}^n \lambda^{-s(x_i|\underline{x}^{i-1})}} + K \tag{8}$$

and from (7):

$$\lambda\beta\lambda^{-s(x_i|\underline{x}^{i-1})} \geq P(x_i|\underline{x}^{i-1}) \tag{9}$$

we obtain with (8):

$$R_1(n) \leq H_n(X) + \log_d \lambda\beta + o(1/n) \tag{10}$$

With $\log_d \lambda = 1/N$ and $\log_d \beta \leq \frac{d^{1-k}}{\ln d}$ we immediately see the dependency of the redundancy on the table size $N(k-1)$.

The rate R_1 can be improved upon by a local design. In this method we first select the stepsizes according to (7) and then when encoding x_{i+1}, we are at position $S(\underline{x}^i)$ in the table, we decrease the stepsizes as much as possible under the restriction of the local test (4) at $S(\underline{x}^i)$. Now we can show:

$$\sum_{x \epsilon AX} A[S(\underline{x}^i) + s(x|\underline{x}^i)] > (1 - \frac{\lambda\beta - 1}{c}) A[S(\underline{x}^i)] \tag{11}$$

Together with (9) we obtain

$$R_2(n) \leq H_n(X) + \frac{1}{\ln 2} \lambda(\beta - 1 + \frac{\lambda\beta - 1}{c}) + o(1/n) \tag{12}$$

A MODIFIED ALGORITHM

For a high cardinality source alphabet the encoding by (3) and the corresponding decoding takes many additions and comparisons. In this case the following adaption might be useful.
Let $Q(x|\underline{x}^*)$ be the conditional cumulative symbol probability:

$$Q(x|\underline{x}^*) = \sum_{y<x} P(y|\underline{x}^*)$$

Now we approximate $Q(x|\underline{x}^*)$ by $A[T(x|\underline{x}^*)]$ where $T(x|\underline{x}^*)$ is an integer
The encoding formulas, cf. (3), become:

$$B(\underline{x}^n) = B(\underline{x}^{n-1}) + A[S(\underline{x}^{n-1}) + T(x_n|\underline{x}^{n-1})] \tag{13a}$$

$$S(\underline{x}^n) = S(\underline{x}^{n-1}) + s(x_n|\underline{x}^{n-1}) \tag{13b}$$

To ensure decodability we need the following local test:

$$A[S(\underline{x}^*) + T(x+1|\underline{x}^*)] \geq A[S(\underline{x}^*) + T(x|\underline{x}^*)] + A[S(\underline{x}^*) + s(x|\underline{x}^*)] \tag{14}$$

This test must of course hold for all $x \in AX$ and $x^* \in AX^*$. From this we may obtain the following global test:

$$\forall x^* \in AX^* : \sum_{x \in AX} (\lambda\beta)^{M-x} \lambda^{-s(x|\underline{x}^*)} \leq 1 \tag{15}$$

If this is satisfied then the local condition is also satisfied and so the $T(x|\underline{x}^*)$'s can be found.
We now give a global and a local procedure for designing these codes. First the global procedure. Choose

$$\forall \underline{x}^* \in AX^*,\ x \in AX : s(x|\underline{x}^*) = \lceil c \log_\lambda \lambda\beta - \log_\lambda P(x|\underline{x}^*) \rceil \tag{16a}$$

Since this satisfies (15) we can find the corresponding $T(x|\underline{x}^*)$ by the following formula:

$$T(x|x^*) = \lceil (c-x)\log_\lambda \beta\lambda - \log_\lambda Q(x|\underline{x}^*) \rceil \tag{16b}$$

With (8) and

$$\lambda(\lambda\beta)^{c_\lambda - s(x|\underline{x}^*)} \geq P(x|\underline{x}^*) \tag{17}$$

we obtain

$$R_3(n) \leq H_n(X) + \log_d \lambda + c\log_d \lambda\beta + o(1/n) \tag{18}$$

A <u>local design</u> emerges if after finding $s(x|\underline{x}^*)$ and $T(x|\underline{x}^*)$ by the global method, we minimize each stepsize locally, i.e. for a given $S(\underline{x}^*)$, so that (14) still holds. Now it follows that

$$\sum_{x\in AX} A[S(\underline{x}^*) + s(x|\underline{x}^*)] > (1-\beta^2(\lambda\beta-\beta^{-1}))A[S(\underline{x}^*)] \tag{19}$$

resulting together with (17) and (8) in

$$R_4(n) \leq H_n(X) + \frac{\lambda(\lambda\beta)^c}{\ln d} (\lambda\beta^2-1) + o(1/n) \tag{20}$$

CARRY BLOCKING

Observe that the addition of A[.] to B(.) is done with k bits precision. The with n increasing offset $S(\underline{x}^n)$ shifts the augend $A[S(\underline{x}^n)+s(x|\underline{x}^n)]$ further to the right. This is equivalent and technically more easy to implement by shifting $B(\underline{x}^n)$ to the left by the same amount. We would like to be able to transmit the symbols from $B(\underline{x}^n)$ already shifted out of the last k positions, but this implies that these digits may not change. However, it is not hard to see that <u>carrys</u> can occur during the additions, at most one per source symbol, and these might change an arbitrary long string of code symbols. To prevent this Langdon and Rissanen [3] describe a <u>blocking technique</u>. However their technique as described is incorrect. While it can be repaired we propose a somewhat different method, that is more in line

with the whole algorithm. Just like Langdon and Rissanen we save the last q symbols shifted out of the adder in a special register that can perform shift and increment operations. If this register does not contain only ones then an occuring carry will be stopped within the register. Otherwise the carry will propagate out of it and this we want to prevent.

We suggest that whenever the register contains only ones after a source symbol is encoded the B(.) string is shifted to the left until the blocking register contains a zero. Assuming the symbol probability in the tail of B(.) to be $(\frac{1}{2},\frac{1}{2})$, since the binary string is the output of a good source encoder, we need two shifts on the average whenever the "all ones" condition occurs. Experiments indicate that this is a slightly conservative estimate. The probability of the "all ones" condition is about one per 2^q source symbols.

DECODING

Decoding is done by simulating the encoder, that is the decoder tries to build its own number $B(\underline{\hat{x}}^n)$ in correspondence with the received codestring $\underline{y}^m$. For this purpose it uses the same equations, tables and carryblocking mechanism as the encoder.

We will give a recursive description of the decoding. We write $V(\underline{y}^\ell)$ for the value $\Sigma_{j=1}^{\ell} y_j 2^{-j}$ represented by $\underline{y}^\ell$. Let $\underline{y}^\ell$ be the received prefix of $\underline{y}^m$, the final codestring. Let $\underline{\hat{x}}^i$ be decoded correctly $(\underline{\hat{x}}^i=\underline{x}^i)$ from $\underline{y}^\ell$. So

$$B(\underline{\hat{x}}^i) \leq V(\underline{y}^\ell) \text{ and} \tag{21}$$

$$B(\underline{\hat{x}}^i) + A[S(\underline{\hat{x}}^i)] > V(\underline{y}^\ell) + 2^{-\ell} \tag{22}$$

Let $\underline{y}^{\ell_1}(\ell_1 \geq \ell)$ be the shortest extension of $\underline{y}^\ell$ such that there exists an $x \in AX$ with

$$B(\underline{\hat{x}}^i x) \leq V(\underline{y}^{\ell_1}) \tag{23}$$

$$B(\underline{\hat{x}}^i x) + A[S(\underline{\hat{x}}^i x)] > V(\underline{y}^{\ell_1}) + 2^{-\ell_1} \tag{24}$$

It is not hard to see that due to (4) or (14) this x is unique and equal to x_{i+1}. So we set $\hat{x}_{i+1}=x$, and the string $\underline{y}^m$ will be decoded correctly into $\underline{x}^n$. After decoding a symbol we perform the carry blocking shifts if needed. In this way decoding is a search operation through all possible c augends. For the algorithm (3) this can only be implemented as a linear search through AX, resulting in c tests at most and $\bar{x}=E\{x\}$ tests on the average.

The modified algorithm might be decoded using binary search resulting in about log c tests per decoded symbol. If the source symbols are ordered such that $P(o)\geq\ldots\geq P(c-1)$, or vice versa, then in stead of the binary search tree we might use a Huffman search tree minimizing the average number of tests to at most H(X)+1 tests. A trivial upperbound to the longest path (and search) in this tree is c. Huffman decoding will be awkward to implement for sources with memory but is quite useful for memoryless sources.

COMPLEXITY

First consider the storage requirements. As said before the table A[.] needs N(k-1) bits of storage. If the stepsizes are precomputed by the global method then we need one, or more in the Markov case, tables to store these. Assuming either that these tables are small compared to A[.] or that we cannot or will not precompute the stepsizes we might solve for the minimum rate $R_1..R_4$ with a constraint on N(k-1).

Now we turn to the amount of work needed to en- and decode a source symbol. The code (3) has an en- and decoding time proportional to $\bar{x}$(or c), while the code (13) has an allmost constant encoding time and a decoding time proportional to log c or H(X). Now compare the local methods with the corresponding global ones. Notice that the local methods start with the global parameters and then optimize for the current symbol. This implies searching in A[.] which can be done in O(logN) tries per point searched. The global stepsizes must be computed or stored in a table. With code (3) all global stepsizes are needed for the local method, so having to compute them every time is expensive. The optimization itself is also comparatively complex

since it involves $\bar{x}$ searches. The local method in code (13) is not much more complex than the global method. We need to compute two T(.)'s in stead of one and we need one search.

CONCLUSIONS

We described two codes and two design methods for each. If we compare the bounds on the rate (and the coding times), then we conclude that

- the local methods are substantially better than the global ones with respect to the redundancy.
- code (3) has a lower redundancy than code (13) for the same table A[.]. For the global methods they differ a factor c, but the local methods can be optimized so that they differ only by a factor of about 2.
- for high cardinality alphabets the local method for code (3) becomes very time consuming and even the global code (3) is more expensive than code (13).
- for code (13) the global and local methods do not differ much in time, while the redundancy bound in the local method is much lower.
- the availability of precomputed stepsizes reduces the en- and decoding times considerably.

REFERENCES

[1] Tj.J.Tjalkens & F.M.J.Willems, Arithmetic Coding, Proc. 6th Benelux Symp.Inform.Theory, Mierlo, The Netherlands, pp.141-150, 1985.

[2] F.Jelinek, *Probabilistic Information Theory*, New York: Mc Graw-Hill, 1968.

[3] G.G.Langdon, Jr. & J.Rissanen, Compression of Black-White Images with Arithmetic Coding, IEEE Trans.Comm. vol.COM-29, no.6, june 1981.

THREE-DIMENSIONAL OBJECT RECOGNITION BY USING STEREO VISION

E.F.P. van Mieghem, J.J. Gerbrands, E. Backer

The recognition of three-dimensional solid objects is a well-known problem from the field of robot vision for industrial applications. In the stereo vision approach two two-dimensional images are obtained from calibrated camera positions. In the method discussed here, a graph is constructed for each of the two images with the nodes corresponding to the object vertices. Both graphs are matched with a branch-and-bound algorithm. Three-dimensional object features are computed and used as attributes in the inexact graph matching recognition stage. The discussion is restricted to trihedral objects.

STEREO VISION

It is well understood that machine vision will play an important role in flexible automation and computer-aided manufacturing. Most robot vision systems functioning to date are essentially two-dimensional (2-D) in nature. In the emerging field of robot vision and sensory control, much research is devoted to the problem of actually obtaining three-dimensional (3-D) information about the robot's environment. This includes the recognition of 3-D objects as well as the determination of object position and orientation in 3-D world coordinates. There are a number of ways in which this problem can be attacked. One distinguishes active and passive imaging techniques. In the active techniques some sort of active source (ultrasound, laser) is used, while the passive techniques employ overall scene illumination. A second dichotomy is to distinguish methods which use triangulation and methods which use the perspective transform.

The stereo vision approach to be discussed here is a passive triangulation method. In stereo vision two 2-D images of the 3-D scene are acquired from two distinct camera positions, as shown in Fig. 1.

Delft University of Technology, Department of Electrical Engineering, P.O Box 5031, NL-2600 GA DELFT

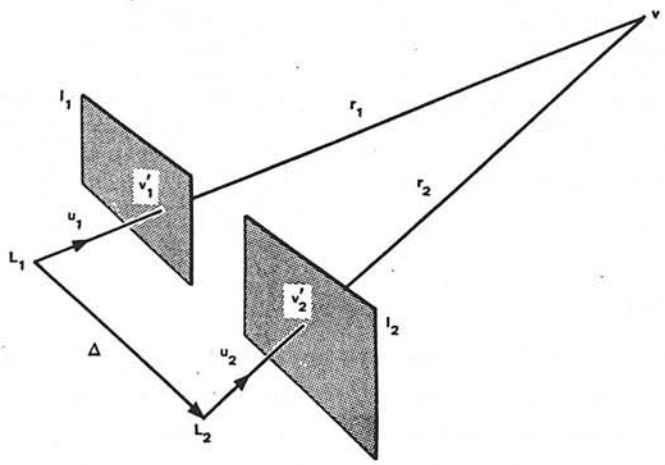


Fig. 1. The principle of triangulation.

The point V in 3-D space is projected on the 2-D coordinates v'_1 and v'_2 in the 2-D images I_1 and I_2, respectively. The 3-D coordinates of the point V can now be computed from the 2-D coordinates v'_1 and v'_2 if the positions and orientations of the cameras are known [1].

THE CORRESPONDENCE PROBLEM

In order to apply the method of triangulation, one has to find pairs of corresponding points v'_1 and v'_2 in the two images. This correspondence problem is greatly facilitated if we restrict the complexity of the scene. In a first attempt we consider isolated trihedral objects. A trihedral object is an object from the blocks world with not more than three edges at every vertex [2]. Some examples are given in Fig. 2. As trihedral objects are completely described in terms of vertices and edges, it is most natural to consider the vertices, as observed in the 2-D images, as characteristic points in the correspondence problem.

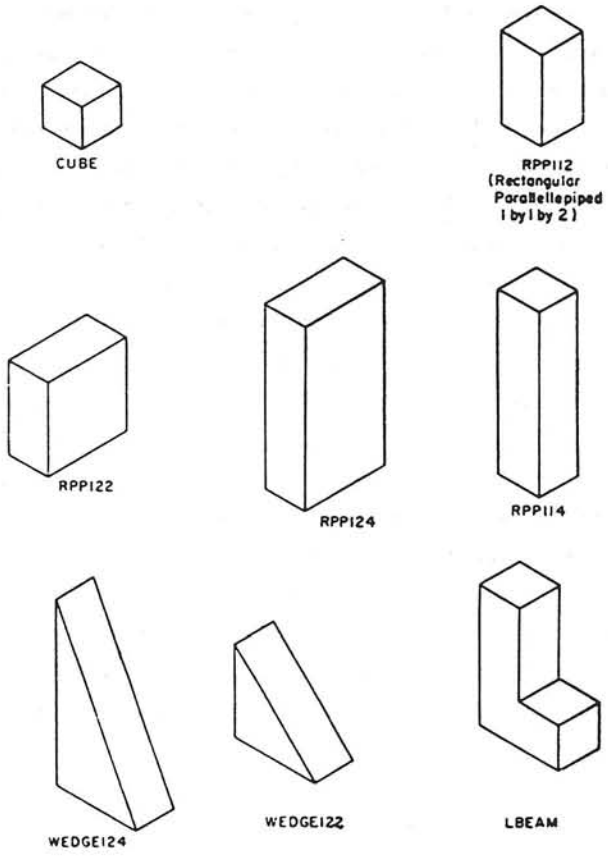


Fig. 2. Trihedral objects

In order to detect the vertices in the 2-D grayvalue images we construct a line drawing of the object [3,4]. First the grayvalue image is convolved with linear discrete difference operators to obtain the components of the Sobel-gradient. Second, a spatial clustering scheme is applied to find clusters of pixels with high gradient values and similar gradient directions. Finally, the projected object edges are found by fitting a straight line through the pixels of each cluster.

The projected vertices are detected at the intersection points of the fitted lines.

Obviously, the same procedure is applied to the second image. Now, we have to find pairs of corresponding vertices in the two images of the stereo pair. From Fig. 1 it is obvious that a point in one of the 2-D images, say point v_1', may be the projection of any point on the projecting ray r_1. The projection of r_1 onto the second image I_2 is called the matchline of v_1', and all points on the matchline are candidates to be the corresponding point of v_1'. So, if we consider a projected object vertex in I_1, we search on or close to its matchline in I_2 for its corresponding projection. Frequently, this is done by computing the cross correlation between greyvalue subimages. This is extremely time consuming. Instead, we use a minimum cost graph matching technique.

In the graph matching approach the line drawings of the projected objects are used as graphs, the nodes of the graphs being the projected object vertices. Consider two nodes: node N_1 of graph G_1 representing vertex N_1 in image I_1, and node N_2 of graph G_2 representing vertex N_2 in image I_2. Let L_1 denote the matchline of N_1 in I_2 and L_2 the matchline of N_2 in I_1. The euclidean distance between a vertex N and a matchline L is denoted as d(N,L). Now we define the costs of matching N_1 and N_2 as

$$C(N_1,N_2) = d(N_1,L_2) + d(N_2,L_1)$$

and compute these cost coefficients for all pairs of nodes of the graphs G_1 and G_2. We then apply the well-known branch-and-bound algorithm to find the minimum cost match between G_1 and G_2, where the costs of matching two graphs is defined as the sum of the cost coefficients of all pairs of nodes in the match. The pairs of nodes in the optimal match define the corresponding projected object vertices.

OBJECT RECOGNITION

Having solved the correspondence problem, the 3-D coordinates of the object vertices can be computed. It is then possible to compute

the lengths of the object edges in 3-D space as well as the angles between edges at the vertices. These values are used as attributes in a 3-D graph representation of the object. Each object class is represented by a prototype graph. Again the inexact graph matching technique is applied to find the optimal match between the vertices in the observed object and those in the prototype. In principle, this is repeated for all prototypes and the observed object is assigned the label of the model with the minimum matching costs. The matching costs are defined as the absolute difference of edge lengths and the absolute difference of angles between the observed object and the model.

The speed of the recognition stage is greatly improved by performing a preselection with respect to the prototypes to be considered in detail. This preselection implies that for each node pair the cost of the best match of edges is computed. This is repeated for all node pairs independently and summed. If these costs exceed a certain threshold, the prototype model is discarded.

CONCLUDING REMARKS

The methods described above have been implemented on a low cost vision system (Motorola 68000). Preliminary experiments using the objects given in Fig. 2 indicate that all objects can be recognized correctly if they are presented one by one to the system. Further experiments are needed to investigate the problems of partly occluding objects, i.e., scenes with a higher complexity [5].

REFERENCES

[1] Y. YAKIMOVSKY and R. CUNNINGHAM, A system for extracting three-dimensional measurements from a stereo pair of TV cameras, Computer Graphics and Image Processing 7 (1978) 195-210.

[2] A. BARR and E.A. FEIGENBAUM, *The handbook of artificial intelligence*, Ditman, London, 1982.

[3] M. AKKERMANS and M. VAN THILLO, An algorithm for the extraction of line drawings for polyhedral scenes and their use in stereo vision, Proceedings SPIE 449 (1984) 534-540.

[4] P. HOFLAND, *De bepaling van een lijntekening van een scene t.b.v. robot vision*, M.Sc. Thesis Delft University of Technology, 1985.

[5] E.F.P. VAN MIEGHEM, *De herkenning van 3-D voorwerpen met behulp van stereo-vision*, M.Sc. Thesis Delft University of Technology, 1985.

MULTIRESOLUTIONAL CLUSTER/RELAXATION IN SEGMENTATION

J.J.Gerbrands*, E.Backer*, X.S.Cheng*

A multiresolutional segmentation algorithm is described. A quadtree based split-merge procedure generates variable-sized quadtree-blocks (multiresolutional data units) being the data units used in a cluster procedure to extract regional features. A nonlinear probabilistic relaxation procedure, then, conducts the final quadtree-block labeling. It is shown that a large reduction in data processing is attained by processing blocks rather than pixels and still the result reasonably approximates the true segmentation. Also, some experimental results are included here.

1. Introduction

The use of clustering and relaxation in image segmentation has appeared in literature over the past decade [1,2]. However, the majority of those approaches are pixel-based and therefore bear inevitable drawbacks and limitations. Firstly, only a limited number of pixels may attend the clustering and relaxation processes to keep computational complexity and memory requirement within limits. Secondly, the representativity of individual pixels may be regarded as quite poor because of inevitable noise influence. As a result, those approaches are vulnerable to yield inconsistent segmentations even if a suitable context-based relaxation process was involved.

Work reported here is an attempt to break the above limitations. The concept which will be implemented is as follows:

1.Generate a number of image primitives (sets of connected pixels)

---

*The authors are with the Delft University of Tecnology, Department of Electrical Engineering, P.O.Box 5031, 2600 GA Delft, The Netherlands

so that the local consistency within each primitive is satisfactory. Note that primitives are not necessarily equally sized.

2. Select 'dominant' primitives and apply a clustering process on them for extracting class information (i.e. characteristics of existing regions).
3. Based on the clustering result, assign initial class memberships to all primitives. For those primitives which have little dominance, the initial distribution may be uniform.
4. Conduct a relaxation process on the primitives using locally dependent compatibility coefficients. Note that primitives with a relatively large dominancy may be excluded from this process.

Clearly, the above concept offers two obvious merits:

a. Replacing single pixels by larger primitives reduces the number of operational data units drastically,
b. By allowing only most-dominant primitives to attend the clustering process, the clusters will be much more reliable while at the same time the clustering process will involve less data units.

Certainly the fundamental assumption is that step 1 can be realized satisfactorily with feasible complexity and implementation. Here, a quadtree based split-merge procedure meets our requirements to generate the desired primitives. The resulting quadtree blocks (QT-blocks) are considered as the (multiresolutional, variable-sized) primitives. Typically, in a quadtree-based procedure a QT-block's dominancy can be related to its size.

In the following section, we will discuss the iterative split-merge scheme, the clustering scheme and the relaxation scheme in greater detail.

2. Design of the Multiresolutional Segmentation Approach

A. The iterative split-merge scheme.

Contrary to the general split-and-merge approach introduced by Horowitz and Pavlidis [3] the procedure here is only to yield a suitable set of QT-blocks with one understanding that an overmerged output quadtree will do more harm than an oversplitted one. In our

example we have adopted the variance criterion for this presegmentation.

Below are two 'goodness'-measures of a resulting quadtree, which we have introduced to provide some sort of a feedback facility letting the process itself iteratively improve its output:

a. AP: Area Preserve

this measure is defined as the total area of blocks which are larger than a size threshold.

b. RP: Region Preserve

this measure is defined as the ratio of the sample variance among the weighted means of all individual blocks larger than a size threshold to the sample variance in the input image.

The size threshold above is a priori chosen based on the smallest expected region(s) and the smallest size of a block whose sample variance may still give a confident estimate for the enclosing region. If AP is too small, it will either indicate the unsuitability of the criterion or reflect the impropriety of the chosen threshold toward the input. On the other hand, an excessively large value of AP will in most practical circumstances suggest that the chosen threshold is too large as in real-world imagery a region will generally bring about many small QT-blocks along its border. In conclusion, if we do acknowledge the suitability of the criterion, a proper value of AP should be within some limited range [AL,AH].

The behaviour of RP is characteristic for preserving the original region structure within the output quadtree. More precisely, each existing region must contain at least one large block and any large block must not cover more than one region. The higher RP the more representative the set of large QT-blocks is for this structure. To ensure such a representativity RP should exceed some lowerbound RL.

B. The clustering process.

As pointed out before, the clustering process is carried out on QT-blocks exceeding some size-threshold. The blocks participating in the clustering process are marked 'active'. Normally for processing images of size 256x256, we may fix this threshold at, say 4x4. How-

ever, to handle possible extreme situations it may be necessary to adjust this threshold. The following criteria may detect such cases and initiate appropriate emergency measures:

a. The ratio of the sample variance among active blocks to the sample variance among all blocks should not be too small,

b. the total area of active blocks must exceed some threshold, and

c. the number of active blocks must not be too large.

Relying on the representativity of relatively large QT-blocks, we may expect no outliers in the clustering process. Together with a fact that only a few handreds of blocks are generally chosen active we can therefore apply more sophisticated clustering procedures for this purpose.

In our example we have adopted a clustering procedure somehow similar to MacQueen's k-means method for variable number of clusters [4]. The central issue here is how to properly assign 'coarsening' and 'refining' parameters. We have settled this by utilizing some a posteriori knowledge, e.g. the within-variances of active QT-blocks.

C. The relaxation process.

Among many recent methods, the so-called nonlinear probabilistic relaxation approach appears to be particularly suitable for many purposes. Letting $P_i^{(k)}\{a\}$ be the probability about block i belonging to region {a} in kth iteration, the modificating operation in such an approach is directed as follows:

$$P_i^{(k+1)}\{a\}=P_i^{(k)}\{a\}[1+Q_i^{(k)}\{a\}]/A_i^{(k)}$$

where $A_i^{(k)}$ is a normalization factor to ensure the distributional nature of P and $Q_i^{(k)}\{a\}$ is some sort of support to labeling {a} at i from its spatial neighborhood. Clearly, Q is the only way for excerting contextual influences under this modificating rule. Typically, it takes the following form:

$$Q_i^{(k)}\{a\}=\sum_{j\in N_i} w_{ij} \sum_{b\in V} r_{ij}(a,b)P_j^{(k)}\{b\}$$

where N_i is the neighborhood of i, V is the label set, w_{ij} is the neighborhood weighing factor ($\sum_j w_{ij}=1$) and $r_{ij}(a,b)$ is the so-called compatibility coefficient within [-1,1], which expresses to what extent labeling {b} at j is compatible with labeling {a} for i.

The very meaning of $r_{ij}(a,b)$ can be seen as some context-dependent and intuitive (in the sense of a priori knowledge and intended goal) measure about to what extent labeling {a} at i is compatible with labeling {b} at a neighboring j when the labeling for i is facing reconsideration, or more naturally as some support to labeling {a} at i from labeling {b} at j. In Rosenfeld, et al. [5], the compatibility coefficients for a similar process were determined based on a finite number of physical evidences while in Zucker et al. [6], they were selected under a clear understanding that neighboring pixels should in most cases have very similar edge properties. However in our case, instead of a general object set we encounter a set of variable-sized QT-blocks. It is hardly possible or reasonable to comment on the (in)consistences among neighboring blocks without some knowledge of existing regions or regional properties of relevant blocks. Assuming that the regions to be searched are somehow convex or at least locally convex, $r_{ij}(a,b)$ can however be reasoned to behave in the following way.

If j is smaller than i, it is then very reasonable that less or even no action should be taken to adjust the labeling at i in order to improve the compatibility with that at j due to the following:

a. j may lie between two larger and unaligned blocks (i is one of them) from two adjacent regions and therefore the current labeling at j is not yet stable on its own,
b. j is certainly more likely to be a border element of a region than i. If they both belong to a common region, then there is obviously a necessity for j to be compatibly labeled with i and not the other way round. Otherwise, any current labeling at i is clearly not incompatible with that at j.

Out of these considerations, it is quite natural to choose a relatively small magnitude for $r_{ij}(a,b)$ based on some non-negative function $F(SIDE_j, SIDE_i)$ with F somehow proportional to $SIDE_j$ and

inversely proportional to $SIDE_i$. Under a similar idea, we tend to let $r_{ij}(a,b)$ also vary according to $F(SIDE_j, SIDE_i)$ when $SIDE_i \leqslant SIDE_j$.

So, an obvious choice will be:

$$r_{ij}(a,b)=c_i(2\delta_{ab}-1)F(SIDE_j, SIDE_i)$$

where δ_{ab} is the Kronecker delta, and c_i is a positive scaling factor which also ensures $r_{ij}(a,b)$ within $[-1,1]$. Clearly, c_i should never exceed $1/F(L,SIDE_i)$ with L being the largest existing block size. Thus,

$$r_{ij}(a,b)=c(2\delta_{ab}-1)F(SIDE_j, SIDE_i)/F(L,SIDE_i)$$

where $c\epsilon(0,1]$ is now independent of any specific i.

An obvious and simple choice for F is $F(x,y)=(x/y)^p$ with $p>0$ and as a result, we obtain:

$$|r_{ij}|=c2^{(k-s)p} \qquad \text{If } SIDE_j=2^k \text{ and } L=2^s$$

3. Experimental Results

In the present experiments two test images of size 256x256 were used (Fig. 1). Based on some a priori judgements we have fixed [AL,AH] at (50,85) and RL at 30 respectively. All blocks exceeding 4x4 are accepted for taking part in the calculation of AP and RP.

The smallest size of clustering-active QT-blocks is always initially set to 8x8. However, to ensure regional representativity of active blocks attempts were made to check such a representativity. The lowerbounds for the area and variance percentages were fixed at 60 and 45 respectively while the upperbound for a tolarable cluster data size was set to 600. If this checking operation fails to approve the selected blocks, subsequent steps will appropriately be taken to adjust the active-block threshold to either 16x16 or 4x4.

For the final relaxation purpose we have adopted the following

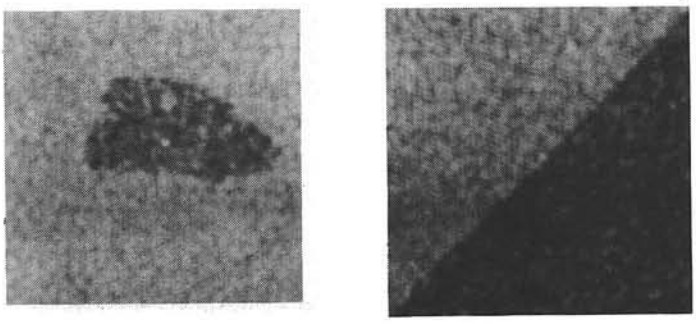


Figure 1. Input test images.

initialization mechanism:

$$P_i^{(0)}\{a\}=[1-d_i\{a\}/\sum_{b\in V} d_i\{b\}]/(N-1)$$

where $d_i\{a\}$ is the feature distance between block i and cluster {a} and N is the number of clusters. To all QT-blocks of a single pixel we have applied the uniform initialization. Furthermore, all blocks larger than 32x32 were excluded from the relaxation process. To reduce undesired artifacts we have set p=1 and c=1 for the compatibility coefficients. The final results are shown in Fig. 2.

Although we still observe some artifacts on the final outputs, the overall quality of the results does exhibit some significance of the proposed approach. Especially, the detected boundaries of the actual regions are quite satisfactory.

4. Conclusions

From the experimental results so far, we may come to the following conclusions:

a. The iterative split-merge scheme is a workable approach and may be fully automated under measures AP and RP proposed here.

b. By clustering and relaxing QT-blocks instead of single pixels the final result can approach the true segmentation quite reasonably.

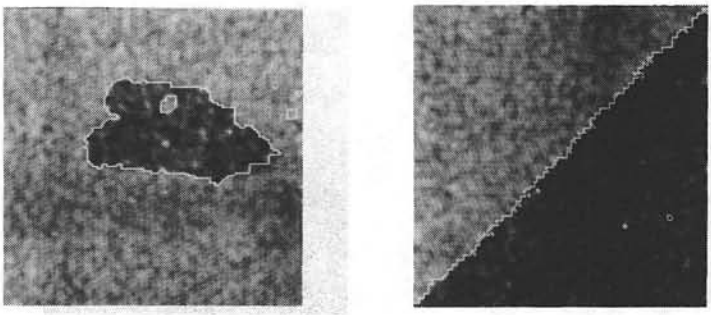


Figure 2. Contours detected on the original inputs.

From the above, we may expect the proposed approach to be further developed into a well-behaved method to tackle segmentation problems for a wide range of purposes and especially for images where regional properties play a dominant role.

5. References

[1] G.B. Coleman & H.C. Andrews, Image segmentation by clustering, Proc. IEEE, Vol. 67, No. 5, May 1979, 773-785.

[2] P.A. Nagin, Segmentation using spatial context and feature space cluster labels, Univ. of Massachusetts, COINS Techn. Report 78-8, May 1978.

[3] T. Pavlidis, Structural pattern recognition, Springer-Verlag, Berlin, 1977.

[4] M.R. Anderberg, Cluster analysis for applications, Academic Press, New York, 1973.

[5] A. Rosenfeld, R.A. Hummel & S.W. Zucker, Scene labeling by relaxation operations, IEEE Trans. on SMC, Vol. SMC-6, No. 6, June 1976, 420-433.

[6] S.W. Zucker, R.A. Hummel & A. Rosenfeld, An application of relaxation labeling to line and curve enhancement, IEEE Trans. on Comp., Vol. C-26, No. 4, April 1977, 394-403.

REGULARIZED ITERATIVE IMAGE RESTORATION

R.L. Lagendijk*, J. Biemond*

In this paper a regularized iterative algorithm is described which solves the ill-posed image restoration problem in a numerically stable way by incorporating a priori knowledge about the original image. Three kinds of a priori knowledge are used: the first type imposes an upperbound on the residual signal, and the second type restricts the high-frequency content of the (restored) signal. We show that by the use of weighted norms in defining the above-mentioned types of a priori knowledge the algorithm concentrates on restoration in the vicinity of edges, and on noise suppression in flat regions. In this way the algorithm is capable of handling spatially varying image statistics in a pleasing manner for the human observer. The third kind of a priori knowledge is a deterministic constraint representing a closed convex set in the solution space. In order to show the significance of our iterative algorithm we present some restoration results on a real photographically blurred image.

1. INTRODUCTION

In image restoration the ultimate goal is the recovery of the original scene from a distorted version. The distortion may be due to motion of the camera with respect to the original scene, defocusing of the lens system, etc. In addition, the distorted image is nearly always corrupted by random noise. We model our noisy blurred images as follows:

$$g = Df + n, \tag{1}$$

where the linear distortion operator D is known or can be satisfactory identified. The original and noisy blurred images are denoted by the (lexicographically ordered) vectors f and g, respectively. The signal-uncorrelated random noise is represented by an additive term n, of which the characteristics are only partially known in practice. Hence, the

* Delft University of Technology, Department of Electrical Engineering, Information Theory Group, P.O. Box 5031, 2600 GA Delft, the Netherlands.

exact original image cannot be computed from the distorted version. Image restoration concentrates on how to filter the distorted data to achieve an improved image, which is an acceptable approximation of the original image.

In recent years iterative signal restoration, and iterative image restoration in particular, has received considerable attention [1]-[5]. Among the advantages of iterative solution methods we mention:

- the possibility of including nonlinear constraints which reflect certain deterministic a priori information about the original image,
- the truncation of the iterative process after a finite number of iterations in order to obtain an optimal result for the human visual system,
- the possibility of avoiding the determination of the inverse distortion operator.

However, most of the existing iterative algorithms are derived without explicitly taking into account the presence of noise in the distorted images. As a result excessive noise amplification will occur when the number of iterations increases. It can be shown that this effect results from the ill-posedness of the restoration problem [4], [6], [7]. In order to solve the ill-posed image restoration problem, a priori information about the original image has to be included in the derivation of the restoration algorithm. Such an approach is known as "regularization" [6], [7].

In section 2 we describe three kinds of a priori knowledge which are used in regularizing the image restoration problem. Furthermore, the concept of weighted norms is introduced in order to incorporate fundamentally spatially varying image statistics. In section 3 we present the derivation of our regularized iterative algorithm. First we follow the Miller regularization approach [11], and next compute iteratively a solution of the obtained regularized equation, simultaneously applying a deterministic constraint in each iteration step. Some experimental results on a real photographically blurred image are given in section 4.

We remark that the iterative restoration algorithm presented in this paper is an extension of the algorithms proposed by Katsaggelos and

Biemond [2], [5].

2. A PRIORI KNOWLEDGE

In this section we introduce three types of a priori knowledge about the original image to be used in the derivation of our algorithm. In the first two we make use of weight matrices to enable the handling of spatially varying image statistics such as the local signal activity (edges, flat regions) in a pleasing manner for the human visual system. The third type of knowledge consists of a (possibly non linear) deterministic constraint, well-known from the theory of the convex projections [10], [12].

Since in image restoration the receiver of a restored image is usually the human observer, we should like to incorporate some characteristics of the visual system into our restoration methods. However, the structure and responses of this system are very complex, and cannot easily be represented by mathematical equations. Therefore, we merely make use of the following two global results from psychophysical experiments [8]:

- noise in flat regions of an image gives rise to extraordinary features to the observer, while the presence of sharp intensity transitions considerably reduces the visibility of noise (masking effect),
- sharp edges contribute strongly to the appraised quality of (restored) images.

From these experimental results we conclude that restoration must prevail over noise suppression in the regions where sharp intensity transitions are found, while on the other hand regions containing only slow intensity variations must be as smooth as possible. We use these experimental results in defining our a priori knowledge.

In the first place we demand the estimate of the original solution to be an element of the set of admissible solutions, defined by:

$$||g-Df||_R^2 = (g-Df)^t R(g-Df) \leq \varepsilon^2. \qquad (2)$$

Here R is a diagonal weight matrix, so the norm is taken in a weighted

Hilbert space. The global bound ε^2 on the weighted length of the residual signal g-Df is assumed a priori known, and is obviously related to the amount of noise present in the distorted image. If we assume this image noiseless (ε=0), equation (2) directly leads to the (pseudo-)inverse filter estimate [9].

The weight matrix R may incorporate certain aspects of the human visual system. For example, in the vicinity of edges the weight coefficients should be assigned large values in order to enforce inverse filtering due to the fixed upperbound ε^2. Consequently, the resolution gain will be large, but inherently related to this, considerable noise amplification may be expected as well. This is, however, not disturbing to the observer due to the masking effect.

The second kind of a priori knowledge about the original image is defined by:

$$||Lf||_S^2 = (Lf)^t \, S(Lf) \leq E^2. \qquad (3)$$

Here S is again a diagonal weight matrix and E^2 a known upperbound on the weighted norm. L is a physically realistic, invertible regularizing operator which reflects some desired properties of the restored image. In fact, we restrict with eq. (3) the set of admissible solutions (eq. (2)) to a smaller subset.

A common assumption made in image restoration is that the noise is broad-banded and that the distortion has a low-pass filtering effect. In consequence of this, particularly high-frequency noise will be magnified enormously. Therefore, the regularizing operator L is generally a low-pass filter, imposing a smoothness requirement on the restored image.

The weight matrix S locally regulates this requirement, depending on the characteristics of the visual system. The coefficients in the matrix S are choosen in such a way that in flat regions high-frequencies (which can merely be noise) are penalized strongly, and in the vicinity of edges high-frequencies are hardly penalized. By doing so, we obtain both sharp edges and smooth flat areas.

Lastly, the third type of a priori knowledge suitable for our iterative algorithm is a (possibly non linear) deterministic constraint C, representing a closed convex set in the solution space [10], [12]. Some well-known constraints in image processing are nonnegativity, maximal energy and (locally) bounding the image intensities. The orthogonal nonexpansive projection P onto the closed convex set C is defined by:

$$\begin{aligned} Pf &= f, && \text{if } f \text{ satisfies } C \\ &= h, && \text{otherwise, where } h \in C \text{ and} \\ & && ||h-f|| \leq ||x-f||, \ \forall x \in C. \end{aligned} \tag{4}$$

3. FORMULATION OF THE REGULARIZED ITERATIVE ALGORITHM

Miller Regularization

Following the Miller regularization approach [11], we combine both sets described by eq. (2) and (3) into a single quadrature formula:

$$\Phi(f) = ||g-Df||_R^2 + \alpha||Lf||_S^2, \tag{5}$$

where the regularization parameter α has the fixed value $\alpha=(\varepsilon/E)^2$. A solution satisfying both eq. (2) and (3) is obtained by minimizing the functional $\Phi(f)$, yielding

$$(D^tRD + \alpha L^tSL)\hat{f}_m = D^tRg. \tag{6}$$

Here $\hat{f}_m$ is the unique Miller solution to the restoration problem. Observe that if we assume the images of size NxM, the actual computation of $\hat{f}_m$ would require the inversion of the matrix $D^tRD + \alpha L^tSL$, which has the size N^2xM^2. Since this matrix represents a space-variant operator, we cannot reduce the computational complexity by applying the standard diagonalization procedure for block-circulant matrices (i.e. Fourier domain filtering) [9]. Furthermore, we cannot guarantee that the solution $\hat{f}_m$ will satisfy the constraint C as well, nor can we modify eq. (6) so that $\hat{f}_m$ always meets the desired deterministic condition. For these reasons the solution $\hat{f}_m$ is approximated by using an iterative

method, which simultaneously offers the possibility of imposing the constraint C on the solution.

Iterative solution method

We rewrite eq. (6) as

$$\hat{f}_m = (I-\alpha\beta L^t SL)\hat{f}_m + \beta D^t R(g-D\hat{f}_m)$$

$$= G(\hat{f}_m), \tag{7}$$

where β is called the relaxation parameter. The unique fixed point of this mapping coincides with the solution of eq.(6), and can be computed by using the contraction mapping theorem [12]:

$$\hat{f}_{k+1} = G(\hat{f}_k). \tag{8}$$

A sufficient condition for the convergence of these iterations is the contractiveness of the mapping G, which results in the following condition:

$$0 < \beta < \frac{2}{||D^t RD + \alpha L^t SL||} \tag{9}$$

We now introduce the constraint C in the iterative algorithm:

$$\hat{f}_{k+1} = P\, G(\hat{f}_k). \tag{10}$$

The iterations $\hat{f}_k$ converge to the unique fixed point $\hat{f}_\ell$ of the composed mapping P G in the convex set C, provided that β satisfies the bound in eq. (9). It can be shown that the iterative algorithm in eq. (10) minimizes the functional $\Phi(f)$ subject to the constraint C [13].

Substituting the definition of G into eq. (10) yields our regularized iterative algorithm:

$$\hat{f}_{k+1} = P[(I-\alpha\beta L^t SL)\hat{f}_k + \beta D^t R(g-D\hat{f}_k)]. \tag{11}$$

The algorithm is considered to be converged if an estimate $\hat{f}_k$ satisfies eq. (2), (3) and the constraint C. Observe that we do not require the theoretical limiting solution $\hat{f}_\ell$ to be computed, but are satisfied with

an estimate in the close neighborhood of $\hat{f}_\ell$. Sufficient convergence conditions are:

(i) a solution described by eq. (2), (3) and the constraint C exists,

(ii) the relaxation parameter β satisfies the bound in eq. (9).

Interpretation of the algorithm

The introduced algorithm is composed of a restoring and stabilizing part. The restoration term $\beta D^t R(g-D\hat{f}_k)$ estimates the correction for the next iteration by comparing the data g with the distorted k-th estimate. The size of the applied correction depends on the value of β, thus regulating the restoration speed. For a finite number of iterations the weighted norm introduced in eq. (2) may now be interpreted as a locally varying relaxation parameter. For example, near edges the large corresponding weight coefficients in the matrix R must enforce a higher restoration speed than in flat regions of the image. Consequently, a trade-off between noise suppression and resolution enhancement is achieved by the matrix R.

In general the regularizing operator L imposes a smoothness condition on the restoration result, hence the stabilizing term $(I-\alpha\beta L^t SL)\hat{f}_k$ acts like a low-pass filter. The regularization parameter α incorporates the global amount of noise into the algorithm. For example, if the data g is noiseless, $\alpha=0$ and the regularizing operator L is disabled. The weighted norm introduced in eq. (3) locally controls the value of α, and hence the strength of the low-pass filtering effect. For example, near edges the coefficients in the weight matrix S take small values to prevent blurring of the edges.

Computation of the weight matrices

To compute the weight matrices R and S we need to know the position of the edges in the original undistorted image. However, only the distorted version is available in which the edges are often very smooth and may even be shifted to a wrong position. The best way out of this dilemma is to compute a non-weighted non-regularized iteration result in advance (using eq. (11) with $\alpha=0$ and R=S=P=Identity), which is an iterative approximation of the inverse filter estimate. This result has

sharp edges, but is very noisy as well. Using the noise suppressing local variance measure from [2], the edges can be estimated quite well from this provisional result. Finally, the weight matrices are calculated from the local variance $\sigma_f^2(i,j)$ as follows:

$$R(i,j) = \frac{\sigma_f^2(i,j)^{\nu}}{\max(\sigma_f^2(i,j)^{\nu})}, \tag{12a}$$

$$S(i,j) = \frac{\min(\sigma_f^2(i,j)^{\mu})}{\sigma_f^2(i,j)^{\mu}}. \tag{12b}$$

4. EXPERIMENTAL RESULTS

Photo 1 shows a real photographically blurred image of size 128x256 pixels. We identified that the train in the image was distorted by horizontal linear motion blur over 8 pixels and by noise with SNR 20 dB. Photo 2 shows the sharp, but also very noisy non-weighted non-regularized iteration result (β=1.0, 100 iterations), which was used to compute the weight matrices R and S. The restoration without making use of these matrices is shown in photo 3 (β=1.0, α=2.0, 100 iterations). No magnified noise can be seen in this result, but the edges are very smooth as well. Finally, the restoration in photo 4 is computed by using the weight matrices and a deterministic constraint C (β=1.0,ν=0.5, α=2.0, μ=2.0, 100 iterations). Because we knew a priori that the image intensities were in the interval [55,125] we used the following projection:

$$P[f(i,j)] = \text{Max}\,[55,\text{Min}(125,f(i,j))]. \tag{13}$$

REFERENCES

[1] R.W. Schafer, R.M. Mersereau and M.A. Richards, Constrained Iterative Restoration Algorithms, Proc. IEEE 69 (1981).

[2] A.K. Katsaggelos, J. Biemond, R.M. Mersereau and R.W. Schafer, Nonstationary Iterative Image Restoration, ICASSP 1985.

[3] Y. Ichioka and N. Nakajima, Iterative Image Restoration Considering Visibility, JOSA 71 (1981).

[4] J.L.C. Sanz and T.S. Huang, Unified Hilbert Space Approach to Iterative least-squares Linear Signal Restoration, JOSA 73 (1983).

[5] J. Biemond and A.K. Katsaggelos, Iterative Restoration of Noisy Blurred Images, Proc. 5th Benelux Symposium Information Theory, Aalten (1984).

[6] A.N. Tikhonov and V.Y. Arsenin, *Solutions of Ill-Posed Problems*, Wiley, Washington, 1977.

[7] R.L. Lagendijk, *Regularized Iterative Image Restoration*, Delft University of Technology, Dep. of Electrical Engineering, Information Theory Group, 1985, Msc. Thesis.

[8] G.L. Anderson and A.N. Netravali, Image Restoration Based on a Subjective Criterion, IEEE trans. SMC 6 (1976).

[9] R.C. Gonzalez and P. Wintz, *Digital Image Processing*, Addison Wesley, Reading Mass., 1977.

[10] D.C. Youla and H. Webb, Image Restoration by the Method of Convex Projections: Part I - Theory, IEEE trans. MI 1 (1982).

[11] K. Miller, Least Squares Methods for Ill-Posed Problems with a prescribed Bound, SIAM J.Math.Anal. 1 (1970).

[12] V.T. Tom, T.F. Quartieri, M.H. Hayes and J.H. McClellan, Convergence of Iterative Non-expansive Signal Reconstruction Algorithms, IEEE trans. ASSP 29 (1981).

[13] J. Biemond and R.L. Lagendijk, Regularized Iterative Image Restoration in a Weighted Hilbert Space, Int. Conf. on Acoustics, Speech and Signal Processing 1986, Japan.

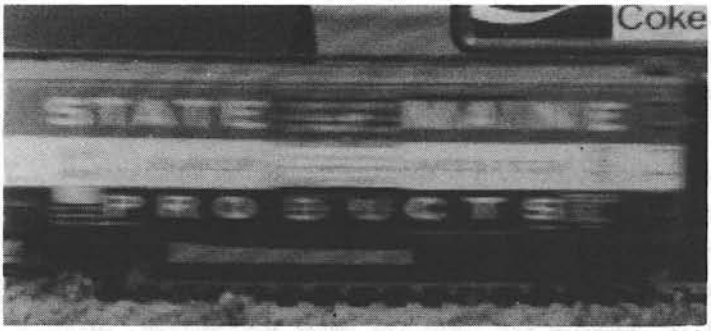


Photo 1.

Photo 2.

Photo 3.

Photo 4.

CLUSAN1: A KNOWLEDGE BASE FOR CLUSTER ANALYSIS

E. Backer, E.J. Eijlers

Cluster analysis as a scientific tool to unravel data is characterized by multiple statistical testing, validation and complex reasoning. Today, it is felt natural to associate such a reasoning process directly with expert systems. This paper is a result of an attempt to develop a knowledge base (CLUSAN1) for the expert system Delfi2 to facilitate the user to obtain validated results of an explorative data-analysis. As a result, the expert system appears to be particularly suitable for potential users which are non-experts but familiarized with the subject matter. Both, the art of knowledge engineering and the resulting structure of the knowledge based are reviewed. A consultation sample will be given in support of the usefulness claimed.

INTRODUCTION

Cluster analysis is known to be one of the major tools in explorative data analysis applicable in many sciences. The analysis of data is characterized by multiple testing, validation and complex reasoning. Usually, quite a number of procedures and routines has to be applied in order to understand the pecularities of the data at hand. The expert-user of a statistical package for explorative data analysis is known by a keen feeling of determining the order in which procedures, routines and validation have to take place. Therefore, it is said that the results of any kind of data analysis is very much determined by a complex reasoning process which may differ from one analysis to another. Nevertheless, a number of common subgoals can be identified, like:

- the validity of the substitution of missing data,
- the validity of a priori labels,
- the detection of outliers,
- the statistical influence of outliers,
- the detection of hinge (or bridging) points,

The authors are with the Delft University of Technology, Department of Electrical Engineering, Information Theory Group, P.O. Box 5031, 2600 GA Delft, the Netherlands

- the statistical influence of hinge (or bridging) points,

to name just a few of them.

These sub-goals are crucial - for example - for estimation the intrinsic dimensionality from a clustering point of view being a next-level goal. To obtain this goal the following statistical procedures (in some order) then have to be used:

- standardization (if necessary),
- correlation analysis,
- eigenvalue analysis,
- discriminant analysis/either on the basis of a priori labelling or on preclustering labelling,
- hierarchical clustering (both on objects and variables),
- hierarchical validation,
- feature analysis.

As mentioned, the above process can be identified as a complex reasoning process. Expert systems have shown their potential usefulness for all kinds of complex reasoning problems. Such a system cannot forget, can combine results and will lead to a guided interpretation of a large set of testing results/properties. Moreover, an appropriate expert system is able to explain the underlying reasoning process explicitly. As such, it provides a valuable tool for less-experienced analysts who can learn from the system itself.

When one is planning to use an expert system for cluster analysis two main requirements show up:

- the expert system should be able to handle numerical problems (most of the existing empty shell expert systems do not fulfill this requirement),
- the expert system should be able to execute external procedures of statistical packages.

In spite of the fact that Delfi2 is not able to handle numerical problems we have chosen for Delfi2 mainly because of the fact that:

- it is developed at Delft University (Computer Science department), so software support is guaranteed,
- it is very easy to link Delfi2 with external procedures which may weaken the requirement for numerical operations within the system.

This paper is the result of an attempt to develop the knowledge base CLUSAN1 for Delfi2. The knowledge base does not have the pretention of being perfect. More important, it was felt valuable to identify and to report on the various aspects of constructing purposive knowledge bases.

In Section 2, the aspect of knowledge engineering is reviewed. Section 3 describes the resulting structure of CLUSAN1.

2. SOME ASPECTS OF KNOWLEDGE ENGINEERING

Evidently, knowledge engineering preceeds the ultimate implementation of the knowledge base. In that, the knowledge engineer plays an important role. His task can be seen as a sequence of five stages.

1. discussion with the expert

Moreoften, the knowledge engineer is no expert in the field; he may have some knowledge of the subject matter, but in general he is unfamiliar with the subject matter. So, a first round of discussions with the expert is needed aiming at a global view upon the subject matter; straight forward recipies, single testing. The expert can advise what to read.

2. literature study

Through literature study the knowledge engineer can familiarize with the jargon characteristic for the subject matter. Various cases may also give rise to common structure and reasoning. Generally, books are very suitable. Journal papers tend to be too detailed and may distract the knowledge engineer from simplification,

3. discussion with the expert about the structure

A first attempt to formulate a - possibly still naive - structure of the reasoning process should be discussed now with the expert. These discussions may iterate towards a final structure.

4. discussion with the expert about drawing conclusions (heuristics)

Once the structure has been fixed the knowledge engineer will focus on the process of drawing conclusions:

- what is being concluded,
- under what conditions can be concluded,
- how certain are conclusions.

Because of the complexity of the reasoning process and used heuristics underlying the drawing of conclusions, initially it is necessary to simplify and generalize. At this stage the expert should be asked to write down some samples of reasoning. Meanwhile the expert becomes aware of the general format of production rules. These discussions do result in the generation of production rules.

5. validation of the knowledge base

After that the production rules have been formulated and implemented the knowledge base should be tested and validated in practice. Again known sample consultations have to be taken as test samples. In this stage the expert should be asked to tune the knowledge base. Special attention to completeness should be given.

The above learning phase may converge if only one expert is involved. If more experts are involved, one may expect conflicting strategies and heuristics. Additional sessions may be needed to resolve conflicting issues.

3. THE STRUCTURE OF CLUSAN1

An example of a context structure is shown in Fig. 1. The context a-priori knowledge checks the possibility of analyzing the set of data. The left path is followed by the analysis of pattern matrices. The right path leads to the analysis of object-object relation matrices. The context pattern-matrix is responsible for some initial data processing (e.g. elimination of constant and redundant variables). The context object-object-matrix is responsible for the initial processing of relation matrices. The path which then has to be followed depends on the scale(s) of the variables. The context hier-clus-var is responsible for the hierarchical clustering of objects. Note that a final context is also linked by production rules of earlier applied contexts (e.g. the context hier-clus-var is connected by production rules with the context ratio-pat).

The contexts for hierarchical-clustering fixes almost the sequence of tests to be done. This is caused by the fact that write-commands can only be implemented in the conclusion part of the production rules. However, the questions about the testresults can only be activated in the premisses of production rules. Therefore, there are two rules necessary to give a write-command and to ask for the testresult. To give the consultation a logical sequence, that means that the testresults are asked directly after the activation of the test, it is necessary to fix the sequence of tests.
Four aspects of the structure of CLUSAN1 are now to be discussed:

1. subconclusions

For large knowledge bases it is good to work towards sub-conclusions. For example, in CLUSAN1 the conclusion about the existence of outliers must be drawn before a conclusion can be drawn about the statistical influence of outliers. Introducing sub-conclusions has at least three advantages:
- more easy to keep an overview of the knowledge base,
- more easy to test the completeness of the knowledge base,
- more easy to debug.

2. representation of conclusions

Two types of conclusions can be distinguished:
- conclusions which can be drawn as a result of one parametervalue,
- conclusions which can be drawn as a result of more than one parametervalue.

The first type can be represented by a scale like: nihil, small, significant, large, very large, where the certainty factor is always 1.00. The second type comes with one parametervalue to which a variable certainty factor [-1.00, 1.00] is attached.

3. interaction of certainty factors

Two types of rules must be distinguished:
- strategy rules,
- concluding rules.

The strategical rules form the skeleton of the reasoning process and

have constant certainty factors of 1.00. The concluding rules are responsible for the drawing of (sub-)conclusions. The certainty factors for these rules are variable. The conclusions of the concluding rules never occur as clauses in the strategical rules. By this distinction it is possible to keep the certainty factors free from distortion.

4. the certainty factors and the user-interface

The implementation is made less complex by prohibiting the answering with certainty factors. Therefore, the user can select a parametervalue out of a detailed set of legal values. At the same time this method has the advantage that the user does not need to answer with hard numerical values.

5. CONCLUSIONS

The knowledge base which has been developed for cluster analysis has proven to be applicable in the training consultations, however, further testing and tuning is necessary to validate its usefulness. The expert system approach may serve well as a training system for the unexperienced cluster analyst because of the extensive explanation facility.

LITERATURE

[1] Romesburg, H.C.: Cluster Analysis for Researchers, Belmont, 1984.

[2] Swaan Arons, H. de, Lith, P. van: Expert systemen, Den Haag, 1984.

legenda:
1. a-priori knowl
2. pattern-matrix
3. obj-obj-matrix
4. mixed-pat
5. ratio-pat
6. nominal-pat
7. ordinal-pat
8. interval-pat
9. ratio
10. binary
11. par-clus
12. hier-clus-var
13. hier-clus-obj

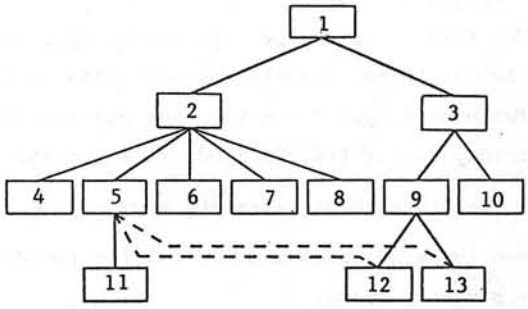


Figure 1. An example of a context-structure in CLUSAN1

Appendix: A sample consultation.

```
DELFI-2 CONSULTATION SYSTEM

Does an a-priori labeling exist?
>> <RETURN>
Legal parameter values:
Present                                 not present
>> n
= not present (1.00)

Please give an estimate of the number of populations?
>> 2

What kind of matrix is the data-matrix?
>> <RETURN>
Legal parameter values:
object-object-matrix                    pattern-matrix
>> p
= pattern matrix (1.00)

What kind of scale have the variables?
>> <RETURN>
Legal parameter values:
mixed          nominal          ordinal       interval      ratio
>> r
= ratio (1.00)

//

What is the number of objects?
>> 100

What is the number of variables?
>> 15

------------------------------------------------------------------
WARNING:
The ratio [# objects]/[# variables] * [# populations]
is too small to analyse this problem.
------------------------------------------------------------------
Although I advise you to stop, what do you want?
>> <RETURN>
Legal parameter values:
1. the-user-follows-up-the-advise
2. the-user-wants-to-ignore-the-advise
>> 2
= the-user-wants-to-ignore-the-advise

What kind of analysis is being wanted?
>> <RETURN>
Legal parameter values:
1. variable analysis                    2. object analysis
3. both
>> 1
= 1. variable analysis

//

What do you know about the existence of missing data?
>> <RETURN>
Legal parameter values:
no-missing-data-present                 missing-data-present
>> m
= missing data present (1.00)
```

```
-------------------------------------------------------------------------
action                    : the substitution of missing data
-------------------------------------------------------------------------
//
-------------------------------------------------------------------------
context                   : hierarchical clustering variables
action                    : normalize results to total scatter
                          : discriminant analysis
                          : discriminant analysis with
                            eliminated outliers
-------------------------------------------------------------------------
What is the result of discriminant analysis with/without outliers?
>> <RETURN>
Legal parameter values:
decrease-between-scatter:
1. no      2. small      3. significant      4. large      5. very large
increase-between-scatter:
6. no      7. small      8. significant      9. large     10. very large
decrease-within-scatter:
11. no    12. small     13. significant     14. large     15. very large
>> 9 la
= 9. large-increase-of-the-between-scatter (1.00)
//
//
-------------------------------------------------------------------------
The intrinsic dimensionality is: 3 with certainty: 0.90
The intrinsic dimensionality is: 2 with certainty: 0.76
-------------------------------------------------------------------------
//
DELFI-2 ACHIEVEMENT SUMMARY
concluded: the intrinsic dimensionality of the hierarchical  clustering
           in variable-analysis is:
        1: is-OKAY with certainty 1.00
concluded: the existence of clusters which behave irregular of the hier-
           archical clustering in variable-analysis is:
        1: UNKNOWN with certainty 1.00
concluded: the influence of outliers of the hierarchical-clustering in
           variable-analysis is:
        1: LARGE with certainty 1.00
concluded: outliers of the hierarchical-clustering in variable-analysis
           is:
        1: DO-EXIST with certainty 1.00
concluded: the validity of the labels of the hierarchical-clustering in
           variable-analysis is:
        1: OKAY with certainty -0.90
concluded: the validity of the correction of missing data of the hierar-
           chical-clustering in variable-analysis is:
        1: OKAY with certainty -0.90.
//
```

SELF SIMILAR HIERARCHICAL TRANSFORMS: a bridge between Block-Transform coding and coding with a model of the Human Visual System

G.H.L.M. Heideman*, H.E.P. Tattje*,
E.A.R. van der Linden**, D. Rijks***

Hierarchical Transforms for time (or spatial) discrete signals are presented. Such Transforms include some familiar orthogonal Block-Transforms, but also non-orthogonal and non-Block Transforms.
Therefore, the degree of freedom of choosing basisfunctions is much larger. Within this family a subclass exists that approximates closely the operations that are performed by the Human Visual System.

INTRODUCTION

At the moment the main concern in Transform Coding is clustering the coefficients in classes (zones, scanning rules, etc.) and quantization schemes. The Transform itself is only a minor point in the discussion. Within the family of Block Transforms the K.L. Transform is mentioned as the best under some specific constraints.
At this place we don't want to critisize the relevance of these constraints in coding applications at length, but it is certain that such an optimum is only an optimum on the average.
Subjectively we do not judge a coded image on the average, but we want that each coded realisation is a natural image with a distortion as low as possible. One of the main drawbacks of Block Transform is in our opinion that each basisfunction has the same spatial support. Why should we base, for instance the measurement of the presence of a low "frequency" component on only one period and the highest "frequency" component on several periods?
If we for instance enlarge or diminish an image, then with a Block Transform, the coefficient of a specific "frequency" is changed because it is measured over more respectively less periods. It is more desirable to have a Transform, that measures spatial "frequency" on a support that increases inversely proportional with "frequency".

* Technische Hogeschool Twente, Afd. Elektrotechniek
Postbus 217, 7500 AE Enschede
** Now with Océ-Nederland B.V.
Postbus 101, 5900 MA Venlo
*** Now with Philips B.V.
Postbus 80000, 5600 JA Eindhoven

Such a Transform gives coefficients that are independent of the scale of the image. Hierarchical Transforms can meet these requirements. At this place we want to mention the strong relationship between a hierarchical description of images and the theory of Fractals[1]. In a realistic image model we have to use the same basisfeatures at different spatial scales.

Such a description leads to a separation of an image in different acuity classes. The basis features (the basisfunctions of the Transform) that are needed in this description can be related to the characteristic operations of neurons in the Human Visual System.

HIERARCHICAL TRANSFORMS

In this paragraph we shall introduce Self Similar Hierarchical Transforms. Before doing so, we relate Block-Transforms to multi-channel sampling models, because it gives more insight to look at Transforms as sampling models, especially in the case of Hierarchical Transforms, then to describe them by matrix formulations.
In what follows we describe only 1-D-Transforms, but it is quite straightforward to define n-D-Transforms in a similar way.
Let the signal x be defined as a sequence

$$\{x(n)\} \; ; \quad n=0,1,2,....M-1$$

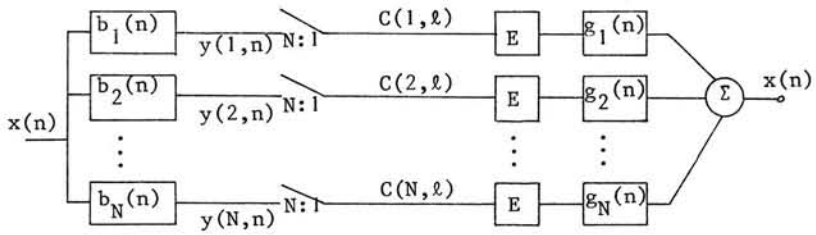
obtained by observing a finite segment of a sampled continuous wave form.
The coefficients of a Nth order linear Block-Transform T of this sequence (with M=L.N; L is an integer) are:

$$C(k,\ell) = \sum_{\ell N}^{(\ell+1)N-1} x(n)f(k,n) \qquad \begin{matrix} \ell=0,1,... \; L-1 \\ k=1,..... \; N \end{matrix}$$

The functions f(k,n) are the basisfunctions of the Transform T and form an orthonormal set if T is orthogonal.
From the coefficients C(k,ℓ) we can reconstruct the original signal x

$$x(n) = \sum_{k=1}^{N} C(k,\ell)r(k,n) \quad \text{for } \ell N \leq n \leq (\ell+1)N-1 \qquad \begin{matrix} \ell=0,1,... \; L-1 \\ k=1,..... \; N \end{matrix}$$

The functions r(k,n) are the basisfunctions of the inverse Transform T^{-1}. The same coefficients can be extracted from the signal x by the following multi-channel sampling model.



with $b_i(n) = f_r(i,n) = f(i,-n)$
$g_i(n) = r_r(i,n) = r(i,-n)$
the subscript r means: time-reversed.

Figure 1

The right part of the scheme represents the reconstruction of x.
E[xpand] means: filling in N-1 zeros between two adjacent coefficients C(k,ℓ) and C(k,ℓ+1) for each ℓ.

This scheme shows that we have a multi-channel sampling model with N filters with the functions $b_K(n)$ as finite impulse responses.
Sampling the outputs y(k,n) at points n=ℓ.N-1, gives the coefficients C(k,ℓ).
Remark that the length of the finite impulse responses is equal to the order of the Transform and equal to the sampling period.

The scheme above is a special case of the general multi-channel sampling model for time contineous band limited signals with equal and synchroneous sampling for all channels.
In such a general sampling model the choice of the filters, $b_1,\ldots b_N$, is free within a constraint[2]. Orthogonality of the impulse responses is not necessary. The reconstruction filters, $g_1,\ldots,g_N$ can always be calculated from the filters, $b_1,\ldots,b_N$.
In such a general sampling model the filters $g_i(n)$ can be found from the filters $b_i(n)$ from the following relations.

Let $$B_i(\omega) = \sum_{n=0}^{N-1} \varepsilon^{j\omega n} b_i(n) \qquad |\omega| \leq \pi$$

and $$G_i(\omega) = \sum_{n=0}^{N-1} \varepsilon^{j\omega n} g_i(n) \qquad |\omega| \leq \pi$$

Define $\bar{B}_i(\omega+k.\sigma_1)$ as a periodic extension of a shifted $(k.\sigma_1)$ version of $B_i(\omega)$ with $\sigma_1 = 2\pi/N$ and the matrix $\bar{B}(\omega)$

$$\bar{B}(\omega) = \begin{vmatrix} \bar{B}_i(\omega) & \cdots & \bar{B}_N(\omega) \\ \bar{B}_i(\omega+1.\sigma_1) & \cdots & \bar{B}_N(\omega+1.\sigma_1) \\ \vdots & & \vdots \\ \bar{B}_i(\omega+(N-1).\sigma_1) & \cdots & \bar{B}_N(\omega+(N-1).\sigma_1) \end{vmatrix}$$

Define the vector $\underline{G}(\omega) = \begin{vmatrix} G_1(\omega) \\ G_2(\omega) \\ \vdots \\ G_N(\omega) \end{vmatrix}$ and $\underline{e}_1 = \begin{vmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \end{vmatrix}$

The solution of

$$\bar{B}(\omega).\underline{G}(\omega) = \underline{e}_1$$

describes the filters $G_i(\omega)$.
A solution $\underline{G}(\omega)$ exists iff

$$\text{DET } \bar{B}(\omega) \neq 0 \qquad \omega \in (-\pi,\pi)$$

In the case of Block-Transform this condition reduces to

$$\text{DET } \bar{B}(\omega) = \text{Constant . Det T}$$

A multi-channel sampling model can be easily extended to Hierarchical Sampling models. The following scheme gives the simplest one, with two channels at each of the two levels in the Hierarchy.

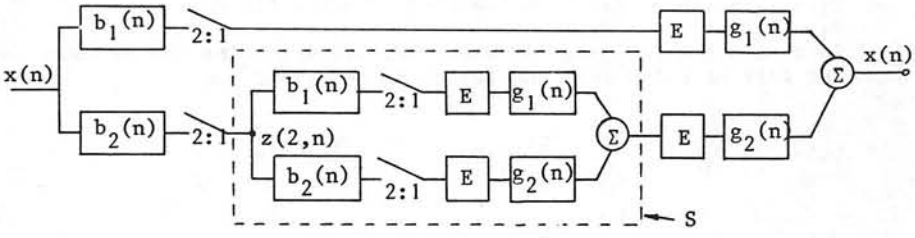


Figure 2

What we actually have done is putting again a two-channel sampling at the output of one (typicaly the low-pass-channel) of the first channels. The multi-channel sampling model ensures that we can reconstruct precisely z(2,n).

At the reconstruction side we have to reconstruct first z(2,n) by means of two filters $g_1(n)$ en $g_2(n)$ and put the sum of their outputs after another expansion at the input of the $g_2(n)$-filter of the first level.
The length N of the impulse responses $b_1(n)$ en $b_2(n)$ is equal to 2, and so is the sampling period.

The number of levels in the hierarchy is two, but can be extended maximally to K levels if $M=2^K$, by repeating the system S after each $b_2(n)$-output. Such a scheme with K levels is equivalent with:

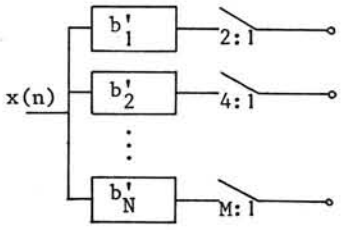


Figure 3

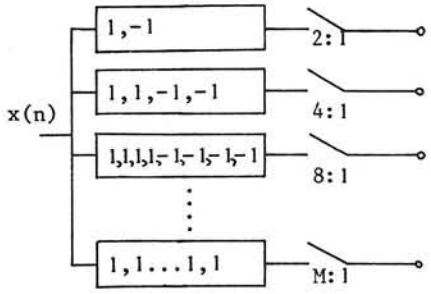


Figure 4

The impulse responses $b'_1, b'_2, \ldots, b'_{K+1}$ in this scheme are defined as follows:

$$b'_{K+1}(n) = (b_{2,K} * b_{2,K-1} * \ldots * b_{2,1})(n)$$
$$b'_j(n) = (b_{1,j} * b_{2,j-1} * b_{2,j-2} * \ldots)(n) \qquad j=1,\ldots,K$$

and $b_{2,k} = \delta(n)$ if $k \leq 0$

with
$$b_{1,j}(2^{j-1}.\ell+i) = b_1(\ell) \quad \text{if } i=0$$
$$= 0 \quad \text{if } i=1 \qquad j=2,3,\ldots$$

and
$$b_{2,j}(2^{j-1}.\ell+i) = b_2(\ell) \quad \text{if } i=0$$
$$= 0 \quad \text{if } i=1 \qquad j=2,3,\ldots$$

$$b_{1,1}(n) = b_1(n)$$
$$b_{2,1}(n) = b_2(n)$$

In fact we have extended the multi-channel sampling for time-discrete signals with equal sampling periods to a sampling model with unequal sampling periods.

If we choose $b_1=(1,-1)$ and $b_2=(1,1)$ then this system is known as the Haar-Transform of rank M (fig. 4). Members of the same family of Transforms (Ter- and Her-Transform for instance) are easily obtained if we put the systems S also one or two times after the $b_1(n)$-output It is straightforward to formulate schemes with more than two, say P, channels at each level of the hierarchy, if $M = P^q$; $q=1,2,\ldots$.

So far we have used orthogonal basisfunctions at each level of the hierarchy, but it is no problem at all to define non-orthogonal Hierarchical Systems as well.
The following scheme gives an example with non-orthogonal basisfunctions of length 3.

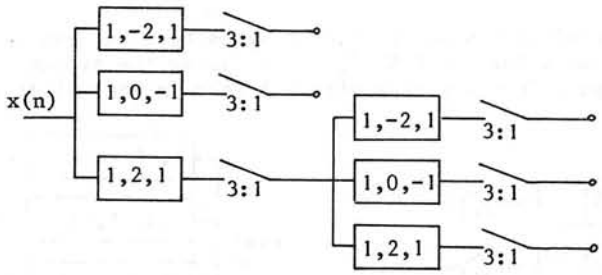


Figure 5

Until here we have used impulse responses with length equal to the sampling rate, so in fact all such Transforms can be seen as hierarchical Block-Transforms, both orthogonal and non-orthogonal. But our aim was not only to get rid of the demand for orthogonality but also to get rid of the relation: length of the impulse response is equal to the sampling period.
The question is: can we define Hierarchical Transforms (with an inverse) with basisfunctions unequal to the sampling period.
The answer is yes, we can. There exists always a specific way of sampling the output of the filters, delivering M coefficients, necessary and sufficient for the reconstruction of x(n), but such a sampling scheme is not necessarily synchroneous (or equal) for all the channels and some times also not homogeneous. Usually there exists more than one possible sampling scheme. These possible schemes split up in the two classes.

A member of the second class is the following scheme, known as the Pyramid Transform[3]

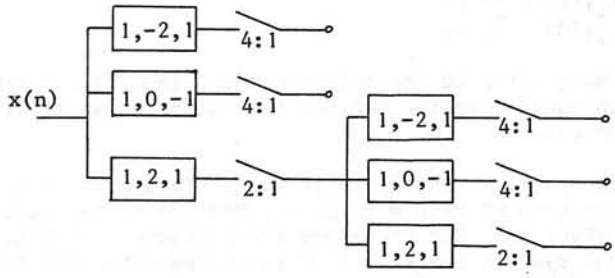


Figure 6a

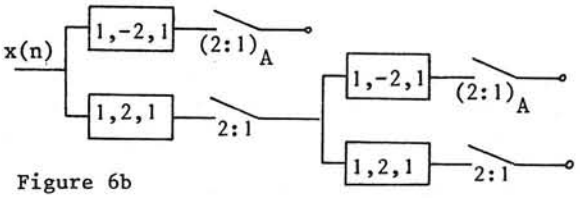


Figure 6b

A member of class 1 can be found by a slight modification of the class 2-scheme. This modification is skipping the b_2-filter and increasing the sampling rate of the b_1-channel to $(2:1)_A$.

With this notation is ment, that the sampling rate is (2:1) on the average, but that the sampling period is not homogeneous.
Two possibilities for such inhomogeneous sampling schemes are:

1) x x . . x x . . x x . . if M is multiple of 4
2) x x x . . . x x x . . . if M is multiple of 6

x and . are points on the original raster
x are the sampling points of the b_1-channel.

Such schemes have the disadvantage that the reconstruction scheme is more complex, not always local and always in-homogeneous. In-homogeneity means that the number of samples that are needed for the reconstruction of a specific point x(n) is dependent on n. Experiments show that this in-homogeneity makes the transform more sensitive for quantization of the coefficients.
Especially if we use these Transforms in coding for low bitrates, this disadvantage is great, because low bitrates can only be achieved by coarse quantizing. Thus we like to have Transforms with a homogeneous reconstruction scheme. This can be achieved only by increasing the sampling rate of the b_1-channel (class 1-scheme) to the possible maximum (1:1). The price we pay is that we need twice as much samples, but the gain is a simpler reconstruction scheme and the possibility of a coarser quantization. We call such Transforms redundant.

SELF SIMILARITY

In all the schemes presented before we use at each level of the hierarchy the same systems b_i after each sampled low-pass output from the previous level. In this sense the Transform is Self-Similar. This Self-Similarity has the effect that the impulse responses of a chain of repetitions of low-pass filters and decimations, followed at the end by a band-pass system, are approximately of the same form, though on a different scale. This approximation is better if the impulse response of the low-pass filters is such that the fixed sampling rate is close to the allowed sampling rate for that particular filter. Ultimately there is no difference between a system that uses sampling at each level in the hierarchy and a system with the same overall impulseresponse that uses sampling only at the end of a chain.

A VISION MODEL

Measurements of the characteristics of neurons in the early visual pathway show that there are neurons with receptive fields that are of a circular form and show an excitation (positive) center and an inhibition (negative) surround or vica versa. Such fields are found with a variety of spatial dimensions.
Apart from non-linearities in the neuronal systems, we can model such neurons as 2-D-bandpass filters with circular symmetric impulse responses of the same form but with a variety of scales.
Marr`and others suggested that these impulse responses can be modelled as 2D-Difference of Gaussian functions or as a Laplacian of a Gaussian function. Such a processing can be put in the following scheme

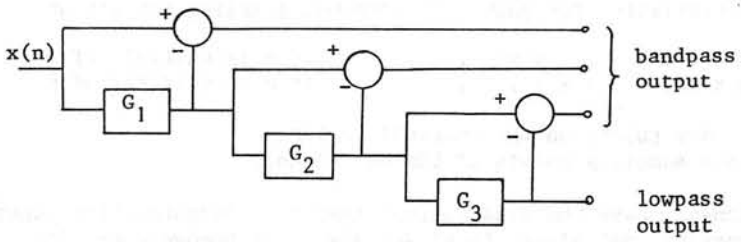


Figure 7: Hierarchical model with a Difference of Gaussian-filter.

or in the following scheme

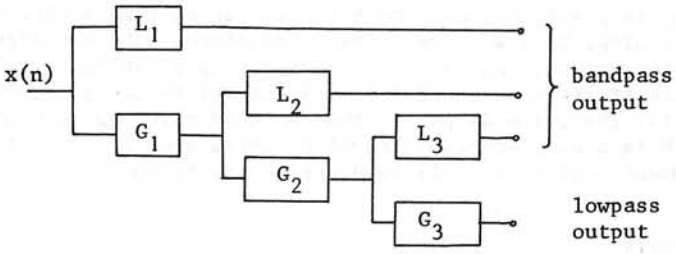


Figure 8: Hierarchical model with a Laplacian of a Gaussian-filter.

In these schemes the Gaussian filters G_i have a σ_i with $\sigma_1<\sigma_2<\sigma_3$.... The L_i-filters are Laplacians. It is easy to see that this part of a vision model can be interpreted as a Self Similar Hierarchical Transform.

It follows from the foregoing that it is allowed to sample the outputs of the L_i-outputs or the DOG-outputs at a (with i-)decreasing rate.

As we have in image processing usually a spatial orthogonal or quincunx sampling raster, the decimation has to be 2^k:1 with k=1,2,... If we want to decimate the rate of the outputs with a 4:1 scheme than the σ_i-parameters of the 2-dimensional gaussian filters have to fullfil the following relation:

$$\sigma_i^2 = 3.4^{i-2}.\sigma_1^2 \qquad i=2,3,...$$

We already mentioned that sampling at the end of the chain can be replaced by sampling after each level in the hierarchy, if $\sigma_1=\Delta/\sqrt{2}$, with Δ = distance between successive samples of x. In that case we get the following system (with a decimation of 4:1 at each level)

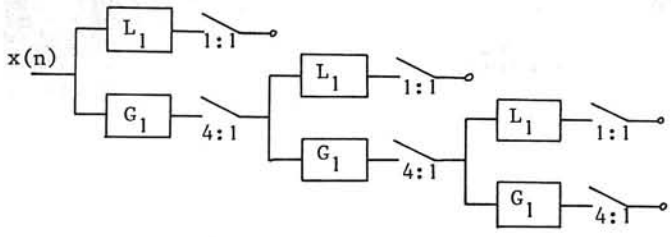


Figure 9

In this system all the G_i-systems are the same, and so are the L_i-systems. Thus finally the structure of this vision model is precisely the same as the redundant Self-Similar non-orthogonal Hierarchical Transform.

But, if we use only the L_1 system, then there is no exact reconstruction scheme. However, the approximation is so good, that it is hard to see any difference between the original and the reconstruction. Recently we found that a perfect reconstruction exist if we use three different L-filters instead of one L_1-filter. The sampling rates of these L-filters are 4:1, 4:1 and 2:1, so (1:1) on the average. The number of samples of the bandpass outputs is in that system not increased. So we state: the vision model with only one L-operator, the Anthropomorfic Transform[5], is a close approximation to a Self Similar Hierarchical Transform with a true inverse.

DISCUSSION

Self Similar Hierarchical Transforms, especially the ones that approximate the vision model have nice properties that Block-Transforms don't have. With Block-Transforms, for instance, if we arrange the coefficients with the same index in an array, such an array doesn't look like a real image. Only the array of the (0,0)-coefficients of each block looks like a low-pass version of the original image. In contrast, the outputs of Hierarchical Transforms are more like meaningfull images.

The connection between local structures in the image is better preserved.

ORIGINAL

OUTPUT

This property can be used for further coding, for instance with vector quantization or, as we call it, local equivalent representation. Block-Transforms and also some of the Hierarchical Transforms are not suitable for such procedures.
Another advantage of the Antropomorfic Transform is, that the low-pass output at each (k-th) level of the hierarchy is a spatially "factor 4^k-scaled down" copy of the original image with precisely the same resolution that we reach if we scale down the original image with the same factor 4^k or look at the image at a distance 2^k as large as the original viewing distance.

Thus we separate the image in a number of images, each of them representing a different acuity. Experiments show that the sensitivity of the vision system is different for this acuity classes, thresholds are higher and quantization regions larger for the higher acuity classes.
Another advantage is the freedom in the choice of the basis functions. This gives the opportunity to construct Transforms with the use of additions and subtractions only, without the disadvantage of the Hadamard Transform family, that all values of the basis functions are equal to +1 or -1.

A disadvantage of the, for coding most suitable Hierarchical Transforms is their redundancy. For a (N*N)-image the number of possible sample values or coefficients is equal to $4/3\ N^2$.
All together, Self Similar Hierarchical Transforms are very promising for image modeling, coding and maybe the wider field of general image processing.

CONCLUSIONS

In the fore-going we have defined Self-Similar Hierarchical Transforms These are a subclass of general Hierarchical Transforms in the sense that at each level in the hierarchy the same filter operations are used. This has the effect that the equivalent basisfunctions for each level are resampled versions of one standardform.
The use of such Transforms in coding for low bit rates does not give the familiar blocking impairments. Besides this advantage, it is also possible to choose the basisfunctions of the transform in such a way, that they are similar to the sensitivity-profils of receptive fields of neurons in the primary visual cortex (the early pathway in the image processing of human observers).
These basisfunctions can be modelled easily by a multiple convolution of (1,1) and (1,-1) functions [Binomium of Newton), resulting in a close approximations of a Difference of Gaussian function.
The use of the only (1,1) and (1,-1) functions has the advantage that only additions are needed in the calculations of the output of the filters. This confirms our opinion that also the human visual system uses additions (excitation) and subtractions (inhibition) only.

So the final conclusiuon is that Self-Similar Hierarchical Transforms cover on the one side some familiar Block-Transforms and on the other side a nice model for the human visual system. Using a self-similar transform as a model for the visual system makes it much easier to design classification and quantization rules that are close to the sensitivity rules of the visual system.

REFERENCES

1. Benoit Mandelbrot: "Fractals; Form, chance and dimension". W. Freeman and Co., San Fransisco.
2. Brown J.L. jr.: "Multi-channel sampling of Low-Pass Signals". IEEE Transactions on Circuits and Systems, vol. CAS-28, no. 2, febr.'81.
3. Meeker G.: "Triangle and Pyramid Signal Transforms and Apparatus". U.S.A. Patent 4, 447, 886, May '84.
4. Marr D.: "Vision". W. Freeman and Co., San Fransisco '82.
5. Heideman G.H.L.M., Wanschers L.H.: "Application and Performance of a Human Observer Oriented Transform for image coding with low bitrates". Proceedings of the International Zurich Seminar on Digital Communication, March '84, IEEE Cat.no. 84, CH 1998-4.

PROPERTIES OF MOTION ESTIMATION IN THE TRANSFORM DOMAIN

R.H.J.M. Plompen*, J.G.P. Groenveld*, D.E. Boekee** and F. Booman*

In this paper we extend the transform domain oriented estimation algorithm introduced in [1] in which the calculation of the displacement vector was obtained from the transform domain coefficients. The performance of the algorithm is verified within a hybrid coding configuration. In this paper only transform domain block matching algorithms are considered. The block-match procedure makes use of the displacement matrix H defined in [1]. A matrix decomposition method is described in order to show that a practical implementation is very well possible. The properties of the translation invariant matrices are explained by using the ordered Walsh Hadamard transform as an example. The procedure, however, enables the use of any other orthogonal transform. An important issue with respect to the hardware complexity of this motion compensated hybrid coder is the use of only one transform. The performance of the proposed new algorithm is shown and a video tape containing a very critical videoconferencing scene (i.e. split screen and a hard switch to full screen with heavy motion) will be presented.

I. INTRODUCTION

For very low-bitrate codecs used for interpersonal videocommunication including videoconferencing, it is necessary to remove redundancy and to allow the introduction of some degradation.

The exploitation of motion compensation techniques is useful for efficient coding but the hardware complexity should be kept in mind. The success of motion compensation schemes that have been introduced depends to a large extent on the accuracy that they obtain in the motion estimation. A practical criterion for their applicability is that they must be rather insensitive to the preprocessing that is usually applied to image sequences. For areas of the image which are detected as changed, the method must be able to determine a displace-

* PTT, Dr. Neher Laboratories, Transmission Section, P.O. Box 426, Leidschendam

** Delft University of Technology, Information Theory Group, P.O. Box 5031, Delft

ment vector. The performance of the displacement estimation algorithm highly depends on the ability to determine the best match. In a hybrid coding scheme (i.e. transform coding in combination with an interframe prediction) the estimation is usually calculated in the pixel domain, whereas the actual compensation takes place either in the transform or pixel domain. In order to achieve a better performance and a less complex hardware realization the complete coding should be performed in the transform domain.

The different operations and techniques like the change detector, quantization and the estimation used in the configuration can now be optimized while the methods are calculated in the same domain.

This paper will demonstrate that the matching in the transform domain will yield a better image quality. To realize motion estimation based on matching in the transform domain displacement matrices H need to be used. The sensitivity of the estimation is influenced by a frequency weighting function.

In literature several displacement estimation algorithms have been proposed [3]. Except the full search algorithm all the algorithms based on block matching are suboptimal. It is not guaranteed that the suboptimal ones will find the global minimum. These algorithms will minimize the prediction error, known as the displaced frame difference (DFD) only. They can all be characterized as best-match methods. The displacement vector is obtained as follows:

$$DFD = \min_{k,\ell \in SW} \{e_{pred}(q_{k,\ell})\}, \qquad (1)$$

$$k,\ell = -(N-1),\ldots,0,1,2,\ldots,N-1$$

where DFD is the displaced frame difference and SW is search window in the previous frame.

$$e_{pred}(q_{k,\ell}) = |f\{q,t\}-f\{q,t-1,D\}|^a \qquad (2)$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{N} |f\{q(i,j),t\}-f\{q_{k,\ell}(i,j),t-1\}|^a \qquad (3)$$

where $f\{q_{k,\ell}(\circ),t\}$ is the actual subblock and $f\{q_{k,\ell}(.),t-1\}$ is the subblock in the previous frame.

Because of the suboptimal solution the algorithms already mentioned do not guarantee that a displacement vector is to be found with sufficient accuracy. Noise, rotation, zooming and occlusion can possibly cause considerable inaccuracy. If the aim of the coding procedure is only to decrease the prediction error it will not be necessary to calculate the real displacement vector. Pairs of displacement matrices [1], [5] have cyclic properties.

II. INTRODUCTION DISPLACEMENT MATRIX

Let transform T be an orthogonal transform and let h be a nilpotent operator of index N. Then h has a block diagonal matrix representation of the form:

$$h_1 = \begin{bmatrix} 0 & \vdots & \\ & \vdots & I_{N-1} \\ - & \vdots & \\ 0 & \vdots & 0 \end{bmatrix} . \tag{4}$$

The major properties of h are:

$$h_1(\Delta) = h_1^{\Delta} \quad \text{e.g.} \quad h_1(2) = h_1^2 = h_2, \tag{5a}$$

$$h_1^N = 0, \tag{5b}$$

$$h_{\Delta-N} = h_{N-\Delta}^t \quad , \text{ denote t as transpose} \tag{5c}$$

$$f(q)\, h_x \Rightarrow \text{horizontal shift} \quad \begin{matrix} x > 0 \text{ to the right} \\ x < 0 \text{ to the left} \end{matrix} \tag{5d}$$

$$h_y f(q) \Rightarrow \text{vertical shift} \quad \begin{matrix} y > 0 \text{ up} \\ y < 0 \text{ down.} \end{matrix} \tag{5e}$$

In order to prevent discontinuities (e.g. no gaps between shifted blocks) the nilpotent operators are always used in pairs with indices Δ and $\Delta-N$. Let $f(q-1,t-1)$, $f(q,t-1)$ and $f(q+1,t-1)$ be three subblocks of size NxN in the previous frame, and assume translation only with

D = (x,0) and x > 0. Then:

$$f(q,t-1,D) = f(q,t-1)\ h_x + f(q-1,t-1)\ h_{x-N} \tag{6}$$

where f(q,t-1,D) is the compensated translated subblock q.

The sum of the pairs of displacement matrices hΔ and hΔ-N are taken to get cyclic matrices described by using subscript c. Using the newly defined operator h^{cl} the subblock q and q-1 become a cylinder. The operator h^{cl} rotates the cylinder along its axis. To obtain the displaced subblock another matrix e(trunc) is introduced:

$$e_N^{\Delta} = \left[\begin{array}{c|c} I_{N-\Delta} & 0 \\ \hline 0 & 0 \end{array}\right]. \tag{7}$$

A combination of both the matrices $h_N^{c\Delta}$ and e_N^{Δ} will result in the same displacement matrix h_{Δ}

$$h_{\Delta} = e_N^{\Delta}\ \ h_N^{c\Delta}. \tag{8}$$

Given the properties of the unitary transformation matrix T, the method described can also be used in the transform domain

$$T(f\ h) = F\ H, \tag{9}$$

where capital characters are used for the calculation in the transform domain i.e. h,f and e becomes H, F and E. Due to the separability of the transform used the displacement matrix becomes:

$$H_{\Delta} = T_c\ \ e_N^{\Delta}\ \ h_N^{c\Delta}\ \ T_r^t, \tag{10}$$

where T_c and T_r are operators on columns and rows resp. The translation invariant matrix becomes:

$$h_N^{c\Delta} = h_{\Delta} + h_{\Delta-N}, \tag{11}$$

and because of (9) and (10),

$$H_{\Delta} = E_N^{\Delta} \; H_N^{c\Delta}. \tag{12}$$

The displacement matrix h (and H) is non-singular. The reversable operation does not exist; data is shifted out of the considered subblock q. The displacement matrix in the pixel domain contains a lot of zeroes, on the other hand the transform domain displacement matrices can be decomposed with a shift-in-place algorithm.

The properties are a valuable tool for the block-matching techniques. Block-matching can be formulated as the search for a reference image, (the actual subblock) within a larger image, (the search area).

Substituting (12) in (6) and applying (9) and (5) then yields

$$F(q,t-1,D) = F(q,t-1)\; H_{\Delta} + F(q-1,t-1)\; H_{\Delta-N'} \tag{13}$$

and

$$F(q,t-1,D) = F(q,t-1)\; E_N^{\Delta}\; H_N^{c\Delta} + F(q-1,t-1)\; H_N^{c\Delta}\; E_N^{N-\Delta}. \tag{14}$$

III. DECOMPOSITION IN SPARSE MATRICES USING THE ORDERED HADAMARD TRANSFORM

The key in developing the desired fast algorithm is the ability to use Kronecker products in the matrices H and E in order to decompose the matrices into sparse matrices. In order to explain this the ordered Hadamard transform is used for simplicity.

$$H_N^{c\Delta} = H_{\Delta} + H_{\Delta-N'} \tag{15}$$

with $\Delta = 1,2,\ldots,N-1$.

The smallest cyclic matrix is H_2^{c1} and can be formulated as:

$$H_2^{c1} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{16}$$

Odd and even shifts are explained separately:

$$H_N^{c\Delta} \rightarrow \begin{cases} \Delta = 1,3,\ldots,N-1 \\ \Delta = 2,4,\ldots,N-2 \end{cases} .$$

In order to expand the matrices, for example from order N to 2N, all the matrices of order N can be used to generate newly expanded matrices of order 2N.

So for the generation of $H_N^{c2\Delta}$ using $H_{N/2}^{c\Delta}$ the following matrix structure results:

$$H_N^{c2\Delta} = \left[\begin{array}{c|c} H_{n/2}^{c\Delta} & 0 \\ \hline 0 & -H_{N/2}^{c\Delta} \end{array}\right] .$$

Because of the symmetry properties:

$$H_N^{c(\Delta-N)} = [H_N^{c\Delta}]^t \text{ for } \Delta > N/2. \qquad (17)$$

In the case of blocksizes N > 2, previously generated matrices can be used for the expansion in the case of an even displacement.

Δ	→				
N = 16	2	4	6	8	N
N = 8	1	2	3	4	
N = 4		1		2	↓
N = 2				1	

For odd displacements the indices become:

N = 16	1	3	5	7
N = 8	1	3	5	
N = 4	1			
N = 2	1			

IV. BLOCK MATCHING USING FREQUENCY WEIGHTING

Because manipulations in the frequency domain are more easily calculated, a more effective frequency weighting function is used. In

order to obtain a search algorithm sensitive to predominated structures a weighting function in the transform domain Tw is introduced. Each coefficient is compared with a weight Tw(u,v):

$$E[(u,v),D] = |\Delta F[q(u,v),D]| - Tw(u,v), \qquad (18)$$

$$\text{with } \Delta F[q(u,v),D] = F[q(u,v),t] - F[q(u,v),t-1,D)].$$

This weighting function is such that all the components have an equal contribution to a decision criterion. Only positive differences are taken into account, i.e. $|F\{q(u,v)\}| > Tw(u,v)$. The displaced block difference DBD is the minimum over the search area. In the case of the brute force search method the global minimum is defined by:

$$DBD = \min_{SW} \{ \sum_{u=1}^{N} \sum_{v=1}^{N} E[(u,v),D] \}.$$

In the case $|F\{q(u,v)\}| < Tw(u,v)$ the weighting function does not influence the error. Then the results we obtain in the pixel domain and the transform domain are the same. Using this method, manipulations in the transform domain are more easily calculated.

V. SIMULATION RESULTS

In order to compare the performance of the proposed estimation two sequences are used i.e. a splitscreen scene with a hard switch (1) and a sequence containing a girl behaving naturally in front of a camera. The blocksize of the transform and the motion compensation is 8x8 pixels. The bitrate for video only is 300 kbit/s. First a comparison of the estimation is given by using the mean square error as optimization criterion, which of course is visually not the optimal one.

Figure 1 gives the results using the coding configuration with the calculation of the estimation in the pixel domain against the estimation in the transform domain. Four curves are shown: for each method two. The odd numbered ones show the result using the estimation in the pixel domain, the even ones show the result using the new transform domain oriented estimation. Curves 1 and 2 are based on the frame

difference, 3 and 4 are based on the quantized displaced frame difference.

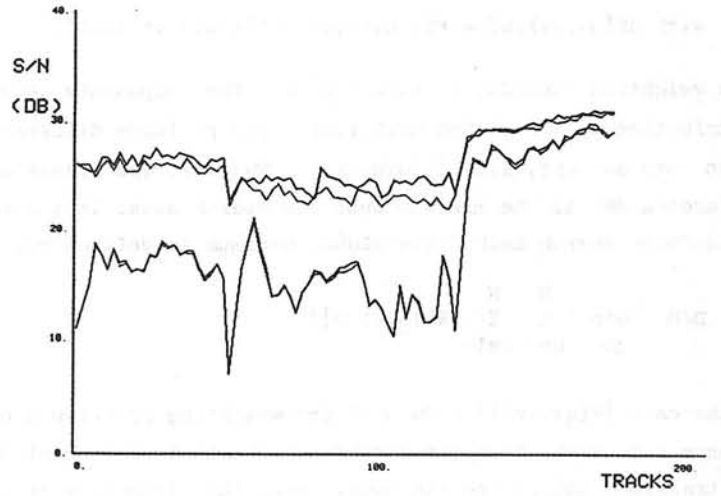


Fig. 1.

ACKNOWLEDGEMENTS

The assistance of A. de Ronde and the support of dr.ir. J. Biemond of Delft University of Technology and ir. G.H.L.M. Heideman of Twente University of Technology and the Dr. Neher laboratories in various aspects of the work reported here is gratefully acknowledged.

REFERENCES

[1] R.H.J.M. Plompen, B.F. Schuurink and J. Biemond, "A New Motion-Compensated Transform Coding Scheme", Proceedings ICASSP 85 International Conference IEEE Acoustics, Speech and Signal Processing, March 1985, Vol. 1, pp. 371-374.

[2] S. Ericsson, "Adaptive Methods for Interframe Picture Coding", Telecommunication Theory Electrical Engineering, Stockholm, Jan. 1984, Ph.D. dissertation.

[3] J.A. Stuller and A.N. Netravali, "Transform Domain Estimation", BSTJ, vol. 58, 3 March 1979, pp. 619-688.

[4] D.F. Elliot and Rao, K.M., Fast Transforms Algorithms Analysis, Application, Academic Press, 1984.

[5] Bruckhart H. and X. Muller, On Variant Sets of Certain Class of Fast Translation Invariant Transforms, IEEE Trans. on Acoustics, Speech and Signal Processing, ASSP-28, no. 5, 1980.

SUB-BAND CODING OF IMAGES USING VECTOR QUANTIZATION

P.H. Westerink*, J.W. Woods** and D.E. Boekee*

In this paper we present a new 2-dimensional sub-band coding technique with particular application to images. We employ a 16 band decomposition where the 16 parallel sub-bands are regarded as a vector, by taking one sample from each sub-band. These 16-dimensional vectors are coded using Vector Quantization (VQ). A comparison will be made between coding each seperate sub-band with DPCM and the new technique proposed here. Some preliminary results show the importance of our approach.

1. INTRODUCTION

Sub-band coding of speech was introduced by Crochiere et al [3] in 1976. Since that time this technique has become quite popular for the medium bandwidth coding of speech [6]. The basic idea of sub-band coding is to split up the frequency band of the signal and then to code each sub-band with either PCM or DPCM using a coder and bit rate accurately matched to the statistics of that particular band. Later contributions on sub-band coding of speech introduce Vector Quantization (VQ), either by taking the parallel sub-bands as a vector [1] or by coding each sub-band seperately with VQ [5].

The extension to multidimensional sub-band filtering was made by Vetterli [11] by considering the case of splitting a multidimensional signal up into sub-bands. However, no coding results were presented in that paper. Results on sub-band coding of images were reported recently by Woods [12], who used adaptive DPCM, and v. Brandt [2], who combined temporal DPCM and conditional replenishment for sub-band coding of videoconference signals.

In this paper we present a form of sub-band coding that makes use of VQ where the vectors consist of samples coming from each sub-band. This can be seen as an extension to 2-D signals from the 1-D case described in [1]. For that purpose 16 equally sized sub-bands will be

---

* Delft University of Technology, Department of Electrical Engineering, Information Theory Group, P.O. Box 5031, 2600 GA Delft, the Netherlands

** Visiting professor from Rensselaer Polytechnic Institute, Electrical, Computer, and Systems Engineering Department, Troy, New York, NY 12180-3590.

split off from the image, using the Quadrature Mirror Filter (QMF) technique. After this the vectors are formed and coded using VQ.

Some preliminary coding results will be presented by comparing this new coding technique for images to coding each seperate sub-band with DPCM. As will be shown VQ gives better results at lower bit rates.

2. SUB-BAND FILTERING

Figure 1 shows the initial four-band partitioning stage that is the basis for the 16-band filter system to be used in our sub-band coding.

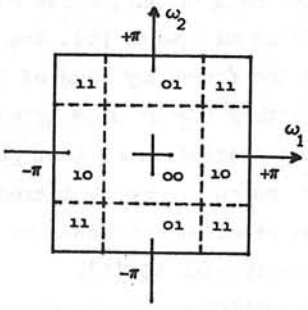


Figure 1. Initial four band partitioning

After each sub-band has been split off, it is demodulated to baseband by a (2x2) downsampling, which will make each sub-band full band at a lower sampling rate (figure 2). For the 16 band system this process is repeated to further split each sub-band into four more sub-bands. The resulting 16 sub-bands will be full band at a sampling rate which is reduced by a factor four in each dimension.

When FIR filters are used to approximate the sub-band characteristics of figure 1, either gaps or aliasing errors will occur due to the effect of downsampling in the transition band of the filter. To compensate for this effect the QMF approach was introduced, first in 1-D sub-band filtering [4] and later for the multidimensional case by Vetterli [11].

Reconstruction by means of 2-D QMF's of the four sub-band system of

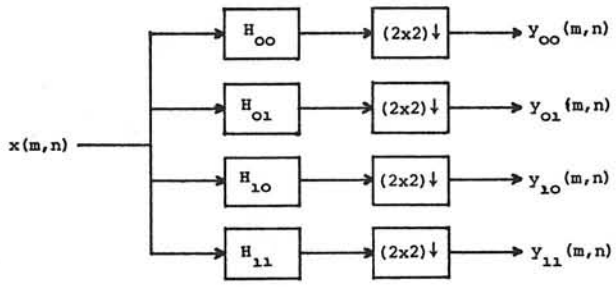


Figure 2. 4 sub-band splitting scheme

figure 2 consists of upsampling each sub-band a factor two in each dimension and filtering using the reconstruction filters

$$F_{ij}(\omega_1,\omega_2) = 4(-1)^{i+j}H_{ij}(\omega_1,\omega_2), \qquad 0 \leq i,j \leq 1, \tag{1}$$

where $H_{ij}(\omega_1,\omega_2)$ are the QMF's that were used to split off the sub-bands. Finally the upsampled and filtered sub-bands are added to obtain the reconstructed image (figure 3).

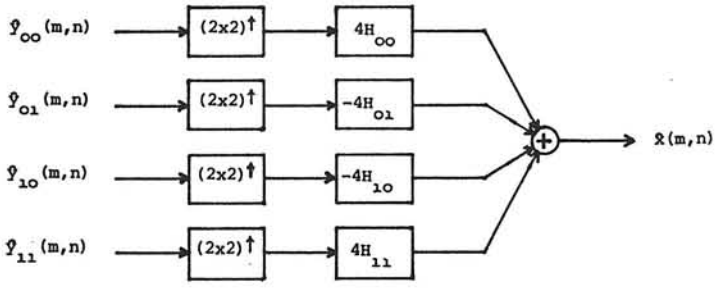


Figure 3. 4 sub-band reconstruction scheme

As Vetterli [11] has shown, the 2-D QMF's can be constructed as a separable product of identical 1-D QMF's

$$H_{ij}(\omega_1,\omega_2) = H_i(\omega_1)\ H_j(\omega_2), \qquad 0 \leq i,j \leq 1. \tag{2}$$

In our coding simulations we used the 1-D 32 point QMF designated as 32D in [7].

3. VECTOR QUANTIZATION

A vector quantizer is a vector generalization of a scalar quantizer (PCM), where a k-dimensional vector consisting of k samples or other parameters of a waveform is quantized as a single entity. The vector is encoded by finding the best matching codevector in a codebook containing 2^m k-dimensional codevectors. A binary word of m bits is transmitted to the receiver, identifying the address of the codevector selected to represent the input vector. The receiver has a copy of the same codebook and generates the output vector by table lookup. The codebook size (number of codevectors) is a critical parameter which determines the encoding complexity needed for searching through the codebook, the memory required to store the codebook in both transmitter and receiver, and the bit rate of the coder.

In this paper, the k-dimensional vectors are formed by taking one sample from each of the sub-bands that were split off from the image as described in the previous paragraph. Because we have 16 sub-bands our vector will be 16-dimensional. The block diagram of the coder system is shown in figure 4.

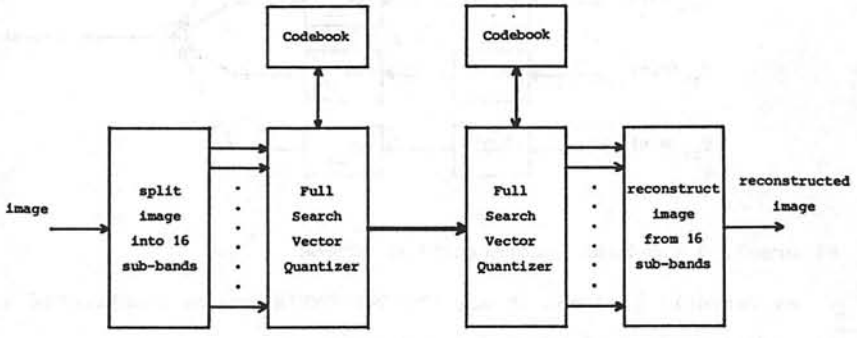


Figure 4. Sub-band coding scheme.

By designing the sub-band coding scheme this way, we attempt to exploit the linear and non-linear correlations among the samples coming from the 16 different frequency bands of the image. This property of VQ is, among others, described in great detail by Makhoul et al [9].

The codebook design, also known as training the codebook, is done using the LBG-algorithm, which is called after Linde, Buzo and Gray [8]. To obtain the initial guess for the LBG-algorithm the splitting-technique is employed, which is usefull when one wishes to design quantizers of successively higher rates until achieving an acceptable level of distortion. For our coding simulations we generated codebooks of sizes 1,2,4,8,...,2048. This enabled us to evaluate the coder behaviour for different codebooksizes and therefore different bitrates.

4. CODING SIMULATION RESULTS

A coding simulation was carried out on a monochrome image of size 256x256 with 8 bit gray levels. Photo 1 shows the coding result when applying DPCM on each seperate sub-band, at a bit rate of 1.0 bits per sample. Photo 2 shows the image that was coded with our sub-band coding method. For the VQ a codebook containing $2^{10}=1024$ codevectors was used, yielding a bit rate of 10/16 = 0.625 bits per sample.

In figure 5 coding simulation results are shown for various values of the bit rate. The dashed line represents the DPCM coding results, the drawn line represents results of the new sub-band coding technique using VQ. The dotted line is taken from Woods [12], (the exact numbers are from [10]), to compare our results to adaptive DPCM on the seperate sub-bands.

5. CONCLUSION

In this paper we proposed a new 2-D sub-band coding technique for images. Taking one sample from each sub-band a 16-dimensional vector is formed, which is coded with Vector Quantization. As preliminary results point out, our new approach allows lower bit rates for the same SNR when compared to DPCM and adaptive DPCM. In contrast to the method where VQ is applied directly to images our method does not introduce

Photo 1: DPCM on each sub-band; 1.0 bits/sample

Photo 2: VQ on sub-bands; 0.625 bits/sample

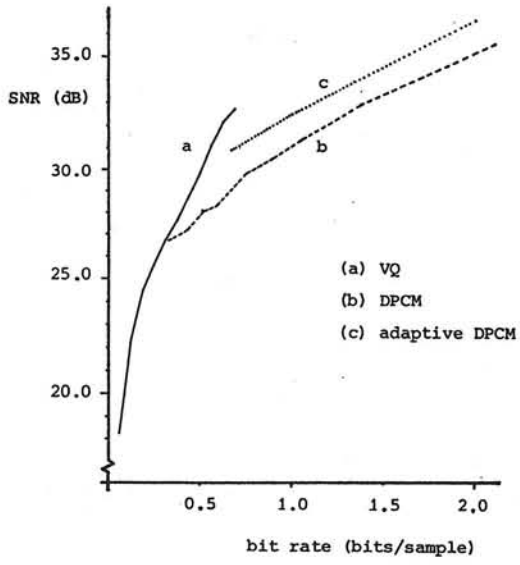


Figure 5. SNR versus bit rate for three methods of sub-band coding.

any blocking effects and errors are therefore less visible for the human observer.

6. REFERENCES

[1] H. Abut and S.A. Luse, "Vector Quantizers for Sub-band Coded Waveforms", Proc. ICASSP, San Diego, CA, March 1984, paper 10.6.

[2] A. v. Brandt, "Sub-band Coding of Videoconference Signals using Quadrature Mirror Filters", Proc. IASTED, Conf. on Applied Signal Proc. and Digital Filtering, Paris, June 1985.

[3] R.E. Crochiere, S.A. Webber and J.L. Flanagan, "Digital Coding of Speech in Sub-bands", Bell System Technical Journal, vol. 55, no. 8, October 1976, pp. 1069-1085.

[4] D. Esteban and C. Galand, "Applications of Quadrature Mirror Filters to Split Band Voice Coding Schemes", Proc. ICASSP, May 1977, pp. 191-195.

[5] A. Gersho, T. Ramstadt and I. Versvik, "Fully Vector Quantized Sub-band Coding with Adaptive Codebook Allocation", Proc. ICASSP, San Diego, CA, March 1984, paper 10.7.

[6] N.S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice-Hall, 1984.

[7] J.D. Johnston, "A Filter Family Designed for Use in Quadrature Mirror Filter Banks", Proc. ICASSP, April 1980, pp. 291-294.

[8] Y. Linde, A. Buzo and R.M. Gray, "An Algorithm for Vector Quantizer Design", IEEE Trans. Comm., vol. CCM-28, January 1980, pp. 84-95.

[9] J. Makhoul, S. Roucos and H. Gish, "Vector Quantization in Speech", Proc. IEEE, vol. 73, no. 11, November 1985, pp. 1551-1588.

[10] S.D. O'Neil, "Sub-band Coding of Images with Adaptive Bit Allocation", MS Thesis, ECSE Dept., R.P.I., Troy, New York, April 1985.

[11] M. Vetterli, "Multi-Dimensional Sub-band Coding: Some Theory and Algorithms", Signal Processing, vol. 6, April 1984, pp. 97-112.

[12] J.W. Woods and S.D. O'Neil, "Sub-band coding of images", to appear in the Proc. ICASSP 86, Tokyo, Japan, April 1986.

An ARMA model identification algorithm

R. Moddemeijer*

To identify from electroencephalogram (EEG) signals the mechanism, which causes the spreading of epileptic seizures in the brain we use a model identification algorithm. We have choosen for autoregressive moving average (ARMA) modelling. We present an off-line maximum likelihood (ML) or least squares algorithm, based on iterative Gauss-Newton minimization. A systematic parameter search is integrated in the inversion of the Hessian-matrix. The optimal model is selected with the Akaike criterion. We are able to identify a model in a large parameter space using only a few active parameters. Finally we present some promising results.

1. INTRODUCTION

An autoregressive moving average (ARMA) model identification and parameter estimation algorithm is presented, which is designed to reveal some aspects of the mechanism which leads to the spreading of epileptic seizures in the brain. Analysing electroencephalogram (EEG) signals we are confronted with a multi-channel and time-dependent system with a tremendous number of parameters.

There is a contradiction between the great flexibility which requires many parameters versus the low variance of estimates which requires a small number of parameters. To deal with this problem we use a large parameter space with only a few active parameters.

It is common practice to select an optimal configuration of parameters and their estimates by minimizing a cost function. Some examples of these functions are final predicting error (FPE) [1], autoregressive transfer function criterion (CAT) [2] and the Akaike information theoretic criterion (AIC) [3]; we only use the latter.

*Technische Hogeschool Twente, Afd. Elektrotechniek
Postbus 217, 7500 AE Enschede

All criteria evaluate the goodness of the fit versus the number of parameters. To overcome the problem of having to evaluate all possible parameters configurations, a systematic non exhaustive parameter search is introduced.

2. THEORY

As an example we restrict ourselves to a one channel stationary ARMA model. We assume the observed discrete time signal x_n ($1 \leq n \leq N$) can be modelled by

$$(2.1) \qquad x_n = \varepsilon_n + \sum_{i=1}^{J} \left(b_i \varepsilon_{n-i} - a_i x_{n-i} + c_i \delta_{n-i} \right) \qquad J = \sup(I,n)$$

The paramaters a_i, b_i and c_i stand for autoregressive (AR), moving average (MA) and initial condition parameters respectively.
The signal ε_n is assumed to be normally distributed stationary white noise with variance σ^2 and the impulse-function is defined by $\delta_n = 0$ if $n \neq 0$ and $\delta_0 = 1$. The maximum model order is given by I.

We estimate the parameters a_i, b_i, c_i and σ^2 by the maximum likelihood (ML) method. This is equivalent to minimization of the sum of squares V as function of the parameters [4]

$$(2.2) \qquad V = \tfrac{1}{2} \sum_{n=1}^{N} \varepsilon_n^2$$

Independent of the other parameters the ML-estimate of σ^2 equals

$$(2.3) \qquad \hat{\sigma}^2 = \frac{2}{N} V$$

For our model the criterion AIC equals

$$(2.4) \qquad AIC = N \ln(2\pi e \hat{\sigma}^2) + 2P$$

with P the number of active parameters. It is a sensible extension to the ML-algorithm to minimize AIC instead of V. The essential problem is how to incorporate changes of the parameter configuration into a minimization algorithm.

We define a parameter vector

$$(2.5) \qquad \theta^T = (a_1, a_2 \ldots a_I,\ b_1, b_2 \ldots b_I,\ c_1, c_2 \ldots c_I)$$

and construct an iterative Gauss-Newton algorithm [5] by approximation of V near $\theta=\theta_0$

$$(2.6) \qquad V(\theta) \approx V(\theta_0) + V_\theta(\theta_0)^T.(\theta-\theta_0) + \tfrac{1}{2}(\theta-\theta_0)^T.V_{\theta\theta}(\theta_0).(\theta-\theta_0)$$

In this formula $V_\theta(\theta_0)$ means the gradient of V with respect to the parameter vector θ in $\theta=\theta_0$:

$$(2.7a) \qquad V_\theta(\theta_0) = \sum_{n=1}^{N} \varepsilon_n(\theta).\nabla\, \varepsilon_n(\theta)\Big|_{\theta=\theta_0}$$

and $V_{\theta\theta}(\theta_0)$ is the Hessian matrix, which we approximate by

$$(2.7b) \qquad V_{\theta\theta}(\theta_0) \approx \sum_{n=1}^{N} (\nabla\, \varepsilon_n(\theta)).(\nabla\, \varepsilon_n(\theta))^T\Big|_{\theta=\theta_0}$$

It is convenient to follow Aaström [4] and calculate the derivatives of ε_n with respect to θ recursively. As example we differentiate (2.1) with respect to a_j and replace n by n+k and j by j+k

$$\frac{\partial \varepsilon_{n+k}}{\partial a_{j+k}} = x_{n-j} \approx \sum_{i=1}^{I} b_i \frac{\partial \varepsilon_{n+k-i}}{\partial a_{j+k}}$$

This recursive relation for the derivatives is invariant for different k, so

$$(2.8) \qquad \frac{\partial \varepsilon_{n+j-1}}{\partial a_j} = \frac{\partial \varepsilon_n}{\partial a_1} = x_{n-1} - \sum_{i=1}^{I} b_i \frac{\partial \varepsilon_{n-i}}{\partial a_1}$$

We rewrite (2.6) as a function of θ instead of $\theta-\theta_0$

$$(2.9) \qquad V(\theta) \approx \tilde{V}(0) + \tilde{V}_\theta(0)^T.\theta + \tfrac{1}{2}\,\theta^T.\tilde{V}_{\theta\theta}(0).\theta$$

The constants $\tilde{V}(0)$, $\tilde{V}_\theta(0)$ and $\tilde{V}_{\theta\theta}(0)$ are calculated by rearranging the righthand part of (2.6) We emphasize that these constants are extrapolations for $\theta=0$ and NOT calculated directly. Using (2.9) we estimate the parameter vector θ at the extreme (minimum) of $V(\theta)$ and than calculate the sum of squares $V(\theta)$ for $\theta=\hat{\theta}$

$$(2.10) \qquad V_\theta(\hat{\theta}) \approx \tilde{V}_\theta(0) + \tilde{V}_{\theta\theta}(0).\hat{\theta} = 0$$

$$(2.11) \qquad V(\hat{\theta}) \approx \tilde{V}(0) + \tfrac{1}{2}\,\tilde{V}_\theta(0)^T.\hat{\theta}$$

Putting $\hat{\theta}=0$ in (2.11) leads to the same result as putting $\theta=0$ in (2.9). It was necessary to transform (2.6) into (2.9) to obtain this particular property. We will make use of this property to incorporate the selection of active parameters into the inversion of the Hessian-matrix, which is the essential step in solving (2.10). It is convenient to combine (2.10) and (2.11) into a matrix equation $Wy=z$

$$(2.12) \qquad \begin{bmatrix} 2\tilde{V}(0) & \tilde{V}_\theta(0)^T \\ \tilde{V}_\theta(0) & \tilde{V}_{\theta\theta}(0) \end{bmatrix} \begin{bmatrix} 1 \\ \hat{\theta} \end{bmatrix} \approx \begin{bmatrix} 2V(\hat{\theta}) \\ V_\theta(\hat{\theta}) \end{bmatrix}$$

We omit the ordering of the elements of θ and split this vector into $\theta^T=(\psi,\phi)$. We only want to minimize V with respect to the active parameters ψ and keep the inactive parameters ϕ equal to zero. This is possible by application of Gauss-Jordan pivots [5,6]. Pivoting the equation $W\,y=z$ using w_{ii} as pivot element results in an equivalent equation $\tilde{W}\,\tilde{y}=\tilde{z}$ in which the elements y_i and z_i are interchanged.

$$(2.13) \qquad \begin{aligned} \tilde{w}_{kl} &= w_{kl} - w_{ki}\,w_{il}/w_{ii} && \text{if } k\neq i \quad l\neq i \\ \tilde{w}_{ki} &= w_{ki}/w_{ii} && \text{if } k\neq i \\ \tilde{w}_{il} &= -w_{il}/w_{ii} && \text{if } l\neq i \\ \tilde{w}_{ii} &= 1/w_{ii} \end{aligned}$$

and

$$(2.14)\qquad \begin{array}{ll} \tilde{y}_k = y_k & \tilde{z}_k = z_k \\ \tilde{y}_i = z_i & \tilde{z}_i = y_i \end{array} \qquad \text{if } k \neq i$$

Pivoting W using in succession all diagonal elements as pivot element is equivalent to matrix inversion. We only pivot the diagonal elements related to ψ and find

$$(2.15)\qquad \tilde{\tilde{W}} \begin{bmatrix} 1 \\ V_\psi(\hat{\psi},\phi) \\ \phi \end{bmatrix} \approx \begin{bmatrix} 2V(\hat{\psi},\phi) \\ \hat{\psi} \\ V_\phi(\hat{\psi},\phi) \end{bmatrix}$$

How to interpret these results? The lefthand side is known, because for active parameters $V_\psi(\hat{\psi},\phi)=0$ and for inactive parameters $\phi=0$. The righthand side provides us with estimates of the active parameters ψ and the sum of squares $V(\hat{\psi},\phi)$. Of course the state of the parameters (active of inactive) can be changed by pivoting.

We make the algorithm recursive by improving the estimate $\hat{\theta}$ using the former estimate of θ as θ_0 in equation (2.6). If the minimization of the sum of squares, without modification of the parameters configuration, is close to convergence, we allow one of these modifications:

a) introduction of one new active parameter

b) reduction of the number of active parameters by one

c) exchange of an active and an inactive parameter

Which modification is optimal? For every modification (one or two subsequent pivots) we predict the sum of squares $V(\hat{\psi},\phi)$. Substitution of these results in (2.4) provides us with predictions of AIC. The most promising modification with respect to AIC is accepted.

Due to the approximations it is necessary to check convergence after every minimalisation step. If an iteration step fails in the original Gauss-Newton algorithm, a parameter estimate $\hat{\theta}'$ on a straight line between θ_0 and $\hat{\theta}$, closer to θ_0 is tried until $V(\hat{\theta}')<V(\theta_0)$ [5]. Due to complications caused by changes of the parameter configuration we can not apply this method. Instead we search for an estimate $\hat{\theta}'$ on

a shrinking set of spheres until $AIC(\hat{\theta}')<AIC(\theta_0)$. This leads to minimization subject to the constraint

$$(2.16) \quad R^2 = (\theta-\theta_0)^T(\theta-\theta_0)$$

We do this using the principle of Lagrangian multipliers, which leads to a slightly modified equation (2.6), where λ is a constant and I the unity matrix:

$$(2.17) \quad V(\theta) \approx V(\theta_0)+V_\theta(\theta_0)^T(\theta-\theta_0)+\tfrac{1}{2}(\theta-\theta_0)^T(V_{\theta\theta}(\theta_0)+\lambda I)\,(\theta-\theta_0)$$

This means that in all equations $V_{\theta\theta}(\theta_0)$ must be replaced by $V_{\theta\theta}(\theta_0)+\lambda I$. For shrinking spheres the constant λ runs from 0 to ∞. This solution resembles the Levenberg-Marquardt procedure [7].

We have given an outline of an algorithm, which is certainly not the final version. We have problems caused by convergence to local minima. The exact formulation is still a point of dicussion.

3. RESULTS

We have investigated the performance of the algorithm by three tests: estimation of the parameters a) of wellknown signals, b) of a simple model and c) of a more complicated model.

We have estimated the parameters of the time-series A, D, E and F of Box & Jenkins [8]. The variance of these time-series is normalized and made equal to one, so $\hat{\sigma}^2$ becomes a measure of fit. The results are given in table 1. In all cases, independent of the criterion we use: minimum variance ($\hat{\sigma}^2$) or minimum AIC, our approach gives slightly better estimates. These improvements are due to the systematic parameters search (E, F) or to a better initialisation (A, D).

We estimated the parameters of a first order AR model using an ARMA model with I=2. Interesting is the number of mistakes (n_m) made by the system identification of 10 sequences of N=100 samples. According to

table 2 the parameter a_1 is underestimated. The statistics are too poor to say anything about the occurence of misidentifications. Interesting is the cause of those mistakes. In table 3 the identified model (I) is compared with the estimate for a First-order AR-model (II). In all cases the identified model is an improvement compared to the AR-model in the sense of the AIC criterion. This means that the algorithm chooses a correct model. Further improvements can only be made by modifying the cost-function.

For the last test we have generated 4th order AR filtered noise (a_1=-2.7607, a_2=3.8106, a_3=-2.6535. a_4=0.9238 [9], N=500). We have estimated the parameters for the correct AR model and for an ARMA model I=5. The resulting parameter configurations are given in table 4. In half of the cases the algorithm provides us with a correct model with respect to AIC. On the other hand in five cases an acceptable model was not found at all. This is mainly caused by local minima to which the algorithm can converge. This indicates a goal for further investigations.

4. DISCUSSION

We have presented our first result, using an other approach to the problem of order determination of ARMA processes. Although our algorithm does not solve problems caused by multiple local minima of the cost-function, it gives mostly better results compared to conventional ARMA estimation methods. The optimum found is dependent of the starting values of the parameters θ_0; this is a known problem in ARMA estimation [10].

The AIC-criterion is in some cases the cause of wrong model selection. We may be able to overcome this problem by choosing another criterion. But in many cases the estimated model and the actual model are indistinguishable within the statistical accuracy, so the criterion is not to be blamed. The algorithm can be shown to be exact for AR-models only, so in this case an iterative solution is not necessary. For these models the estimates are reliable.

We conclude that the idea of ARMA-model estimation using active and inactive parameters seems to be fruitful. The problem of local minima and the initial parameter estimates has to be solved. The performance of the algorithm is slightly better compared to conventional ARMA algorithms. At this moment the algorithm is for our purpose not reliable enough to be used for ARMA model identification.

ACKNOWLEDGEMENT - I am pleased to thank Dr. M.R. Best for some fruitful suggestions.

5. REFERENCES

[1] H. Akaike, Ann.Inst.Stat.Math. 21, 243-247 (1969) and 22, 203-217 (1970)
[2] E. Parzen, An Approach to Time Series Modelling and Forcating Illustrated by Hourly Electricity Demands, State University of New York in Buffalo, Techn.Rpt 37 (1976)
[3] H. Akaike, IEEE Trans. in AC 19, 716-723 (1974)
[4] K.J. Aaström, Automatica 16, 551-574 (1980)
[5] J. Stoer, Einfuhrung in die Numerische Mathematik I, Springer-Verlag, Heidelberg (1972)
[6] R.I. Jennrich & P.F. Sampson, Technometrics 10, 63-72 (1968)
[7] N.K. Gupta & R.K. Mehra, IEEE Trans. on AC 19, 774-783 (1974)
[8] G.E.P. Box & G.M. Jenkins, Time Series Analysis; Forcating and Control (Holdon-Day, San Francisco)
[9] S. Treitel e.a., Topics in numerical analysis (1977)
[10] T. Bohlin, Automatica 7, 199-210 (1971)

TABLE 1

		$\hat{\sigma}^2$	AIC	N	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{b}_1$	$\hat{c}_1$	$\hat{c}_2$	$\hat{c}_3$	$\hat{c}_4$
A	B&J	0.618	468	197	-0.87			-0.48				
		0.605	466	197	-0.90			-0.57		1.36		
B	B&J	0.257	460	310	-0.86							
		0.242	446	310	-0.85				1.88		1.28	
E	B&J	0.209	131	100	-1.32	0.63						
	B&J	0.208	132	100	-1.37	0.74						
		0.140	97	100	-0.83		0.30	0.81	-1.54			0.50
F	B&J	0.823	189	70	0.32	0.18						
		0.750	185	70	0.66			0.37				

The estimates of Box & Jenkins versus ours for I=4

TABLE 2

a_1	Av.$\{\hat{a}_1\}$	$\sigma\{\hat{a}_1\}$	n_m
-0.8	-0.769	0.037	1
-0.6	-0.498	0.078	3
-0.4	-0.342	0.065	1

Parameter estimates for a first order AR model

TABLE 3

a_1		$\hat{a}_1$	$\hat{a}_2$	$\hat{b}_1$	$\hat{b}_2$	$\hat{\sigma}^2$	AIC
-0.8	I	-0.62				0.535	225
	II	-0.68			0.16	0.548	226
-0.6	I	-0.45				0.586	234
	II	-0.62		0.30		0.604	235
-0.6	I	-0.50				0.650	247
	II	-0.56			0.21	0.694	249
-0.6	I	-0.52				0.596	238
	II	-0.59			0.24	0.655	244
-0.4	I	-0.39	0.18			0.865	273
	II	≈0.36				0.840	270

Details of 5 misidentifications

TABLE 4

	AIC-AR	AIC-ARMA	$\hat{a}$ 1 2 3 4 5	$\hat{b}$ 1 2 3 4 5	$\hat{c}$ 1 2 3 4 5
1	-1795	-1800	x x x x .	x	x x x x .
2	-1930	-1935	x x x . x	x . . x .	x x x . x
3	-1740	-1745	x x x x x	. . . . .	x x x x .
4	-1965	-1730	x x x . x	x x x x .	x x x . x
5	-1895	-1780	x x . x x	x x x . x	x x x x x
6	-1975	-1770	x x x . x	x x x x .	x x x . x
7	-1745	-1745	x x x x .	. . . . .	x x x x .
8	-1645	-1475	x x x . x	x x x x .	x x x . .
9	-1970	-1970	x x x x .	. . . . .	x x x x .
10	-2050	-1785	x x x . x	x x x x .	x x x . .

Parameter configurations for a 4th order model

CORRELATIVE LEVEL DECISION FEEDBACK EQUALIZATION

J.W.M. Bergmans*

ABSTRACT - This paper studies the properties of the decision feedback equalizer (DFE) when applied to equalize the residual channel which originates by secluding an a priori selected partial response from a noisy dispersive communications channel. Unlike the conventional linear equalizer, both the optimum filtering which takes place in the forward path of the DFE and the optimally attainable performance are found to be essentially independent of the partial response used.

1. INTRODUCTION

Correlative level transmission techniques (also known as Partial Response techniques) have traditionally been used in conjunction with the linear equalizer [1,2]. They involve the introduction of a controlled amount of intersymbol interference and the detection of a correlated data sequence with an increased number of amplitude levels, from which the original transmitted data sequence can be recovered by means of a deterministic transformation. This approach generally increases the complexity of the system relative to full response signalling, in which all intersymbol interference is eliminated prior to detection, but in return potentially enhances the performance. A good survey of the application area is provided in [3]. In the present paper we extend the correlative level technique to the inherently more powerful decision feedback equalizer

* Philips Research Laboratories, P.O. Box 80.000, 5600 JA Eindhoven, The Netherlands.

[4]. We will study the situation that the DFE attempts to estimate a (usually virtual) data sequence which arises at the output of an a priori selected partial response, which is considered to form a part of the overall channel. The optimum DFE for a given channel subdivision will be found to achieve a performance which is essentially independent of the selected partial response, thereby designating the additional complexity incurred by the introduction of correlative level techniques in the DFE as an unremunerative investment.

2. CHANNEL FACTORIZATION

We set out by considering a discrete-time reception of the form

$$r_k = (a*f)_k + n_k, \tag{1}$$

in which $a_k \epsilon \{-1,1\}$ is an uncorrelated binary data sequence, f_k is the overall sampled impulse response of the channel, "*" denotes linear convolution, and n_k is a zero mean white Gaussian noise sequence having variance N_0. By performing a strong factorization of the sampled channel autocorrelation function the optimum reception problem for any continuous-time PAM system operating over a linear noisy channel can be reformulated in this canonic form ([5], chapter 6). To prevent equalization problems due to spectral zeros from occurring f_k may be factored into a predefined partial response g_k of length L, which contains the "problematic" part of f_k (such as spectral zeros at DC or the Nyquist frequency), and a residual response h_k which can be equalized without severe noise enhancement. This factorization can be denoted as

$$f_k = (g*h)_k. \tag{2}$$

With the aid of (2), (1) can be written as

$$r_k = (b*h)_k + n_k, \tag{3}$$

where

$$b_k \triangleq (a*g)_k \tag{4}$$

is a correlated data sequence taking on one out of at most 2^L (and generally less) values. Figure 1 depicts the model described by (1)-(4).

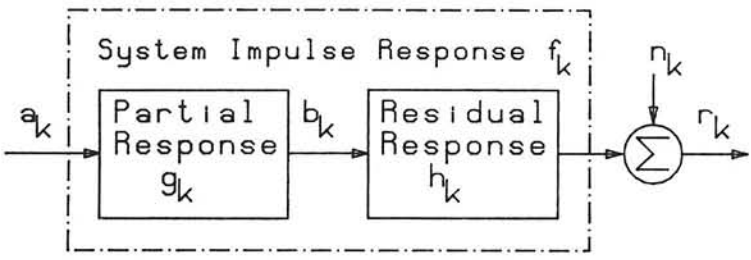


Fig. 1. Distributed discrete-time channel model.

For ease of reference we will in the remainder of this text specify any concrete function g_k in terms of its D-transform g(D), defined as

$$g(D) \triangleq \sum_{i=-oo}^{oo} g_i D^i. \tag{5}$$

In this notation, the most commonly used partial responses have the form

$$g(D) = (1-D)^m(1+D)^n, \quad m,n>=0. \tag{6}$$

The factorization of the overall system impulse response f_k into a partial response g_k and a residual response h_k can be governed by

both performance considerations and engineering convenience (e.g. by the allowable number of data levels in b_k). The choice for g_k is implicitly reflected in the receiver, which produces estimates $\tilde{b}_k$ of the (virtual) data sequence b_k. Whenever g_k represents an invertible operation, an estimate $\tilde{a}_k$ of the original data sequence a_k can be uniquely determined from $\tilde{b}_k$.

3. MMSE DECISION FEEDBACK EQUALIZATION

Apart from a forward filter that suppresses noise and conditions (pre-cursive) intersymbol interference the decision feedback equalizer (see fig. 2) also contains a feedback filter which allows previous decisions to assist in the detection of subsequent digits. The presence of a feedback filter makes the DFE intrinsically more powerful than the linear equalizer.

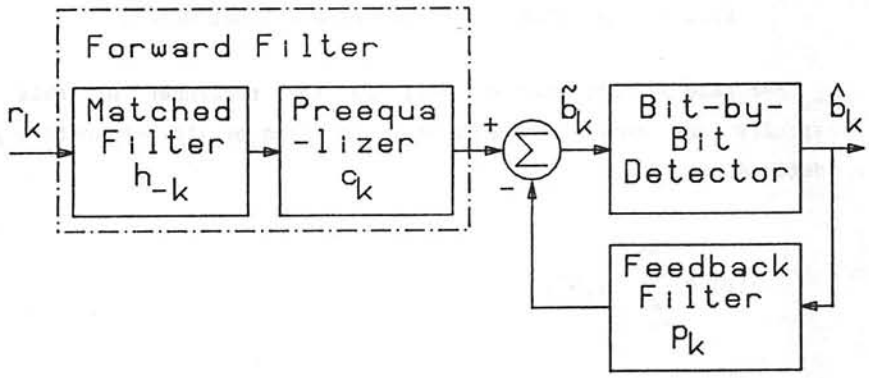


Fig. 2. Decision Feedback Equalizer (DFE).

In order to adhere to common practice we will decompose the forward filter into a filter matched to h_k (having an impulse response $\bar{h}_k \triangleq h_{-k}$) and a preequalizer having an impulse response c_k. Defining the autocorrelation function z_k by

$$z_k = (h*h^-)_k, \tag{7}$$

denoting the feedback filter impulse response by p_k, k>=1, and making the assumption that all relevant previous decisions are correct, we see that the DFE produces a sequence of decision variables $\tilde{b}_k$ given by

$$\tilde{b}_k = (b*(z*c-p))_k + (n*h^-*c)_k. \tag{8}$$

By invoking (4), recalling the assumptions about a_k, and taking signal and noise to be statistically independent, it is easily verified that the mean-square error ϵ of $\tilde{b}_k$, defined as

$$\epsilon \triangleq E[(\tilde{b}_k - b_k)^2], \tag{9}$$

can be expressed as

$$\epsilon = ((z*c-p-\delta)^-*y*(z*c-p-\delta))_0 + N_0(c^-*z*c)_0, \tag{10}$$

where δ_k represents the Kronecker delta function, the superscript "$^-$" again denotes time reversal, and y_k is the autocorrelation function of the partial response g_k, i.e.

$$y_k \triangleq (g^-*g)_k. \tag{11}$$

In order to find the global minimum of ϵ as a function of c_j, -oo<j<oo, and p_j, 1<=j<oo, we first focus upon the optimum setting of the feedback filter coefficients. Differentiating (10) with respect to p_j and requiring all partial derivates to be zero, we see that

$$\frac{\partial \epsilon}{\partial p_j} = -2(y*(z*c-p-\delta))_j = 0, \quad 1<=j<oo. \tag{12}$$

We next differentiate (10) with respect to the forward filter coefficients c_j, -oo<j<oo. This yields:

$$\frac{\partial \epsilon}{\partial c_j} = 2(z^*[y^*(z^*c-p-\delta)+N0c])_j, \text{ all } j. \tag{13}$$

Whenever z_j has no spectral zeros, (13) can only be zero for all j if the term within square brackets equals zero at all instants, i.e.

$$(y^*(z^*c-p-\delta))_j + N0c_j = 0, \text{ all } j. \tag{14}$$

By using (12) we see at once that

$$c_j = 0, \ 1<=j<oo, \tag{15}$$

so that the optimum preequalizer is anticausal. Rearranging terms in (14) and realizing that (15) must hold, we arrive at a set of equations in the variables c_j, $-oo<j<=0$, and p_j, $1<=j<oo$, which can in principle be solved:

$$(c^*(x+N_0\delta))_j = ((p+\delta)^*y)_j, \text{ all } j. \tag{16}$$

In this expression, x_k denotes the autocorrelation function of the channel impulse response f_k, i.e.

$$x_k \triangleq (\bar{f}^*f)_k = (y^*z)_k. \tag{17}$$

It remains to find a simple expression for the minimal mean-square error ϵ_{min}. Combining (14) and (10), and realizing that $\bar{p}_j=c_j=0$, $j>=1$, we find that

$$\epsilon_{min} = N_0c_0. \tag{18}$$

For mathematical convenience we now make the assumption that the partial response g_k is causal and has minimum-phase, so that its convolutional inverse g_k^{-1} is both stable and causal. (Responses within this category that have zeros on the unit circle are

accomodated by moving the zeros a small distance from the unit circle and performing a limiting operation at the end of the derivation.) Moreover we factor the sequence $(x+N_0\delta)_k$ as the convolution of a causal minimum-phase sequence γ_k and its anticausal image γ_k^-, i.e.

$$(x+N_0\delta)_k = (\gamma^- * \gamma)_k, \text{ all } k. \tag{19}$$

According to Doob ([6],pp. 159-161) this factorization is unique, and exists under mild regularity conditions (the most stringent whereof is that the Fourier transform $X(f)+N_0$ of $(x+N_0\delta)_k$ be strictly larger than zero, which is implied by $N_0>0$). The sequence γ_k can be expressed recursively in its k=0 value, which equals

$$\gamma_0 = \exp\{\ 0.5 \int_{-0.5}^{0.5} \ln\,[X(f)+N_0]\ df\ \} \tag{20}$$

The recursion relation is [7]

$$\gamma_k = \frac{2}{k}\sum_{i=0}^{k-1}(k-i)v_{k-i}\gamma_i, \quad k>=1, \tag{21}$$

where the coefficients v_k, k>=1, are defined as

$$v_k \triangleq \int_0^{0.5} \ln[X(f)+N_0]\cos(2\pi kf)df. \tag{22}$$

Convolving both sides of (16) by $(\gamma^{-1} * g^{-,-1})_k$ (where the superscript "$^{-1}$" indicates the convolutional inverse operator) we find that

$$(c*g^{-,-1}*\gamma^-)_k = ((p+\delta)*g*\gamma^{-1})_k, \text{ all } k. \tag{23}$$

The left and right hand sides of this expression are (by construction) causal and anticausal, respectively. Since (23) requires them to be equal, they can only for k=0 be nonzero, and then assume the (right hand side) value g_0/γ_0. (The latter fact can be easily deduced

by expressing $\gamma^{-1}{}_k$ in Z-transform notation and making use of the fact that $\Gamma(Z)$ has no zeros outside the unit circle.) We therefore conclude that

$$c_k = \frac{g_0}{\gamma_0} (g^{-}*\gamma^{-,-1})_k, \text{ all } k, \tag{24}$$

and

$$p_k = \frac{g_0}{\gamma_0} (g^{-1}*\gamma)_k - \delta_k, \text{ all } k. \tag{25}$$

An immediate consequence of (24) and (25) is that every spectral zero of g_k induces an (identically located) zero and pole of c_k and p_k, respectively. This implies in particular that p_k will generally have an infinite extent, so that one decision error degrades the quality of subsequent decisions ad infinitum, which is clearly highly undesirable. Although it is feasible without loss of mean-square performance to reduce these error propagation problems considerably, we shall for the sake of brevity not concern ourselves with this topic here. Using (24), we see that the optimum forward filter impulse response $(h^{-}*c)_k$ is completely determined by $(h^{-}*g^{-})_k=f^{-}{}_k$ and γ_k, neither of which depend upon the partial response g_k. The tailoring of the decision variable b_k required to match a prescribed correlation structure (i.e., partial response) is therefore the exclusive responsibility of the feedback filter. Combining (18), (20) and (24) we finally arrive at the desired closed-form expression for ϵ_{min}:

$$\epsilon_{min} = N_0 \frac{g_0^2}{\gamma_0^2} = g_0^2 \exp\{ \int_{-0.5}^{0.5} \ln[\frac{N_0}{X(f)+N_0}] \, df \}. \tag{26}$$

For partial responses g(D) of the (causal and minimum-phase) form $(1-D)^m(1+D)^n$ it is easily verified that $g_0=1$, so that (26) reduces to the expression describing the optimum mean-square error in the non partial response situation [4].

4. CONCLUSION

This paper has studied the consequences of applying the decision feedback equalizer to cope with the intersymbol interference arising in a given ("residual") part of the channel rather than in the entire channel. Unlike the non partial response case, it was found that the optimum feedback filter coefficients are not a replica of the overall system impulse response, but rather cause a well defined amount of trailing intersymbol interference to arise which allows previous decisions to contribute constructively in the detection of subsequent digits. The optimum DFE performance was shown to be independent of the applied partial response whenever the D-transform of the response is the usual product of (1+D) and (1-D) terms. Thus the significant additional complexity incurred by the application of a partial response of this type is never rewarded in terms of a (mean-square) performance gain. Going by the significant improvements that are in specific cases feasible for the linear equalizer [1] one would a priori not have anticipated such a stingent statement to hold.

5. REFERENCES

[1] A. LENDER, Correlative Level Coding for Binary-Data Transmission, IEEE Spectrum, Vol. 3 (Feb. 1966), pp. 104-115.

[2] P. KABAL, S. PASUPATHY, Partial Response Signalling, IEEE Trans. Commun., Vol. COM-23 (1975), pp. 921-934.

[3] H. KOBAYASHI, A Survey of Coding Schemes for Transmission and Recording of Digital Data, IEEE Trans. Commun. Technol., Vol. COM-19 (1971), pp. 368-375.

[4] C. BELFIORE, J. PARK, Decision Feedback Equalization, Proceedings of the IEEE, Vol. 67 (1979), pp. 1143-1156.

[5] J. PROAKIS, Digital Communications, McGraw Hill, 1983.

[6] J. DOOB, Stochastic Processes, Wiley, New York, May 1967.

[7] D. MESSERSCHMITT, A Geometric Theory of Intersymbol Interference, Part II, BSTJ, Vol. 52 (1973), pp. 1521-1539.

EVENT SERIES PROCESSING: A SIGNAL ANALYSIS APPROACH

O. Rompelman

Event series with a small coefficient of variation may be treated as a time signal consisting of δ-functions. This approach yields simple algorithms for filtering, spectral analysis and correlational analysis.

INTRODUCTION

Many processes in nature can be described in terms of a series of repetitively occurring and identical events. The relevant information is contained in the way these events occur in time. A point process is a mathematical model for this kind of processes. A detection/estimation procedure transforms physical or physiological events into time instants. If the process behaves in an unpredictable way it may be described as a stochastic point process, the random variable being e.g. the number of events within a time bin. When the event intervals have a low coefficient of variation and exhibit relatively slow variations a simpler approach is to be favoured. An example of such a process is the cardiac event series (heart beats).

Let n(t) be the number of events in (0,t]. We may write n(t) as a function of time

$$n(t) = \sum_{\forall_i} u(t-t_i). \qquad (1)$$

Differentiating n(t) with respect to time yields x(t), a signal description of the event process:

$$x(t) = \sum_{\forall_i} \delta(t-t_i). \qquad (2)$$

O. Rompelman is with the Delft University of Technology, Department of Electrical Engineering, Information Theory Group, P.O. Box 5031, 2600 GA Delft, the Netherlands

This is in fact the differential counting process as discussed in [1]. In practice when dealing with digital signal processing we have to convert the continuous time series into a discrete time series. This conversion is called regularization [2] and is equivalent to the sampling procedure in conventional signal processing. The discrete time event series can be written as

$$\tilde{x}(k\theta) = \sum_{\forall_i} \delta[(k-k_i).\theta]. \tag{3}$$

Note that in fact the δ in (2) is a Dirac δ whereas in (3) the Kronecker δ is meant.
On the basis of (2) and (3) it is possible to carry out linear operations on the signal in a much simplified way since integral operations on a series of δ functions will become simple summations.

FILTERING

Assume that the impulse response of a linear filter is $h(t)$. The filtered continuous time event process as defined in (2) is then found from

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} x(t-\tau)\ h(\tau)\,d\tau \\ &= \sum_{\forall_i} h(t-t_i). \end{aligned} \tag{4}$$

Similarly the filtered discrete time event process is found from

$$\tilde{y}(k\theta_o) = \sum_{\forall_i} h[(k\theta_o - n_i\theta)] \tag{5}$$

with θ_o the discretisation interval of the output signal. Introducing the output/input sampling rate ratio

$$\eta = \frac{\theta_o}{\theta} \tag{6}$$

(6) can be rewritten as

$$\tilde{y}(\eta k\Theta) = \sum_{\forall_i} h[(\eta k - n_i).\Theta]. \qquad (7)$$

Though it seems obvious to make $\eta=1$ this will not be a good choice in practice. As an example we discuss a low pass filter. Assume that $\Theta=1$ ms and the cut off frequency of the filter is 0.5 Hz, these values being realistic for HRV*. The cut off frequency of the filter leads to a minimal necessary value for the output sampling frequency $1/\Theta_o$ = 1 Hz. Since the filter in practice is non-ideal, an output sampling frequency of 2 Hz is chosen. This implies that Θ_o = 0.5 s yielding $\eta=500$. It can be concluded that a significant data reduction can be obtained by this procedure which motivates a hard ware implementation [3].

SPECTRAL ANALYSIS

The Fourier transform of the continuous time event process x(t) (again as defined in (2)) yields the complex spectrum $\bar{X}(f)$:

$$\bar{X}(f) = \int_{-\infty}^{\infty} x(t).e^{-j2\pi ft}dt$$

$$= \sum_{\forall_i} e^{-j2\pi ft_i}. \qquad (8)$$

The complex spectrum of the discrete time process is given by

$$\tilde{\bar{X}}(m\Delta f) = \sum_{\forall_i} e^{-j2\pi mn_i\Theta\Delta f}. \qquad (9)$$

For a finite segment of duration T a preferable value of Δf is

$$\Delta f = \frac{1}{T} \qquad (10)$$

---

* Heart Rate Variability

hence

$$\tilde{X}(m) = \sum_{\forall_i} e^{-j\frac{2\pi m n_i}{N}} . \qquad (11)$$

An implementation of an HRV spectrum analyser on a personal computer, based on the principles discussed above has been reported before [2]. An attempt has been made to apply AR-spectral analysis techniques to the event series. According to the Burg method (viz. the extrapolation of the data) this has been achieved by extrapolating the series of intervals. The extrapolated intervals were reconverted to an event series, whereafter the spectrum is calculated as discussed above [4]. The results seem promising for certain applications where low frequency components in relatively short data segments have to be detected. As an example we may refer to the assessment of the so-called 10 sec rhythm in patients suffering from autonomic neuropathy [5].

REFERENCES

[1] D.R. Cox, P.A.W. Lewis, *The Statistical Analysis of Series of Events*, Methuen & Co., Ltd., London, 1968.

[2] O. Rompelman, J.B.I.M. Snijders, C.J. van Spronsen, The measurement of heart rate variability spectra with the help of a personal computer, IEEE Trans. on Biomed.Eng., vol. BME-29, pp. 503-510, 1982.

[3] A.J.R.M. Coenen, O. Rompelman, R.I. Kitney, Measurement of heart rate variability: Part II - Hardware digital device for the assessment of heart rate variability, Med. & Biol. Eng. & Comp., vol. 15, pp. 423-430, 1977.

[4] L.A. Vervoort, Parametrische spectrum analyse voor hartritmevariabiliteit, MSc-Thesis, Inform.Th. Lab., Delft Univ. of Techn., 1986.

[5] O. Rompelman, Parametric spectral analysis of heart rate variability in autonomic neuropathy, Biomed. Technik, Band 29, 139-140, 1984.

OPTIMAL DETECTION OF THE RAPID-EYE-MOVEMENT BRAIN STATE

B. Kemp*

Abstract. A model has been proposed for the stochastic occurrence of bursts of rapid eye movements (REMs) during sleep. REM-bursts are simulated by a Poisson counting process with a rate that depends on a binary Markov 'sleep state'. The corresponding maximum likelihood detector, that continuously monitors the current sleep state based on the observed REM-bursts, has been derived.

1 INTRODUCTION

The various stages of human sleep can be recognized by (a.o.) different eye or body movements, muscle tension and several components of the electroencephalogram. One of the former are the rapid eye movements (REMs): fast rotations of the eyes which occur irregularly, but almost exclusively during wakefulness and during one of the sleep stages that is consequently called REM-sleep. Most systems for automatic sleep stage monitoring are, therefore, partly based on monitoring the 'REM-state' of the brain; that is the state (either wakefulness or REM-sleep) during which REMs do occur.

Because both the brain state and the REMs during the REM state seem to be stochastic processes, an appropriate smoother should take into account their statistical properties. In this paper, such a smoother has been developed by modeling those processes and deriving the likelihood ratio for the problem 'state REM or not'

*. Academic Hospital, Department of Clinical Neurophysiology, Rijnsburgerweg 10, NL 2333AA Leiden, The Netherlands.

2 PROBLEM FORMULATION: A MODEL

We assume sleep stages to be generated by a continuous time Markov chain process. This process has already been proposed as a sleep stage generator model by Zung et al. (1965). Its simulations show a good resemblance to real sleep stage patterns (Kemp and Kamphuisen, 1986).

In the present paper we are interested only in the transitions between REM states (many REMs) and NREM states (few or no REMs). We have therefore simplified the Markov chain to a binary one, $p(t)\epsilon\{0,1\}$, and REMs are generated predominantly when $p(t)=1$. The sleep state generating mechanism shows clear periodicities and trends that are different for different individuals (Kemp and Kamphuisen, 1986). Since one generally does not avail of sufficiently reliable a priori information about these individual dynamics, we have further simplified the binary Markov chain to one with constant transition rates, i.e. a homogeneous one. The corresponding homogeneous brain state generating differential equation reads (Kemp et al., 1985):

$$\begin{aligned} dp(\theta) &= [1-p(\theta)]dq_0(\theta) - p(\theta)dq_1(\theta) \\ &= ([1-p(\theta)]/\tau_0 - p(\theta)/\tau_1)d\theta \\ &\quad + [1-p(\theta)][dq_0(\theta) - d\theta/\tau_0] - p(\theta)[dq_1 - d\theta/\tau_1] \\ &= ([1-p(\theta)]/\tau_0 - p(\theta)/\tau_1)d\theta + dm_1(\theta) \end{aligned} \qquad (1)$$

$$p(0)\epsilon\{0,1\}$$

where $q_1(\theta)$ and $q_0(\theta)$ are mutually independent Poisson counting processes. These processes are constant (i.e. $dq_i(\theta)=0$) except for positive unit counts (i.e. $dq_i(\theta)=1$) that occur with rates $1/\tau_1$ and $1/\tau_0$, respectively. Consequently, $m_1(\theta)$ according to (1) is a martingale. The average sojourn times in the REM state ($p(\theta)=1$) and the NREM state ($p(\theta)=0$) are τ_1 and τ_0 (about 20 min. and 60 min.), respectively.

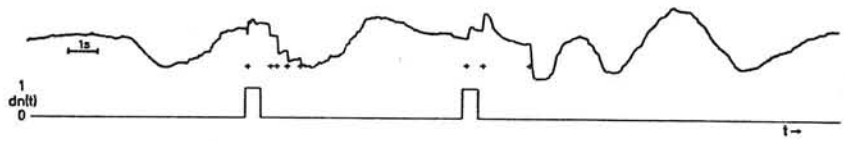


figure 1: Eye movement recording during REM-sleep. Note the two bursts, dn(t)=1, of rapid eye movements (+).

Because the rather long times between REM-bursts within the REM state (fig.1) are critical for REM state monitoring, they will form the basis for our model. The inter-burst times are approximately exponentially distributed. Therefore we have adopted the Poisson counting process, $n(\theta)$, that is suggested by these distributions, as a reasonable first approximation of the REM-burst statistics. Each count, $dn(\theta)=1$, generates a REM-burst (fig.1) and the intercount intervals are exponentially distributed. As in (1) we can write the REM-burst counting process in the form of a martingale-driven differential equation:

$$\begin{aligned} dn(\theta) &= r(\theta)d\theta + dn(\theta) - r(\theta)d\theta \\ &= r(\theta)d\theta + dm_2(\theta) \end{aligned} \qquad (2)$$

$$n(0)=0$$

where $r(\theta)$ is the rate of the Poisson process, $n(\theta)$, i.e. the density of the REM-bursts. Consequently, $m_2(\theta)$ is a martingale. During REM states ($p(\theta)=1$) this rate equals r_1 (about 1/min). During NREM states ($p(\theta)=0$), the rate is partly determined by false positive REM detections and equals r_0 (about .02/min). Or equivalently:

$$r(\theta) = r_0 + (r_1 - r_0)p(\theta) \qquad (3)$$

Equation (1) describes the generation of the brain state,

while (3) and (2) describe the related generation of the observed REM-bursts, $dn(\theta)$. Fig.2 shows a simulation by this model. Based on this description, the problem can be reformulated as follows: find the monitor which provides at all times, t, during the recording interval $(t, 0 \leq t < T)$, the optimal decision, $\hat{p}(t)$, on the current value of $p(t)$, based on the whole night observation, $N(0,T)=(dn(\theta), 0 \leq \theta < T)$, of REM bursts, $dn(\theta)$. As an optimization rule, we have adopted a Bayes criterion: the monitor should minimize the expected false-decision rate.

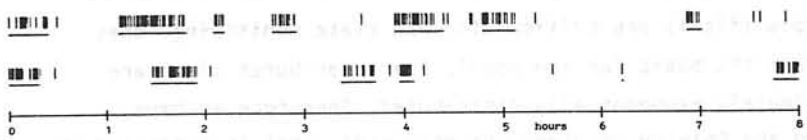


figure 2: Simulating the model during 2 'nights' of 8 hours. | || are REM-bursts. ___:'brain state' is 'REM', when interrupted 'NREM'. Parameters: τ_1=20 min., τ_0=60 min., r_1=1/min., r_0=.02/min.

3 DERIVATION OF THE REM STATE MONITOR

The optimal decision, $\hat{p}(t)$, on $p(t)$ can be obtained from the set of observations, $N(0,T)$, by comparing the likelihood ratio:

$$L(t)=f(N(0,T)|p(t)=1)/f(N(0,T)|p(t)=0) \qquad (4)$$

where $f(.|.)$ are conditional probability densities, to a constant threshold, K. K depends on the optimality criterion. We adopted the Bayes' threshold for minimizing the expected false-decision rate, which reads:

$$K=\tau_0/\tau_1 \qquad (5)$$

When $L(t)>K$ the optimal decision is 'brain state REM', i.e. $\hat{p}(t)=1$. When $L(t)<K$ it is 'brain state NREM', i.e. $\hat{p}(t)=0$. Using the Markov property of $p(\theta)$ and the mutual independency of the Poisson counts, $dn(\theta)$, we may split $L(t)$ into a 'future observations' part and a 'past observations' part as follows:

$$L(t)=(L_-(t)).(L_+(t)) \qquad (6)$$

where (from Bayes' rule applied to (4)):

$$L_i(t)=(\tau_0/\tau_1)\beta_i(t)/[1-\beta_i(t)] \qquad i\in(-,+) \qquad (7)$$

where the expectations, based on 'past' and 'future' observations are:

$$\beta_-(t)=E(p(t)\,|\,dn(\theta),0\leq\theta<t)$$
$$\beta_+(t)=E(p(t)\,|\,dn(\theta),t\leq\theta<T) \qquad (8)$$

In this paper we will concentrate on the 'past observations' likelihood ratio, $L_-(t)$, that is a function of $\beta_-(t)$ according to (7). A differential equation to obtain $\beta_-(t)$ has been derived by Van Schuppen (1977, theorem 4.2). It reads:

$$d\beta_-(\theta)=([1-\beta_-(\theta)]/\tau_0-\beta_-(\theta)/\tau_1)d\theta \qquad (9)$$
$$+\frac{(r_1-r_0)\beta_-(\theta)[1-\beta_-(\theta)]}{r_0+(r_1-r_0)\beta_-(\theta)}(dn(\theta)-[r_0+(r_1-r_0)\beta_-(\theta)]d\theta)$$

Driving (9) from $\theta=0$ to $\theta=t$ yields $\beta_-(t)$. The initial condition, $\beta(0)$, depends on experimental conditions. For instance, if we are rather sure that the monitor is started during wakefulness, $\beta(0)\simeq 1$. Equations (9) and (7) are sufficient for the recursive computation of $L_-(t)$. However, the number of required multiplications can be reduced and the algorithm can be interpreted more clearly by the application of a logarithmic transformation. The transformed variable equals:

$$\lambda(t)=\ln(L(t))=\ln(L_-(t))+\ln(L_+(t))=\lambda_-(t)+\lambda_+(t) \qquad (10)$$

where, according to (7):

$$\lambda_-(t)=\ln((\tau_0/\tau_1)\hat{p}_-(t)/[1-\hat{p}_-(t)]) \qquad (11)$$

The differential equation for the computation of $\lambda_-(t)$ can be obtained from (11) and (9). Because (9) is driven by discontinuous counts, the Itô/Doléans-Dade/Meyer differentiation rule (Van Schuppen, 1977) must be applied as follows:

$$d\lambda_-(\theta)=[\delta\lambda_-(\theta)/\delta\hat{p}_-(\theta)].\omega(d\hat{p}_-(\theta))+\Delta(\lambda_-(\theta)).dn(\theta) \qquad (12)$$

where $\delta./\delta.$ denotes a partial derivative, $\omega(d\hat{p}_-(\theta))$ denotes the continuous part (i.e. $dn(\theta)=0$) of $d\hat{p}_-(\theta)$ in (9) and $\Delta(\lambda_-(\theta))$ denotes the discontinuous jump in $\lambda_-(\theta)$ that is caused by a count, $dn(\theta)=1$. According to (11), this jump equals:

$$\Delta(\lambda_-(\theta))=\ln((\tau_0/\tau_1)[\hat{p}_-(\theta)+\Delta(\hat{p}_-(\theta))]/[1-\hat{p}_-(\theta)-\Delta(\hat{p}_-(\theta))]) \\ -\ln[(\tau_0/\tau_1)\hat{p}_-(\theta)/(1-\hat{p}_-(\theta))] \qquad (13)$$

where $\Delta(\hat{p}_-(\theta))$ denotes the jump in $\hat{p}_-(\theta)$ that is caused by a count, $dn(\theta)=1$. According to (9), this jump equals:

$$\Delta(\hat{p}_-(\theta))=(r_1-r_0)\hat{p}_-(\theta)[1-\hat{p}_-(\theta)]/(r_0+(r_1-r_0)\hat{p}_-(\theta)) \qquad (14)$$

Substitution of (14) in (13) shows that:

$$\Delta(\lambda_-(\theta))=\ln(r_1/r_0) \qquad (15)$$

The partial derivative in (12) can be obtained from (11):

$$\delta\lambda_-(\theta)/\delta\hat{p}_-(\theta)=1/(\hat{p}_-(\theta)[1-\hat{p}_-(\theta)]) \qquad (16)$$

Substitution of (15), (16) and the continuous part, $\omega(d\hat{p}_-(\theta))$, of (9) in (12) yields the differential equation for $\lambda_-(t)$:

$$d\lambda_-(\theta)=([e^{-\lambda_-(\theta)}-1]/\tau_1-[e^{\lambda_-(\theta)}-1]/\tau_0)d\theta$$
$$+\ln(r_1/r_0)\cdot dn(\theta)-(r_1-r_0)d\theta \qquad (17)$$

Driving (17) from $\theta=0$ to $\theta=t$ yields $\lambda_-(t)$. The initial condition, $\lambda_-(0)$ is a function (11) of $\beta_-(0)$, which depends on the state of the subject when the monitor is started.

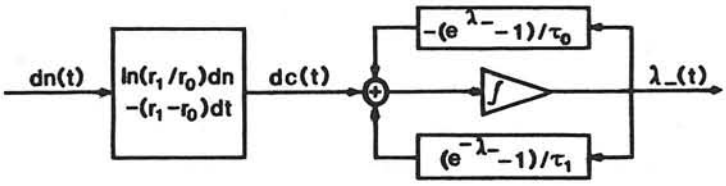


figure 3: Block diagram of the exponential feedback integrator that generates the test statistic, $\lambda_-(t)$, if driven by the observed REM-bursts, $dn(t)$.

Figure 3 shows a block diagram of the algorithm. It appears to be an integrator with exponential feedback that is driven by:

$$dc(\theta)=\ln(r_1/r_0)\cdot dn(\theta)-(r_1-r_0)d\theta \qquad (18)$$

The interpretation of this driving term is simple. Assume $r_1>r_0$. Between REM-bursts, i.e. $dn(\theta)=0$, the integrator will be driven to negative values while REM-bursts kick it to more positive values. According to (2) and (3), the conditional expectation:

$$E\{dn(\theta)\,|\,p(\theta)\}=(r_0+(r_1-r_0)p(\theta))d\theta \qquad (19)$$

which implies that the conditional expectation of the driving term equals:

$$E\{dc(\theta)\,|\,p(\theta)\}=(\{r_0+(r_1-r_0)p(\theta)\}\ln(r_1/r_0)-(r_1-r_0))d\theta$$
$$=r_0((r_1/r_0)^{p(\theta)}\ln(r_1/r_0)-r_1/r_0+1)d\theta \qquad (20)$$

Apparently, the integrator is, on average, driven to positive values during REM-states (i.e. $p(\theta)=1$) and to negative values during NREM-states (i.e. $p(\theta)=0$). The saturation effect that is

caused by the exponential feedback limits the time needed to react to a transition of $p(\theta)$. Quite satisfactory, this effect increases with increasing transition rates.

4 DISCUSSION

For practical sleep stage scoring we conclude that using mathematical models original and attractive solutions may be created for the processing of discontinuous observations. In particular, the exponential feedback integrator might prove to be a simple and effective alternative to the usually applied integration over segements of sleep recordings.

Acknowledgements

We gratefully acknowledge the constructive critisism and help of Dr. J.H. Van Schuppen of the Centre for Mathematics and Computer Science in Amsterdam, Prof. Ir. E.W. Gröneveld of the Twente University of Technology and Prof. Dr. F.H. Lopes da Silva of the University of Amsterdam.

References

B. KEMP, P. JASPERS, J.M. FRANZEN, A.J.M.W. JANSSEN, An optimal monitor of the electroencephalographic sigma sleep state. Biol. Cybern. 51 (1985) 263-270

B. KEMP, H.A.C. KAMPHUISEN, Simulation of human hypnograms using a Markov chain model. Sleep (1986) in press

J.H. VAN SCHUPPEN, Filtering, prediction and smoothing for counting process observations, a martingale approach. SIAM J. Appl. Math. 32 (1977) 552-570

F. SCHILLER, Historical note on sleep and eye movements. Sleep 7 (1984) 199-201

W.W.K. ZUNG, T.H. NAYLOR, D. GIANTURCO, W.P. WILSON, A Markov chain model of sleep EEG patterns. Electroencephalogr. Clin. Neurophsyiol. 19 (1965) 105

CODING FOR THE BINARY SWITCHING MULTIPLE ACCESS CHANNEL

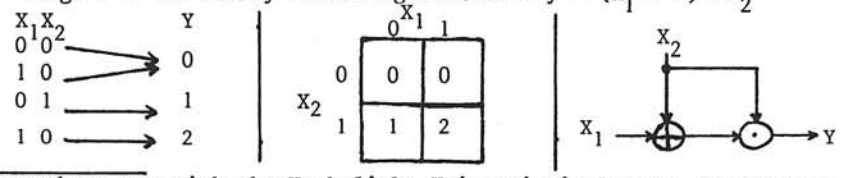
P. Vanroose* and E.C. van der Meulen**

Within the class of all deterministic binary two-user multiple-access channels we consider a recently introduced channel [1], namely the binary switching MAC. The problem of constructing uniquely decodable code pairs is considered. Three classes of interesting codes are developed, based on the use of an MDS code. The highest rate sum found is 1.33333, which is well above the time sharing line.

1. INTRODUCTION

Consider all deterministic binary 2-user multiple-access channels (MAC's). The only non-trivial ones are those with 2 or 3 channel output symbols. In the first case there are essentially two non-equivalent channels, namely the EOR channel [2,3] and the OR channel [2,4]. Both have as capacity region $\{(R_1,R_2) \in [0,1]^2 | R_1 + R_2 \leq 1\}$. In the second case there are also two channels, the well-studied binary adder channel (e.g. [5]) with capacity region $\{(R_1,R_2) \in [0,1]^2 | R_1 + R_2 \leq 1.5\}$, and an asymmetric channel, shown in Fig. 1, introduced by J. Vinck [1], which we will call the *binary switching channel* (BS-MAC) because "user X_2 switches the connection between X_1 and Y on and off" [1]. For this channel we give coding strategies, leading to some good code rate pairs above time sharing, such as (2/3, 2/3), (0.5, 0.7925) and (1/3, 0.9358).

Fig. 1 : The binary switching channel : $y = (x_1 + 1) \cdot x_2$



The authors are with the Katholieke Universiteit Leuven, Department of Mathematics, Celestijnenlaan 200 B, B-3030 Heverlee, Belgium.

* Supported by the Belgian Program for the reinforcement of the scientific potential in the new technologies - PREST/KUL/01 (Prime Minister's Office for Science Policy).

** Partially supported by Project GOA 83/88-53, Ministerie van Wetenschapsbeleid, Brussels, Belgium.

2. THE CAPACITY REGION OF A DETERMINISTIC MAC WITH TWO INPUT USERS

Suppose the input alphabets are $\mathcal{X}_1 = \{0,\ldots,M_1\}$ and $\mathcal{X}_2 = \{0,\ldots,M_2\}$ and the output alphabet is $\mathcal{Y} = \{0,\ldots,N\}$. If the (deterministic) channel is defined by $y = f(x_1,x_2)$, call

$$I_{jk} := \{i \in \mathcal{X}_1 \,|\, f(i,j) = k\}, \quad J_{ik} := \{j \in \mathcal{X}_2 \,|\, f(i,j) = k\} \text{ and}$$

$$K_k := \{(i,j) \in \mathcal{X}_1 \times \mathcal{X}_2 \,|\, f(i,j) = k\} = f^{-1}(k) .$$

For each product input distribution $(\{p_0,\ldots,p_{M_1}\},\{q_0,\ldots,q_{M_2}\})$ on $\mathcal{X}_1 \times \mathcal{X}_2$, call

$$P_{jk} := \sum_{i \in I_{jk}} p_i \quad , \quad Q_{ik} := \sum_{j \in J_{ik}} q_j \quad \text{and}$$

$$R_k := \sum_{(i,j) \in K_k} p_i q_j = \sum_{i \in \mathcal{X}_1} p_i Q_{ik} = \sum_{j \in \mathcal{X}_2} q_j P_{jk} .$$

Then the capacity region of this channel is [6,7] the convex hull of the union over all input distributions of

$$\{(R_1,R_2) \,|\, 0 \leq R_1 \leq \sum_{j \in \mathcal{X}_2} q_j H(\mathcal{P}_j) \quad , \tag{2.1}$$

$$0 \leq R_2 \leq \sum_{i \in \mathcal{X}_1} p_i H(\mathcal{Q}_i) \quad , \tag{2.2}$$

$$R_1 + R_2 \leq H(\mathcal{R}) \qquad \} \tag{2.3}$$

where $\mathcal{P}_j := (P_{j0},\ldots,P_{jN})$, $\mathcal{Q}_i := (Q_{i0},\ldots,Q_{iN})$, $\mathcal{R} := (R_0,\ldots,R_N)$ and $H(.)$ is the $N+1$-ary entropy function.

Theorem 1 : If for every $k \in \mathcal{Y}$ there exist $A \subseteq \mathcal{X}_1$ and $B \subseteq \mathcal{X}_2$ such that $K_k = A \times B$ then condition (2.3) vanishes.

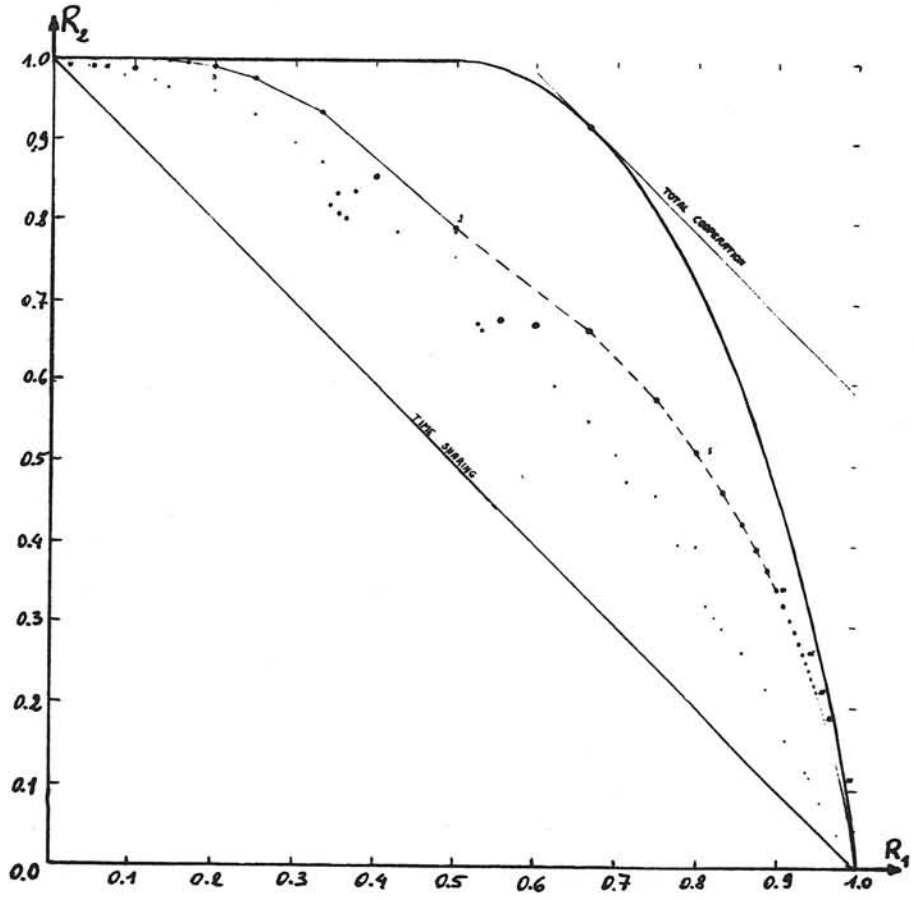
Theorem 2 : If for every $k \in \mathcal{Y}$ the set J_{ik} does not depend on i then condition (2.2) becomes $0 \leq R_2 \leq H(Q_i)$.

Consider now the binary switching channel, where $\mathcal{X}_1 = \mathcal{X}_2 = \{0,1\}$, $\mathcal{Y} = \{0,1,2\}$ and $y = (x_1 + 1).x_2$. This channel satisfies Theorems 1 and 2. Consequently its capacity region is the convex union over all $p,q \in [0,1]$ of $\{(R_1,R_2) \,|\, 0 \leq R_1 \leq q.h(p),\ 0 \leq R_2 \leq h(q)\}$. Because R_2 does not depend on p, we can take $p = 0.5$, which gives the explicit form

$$\{(R_1,R_2) \,|\, 0 \leq R_1 \leq 0.5, 0 \leq R_2 \leq 1\} \cup \{(R_1,R_2) \,|\, 0.5 \leq R_1 \leq 1, 0 \leq R_2 \leq h(R_1)\}. \tag{2.4}$$

This region is sketched in Fig. 2. Notice that the total cooperation line $R_1 + R_2 = \log_2 3$ touches this region in $(2/3,h(1/3))$. Thus in this respect this channel can do better than the binary adder channel.

Fig. 2 :
The capacity region of the BS-MAC
with some achievable rate points.



3. CODING FOR THE BINARY SWITCHING CHANNEL

The code pair (C_1,C_2) with $C_1,C_2 \subseteq \{0,1\}^n$ is *uniquely decodable* (UD) for the binary switching channel if for every $c_1,c_1' \in C_1, c_2 \in C_2$ the following implication holds :

$$c_1 \wedge c_2 = c_1' \wedge c_2 \longrightarrow c_1 = c_1' \ , \tag{3.1}$$

where $\wedge$ denotes componentwise minimum (or product).

Remark that the code pairs $(\{0,1\},\{1\})$ and $(\{0\},\{0,1\})$ are UD, attaining the rate points (1,0) and (0,1) respectively in the capacity region, so only codes above time sharing are interesting. Note also that the points (r,1) with $0 < r \leq 0.5$ are *not* achievable with UD code pairs, because then C_2 would contain the all 0 vector, which always gives the all 0 vector as channel output.

A first class of codes for this channel, of block length n, are obtained by taking $C_1 := \{00\ldots0,11\ldots1\}$ and $C_2 := \{0,1\}^n \backslash \{00\ldots0\}$. This pair is UD since, if $c_1 \wedge c_2 = c_1' \wedge c_2$, then at least one bit of c_1 equals one bit of c_1', so $c_1 = c_1'$. With these codes we obtain rate pairs

$$R_1 = \frac{1}{n} \quad , \quad R_2 = \frac{1}{n} \log_2(2^n - 1) \ . \tag{3.2}$$

A second class of interesting codes consists of the code pairs (C_1,C_2) of block length n with $C_1 := \{c_1 \in \{0,1\}^n | w(c_1) \text{ is even}\}$ and $C_2 := \{c_2 \in \{0,1\}^n | w(c_2) \geq n-1\}$, where w(.) is the Hamming weight. This pair is UD since, if $c_1 \wedge c_2 = c_1' \wedge c_2$, then $c_1 = c_1'$ because c_2 has at most one 0 and the erasure of an arbitrary bit of c_1 does not affect its information content. This fact is typical for an MDS (*maximum distance separable*) code with minimum distance d = 2. The other code C_2 may then contain every vector of weight $\geq$ n-1. Using these codes we obtain rate pairs

$$R_1 = 1 - \frac{1}{n} \quad , \quad R_2 = \frac{1}{n} \log_2(n+1) \ . \tag{3.3}$$

For several values of n the rate pairs (3.2) and (3.3) are given in Table 1. They are also shown in Fig. 2, lying on the solid and the dashed lines respectively. The code pair with highest rate sum is $(\{000,011,101,110\},\{011,101,110,111\})$, with $R_1 = R_2 = 2/3$. Table 2 is the decoding table of this code.

Table 1 :

	First class of uniquely decodable code pairs.			Second class of uniquely decodable code pairs.		
n	R1	R2	R1+R2	R1	R2	R1+R2
1	1.00000	0.00000	1.00000	0.00000	1.00000	1.00000
2	0.50000	0.79248	1.29248	0.50000	0.79248	1.29248
3	0.33333	0.93578	1.26912	0.66667	0.66667	1.33333
4	0.25000	0.97672	1.22672	0.75000	0.58048	1.33048
5	0.20000	0.99084	1.19084	0.80000	0.51699	1.31699
6	0.16667	0.99621	1.16288	0.83333	0.46789	1.30123
7	0.14286	0.99838	1.14124	0.85714	0.42857	1.28571
8	0.12500	0.99929	1.12429	0.87500	0.39624	1.27124
9	0.11111	0.99969	1.11080	0.88889	0.36910	1.25799
10	0.10000	0.99986	1.09986	0.90000	0.34594	1.24594
15	0.06667	1.00000	1.06666	0.93333	0.26667	1.20000
20	0.05000	1.00000	1.05000	0.95000	0.21962	1.16962
25	0.04000	1.00000	1.04000	0.96000	0.18802	1.14802
50	0.02000	1.00000	1.02000	0.98000	0.11345	1.09345

Table 2 : Decoding table of a uniquely decodable code pair.

C1 \ C2	011	101	110	111
000	011	101	110	111
011	022	102	120	122
101	012	202	210	212
110	021	201	220	221

Table 3 : Some rate pairs for the third class of uniquely decodable code pairs.

ε	m	b	n	R1	R2	R1+R2
1	2	2	10	0.60000	0.67279	1.27279
1	2	3	10	0.40000	0.85546	1.25546
1	3	4	27	0.55556	0.67948	1.23503
0	3	5	24	0.37500	0.83782	1.21282
1	4	8	68	0.52941	0.67504	1.20445
-1	4	7	60	0.53333	0.66793	1.20126
1	4	11	68	0.35294	0.83422	1.18716
1	5	21	165	0.36364	0.80314	1.16677
-1	5	20	155	0.35484	0.80967	1.16451
0	5	21	160	0.34375	0.81915	1.16290

Let us now generalize this idea of C_1 as MDS code. Let $q := 2^m$. Take C_1 a $[q-1,q-b-1,b+1]$ *Reed-Solomon* code over GF(q), in some binary representation; or a $[q,q-b,b+1]$ extended RS code over GF(q); or a $[q+1,q-b+1,b+1]$ MDS code ([8], Ch. 11). Summarizing, we take C_1 a $[q+\varepsilon,q-b+\varepsilon,b+1]$ MDS code over GF(q) in binary representation, where $\varepsilon \in \{-1,0,1\}$ and $b \in \{1,2,\ldots,q+\varepsilon-1\}$. Take C_2 the set of all vectors in $(GF(q))^{q+\varepsilon}$ at a distance $\leq b$ from the all 1 vector. This is a nonlinear $(q+\varepsilon, V_q(q+\varepsilon,b))$ code over GF(q) in binary representation, where $V_q(n,r) := \sum_{i=0}^{r} \binom{n}{i}(q-1)^i$ is the volume of a sphere of radius b ([9], p. 55). With these codes we attain rate pairs

$$R_1 = 1 - \frac{b}{q+\varepsilon} \quad , \quad R_2 = \frac{1}{m(q+\varepsilon)} \log_2 V_q(q+\varepsilon,b) \; . \tag{3.4}$$

The binary block length is $n = m(q+\varepsilon) = m(2^m + \varepsilon)$.
The only MDS codes known with other parameters are $[q+2,q-1,4]$ and $[q+2,3,q]$ codes over GF(q), so we can also take $\varepsilon=2$ and $b \in \{3,q-1\}$. Table 3 gives rate pairs of this class of codes for some values of ε,m and b. The code pair with highest rate sum is attained for $\varepsilon = 2$, $m = 2$, $b = 3$, namely $R_1 + R_2 = 1.2866$. Several good code rate pairs achievable with this procedure are shown in Fig. 2. They are all subobtimal with respect to the codes of class 1 and class 2, which can be explained by the fact that these codes are not really binary while the channel is binary. Although, the code pairs of Table 3 are optimal in the sense that they can only be improved by time-sharing some codes of classes 1 and 2.

4. CONCLUSION

In this article we described good achievable code pairs for the binary switching channel, based on MDS codes which combat arbitrary erasures. So far no codes had been constructed for this channel, which forms an interesting counterpart to the extensively studied binary adder channel within the class of binary deterministic MAC's. It is quite possible that based on other methods still better code pairs then given here can be derived.

Generalisations of this channel such as the noisy BS-MAC yet have to be investigated, as well as the problem of finding good

δ-decodable code pairs for this channel.

The feedback capacity region of this channel is known, since it belongs to the class introduced by F. Willems [10]. However J. Vinck showed that in this case the feedback capacity region coincides with the non feedback region given in Fig. 2.

5. REFERENCES

[1] A.J. Vinck, "On the multiple access channel", *Proc. 2nd Joint Swedish-Soviet Int. Workshop on Inform. Theory*, Gränna, 1985, 24-29.

[2] E.C. van der Meulen, "The discrete memoryless channel with two senders and one receiver", *Proc. 2nd Int. Symp. on Inform. Theory*, Tsahkadsor, 1971, 103-135.

[3] P.G. Farrell, "Survey of channel coding for multi-user systems", in : *New Concepts in Multi-User Communication* (ed. J.K. Skwirzynski), NATO Adv. Study Inst. 43, Sijthoff & Noordhoff, Alphen a/d Rijn, 1981, 133-159.

[4] L. Györfi and I. Kerekes, "A block code for noiseless asynchronous multiple access OR channel", *IEEE Trans. Inform. Theory*, IT-27, 1981, 788-791.

[5] P.A.B.M. Coebergh van den Braak and H.C.A. van Tilborg, "A family of good uniquely decodable code pairs for the two-access binary adder channel", *IEEE Trans. Inform. Theory*, IT-31, 1985, 3-9.

[6] R. Ahlswede, "Multi-way communication channels", *Proc. 2nd Int. Symp. on Inform. Theory*, Tsahkadsor, 1971, 23-52.

[7] H. Liao, "A coding theorem for multiple access communications", Ph.D. Diss., "Multiple Access Channels", Univ. Hawaii, 1972.

[8] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland Publ. Comp., Amsterdam, 1977.

[9] J.H. van Lint, *Introduction to Coding Theory*, Grad. Texts in Mathematics 86, Springer, New York, 1982.

[10] F.M.J. Willems, "The feedback capacity region of a class of discrete memoryless multiple access channels", *IEEE Trans. Inform. Theory*, IT-28, 1982, 93-95.

ON MINIMUM BREAKDOWN DEGRADATION IN BINARY MULTIPLE DESCRIPTIONS

J.C.C.M. Remijn*

In this paper we give more evidence for the fact that problems in source coding for multiple descriptions in the no excess rate case are easier to solve than in the case when excess rate occurs. A converse is proved for the binary 2-descriptions problem. Determining the minimum breakdown degradation is shown to be reducible to solving the corresponding zero-error problem. It is demonstrated that this does not work for problems with excess rate.

INTRODUCTION

The problem of source coding for multiple descriptions, formulated in 1980, appears to be a problem for which the known methods in source coding are not strong enough. Therefore many authors have studied special cases, in order to get a better understanding of the problem. We will restrict ourselves to the binary case and start with a formulation of the problem. When the output of a binary symmetric information source is of great importance, it is advisable to spread the information over more than one channel, in order to protect the decoder against a complete loss of information. We assume that the channels are noiseless, but sometimes a few channels break down completely. The encoder does not know anything about the state of the channels. When two channels are used the communication system is depicted in Figure 1. The encoder sends its messages at rates R_1 resp. R_2 over channels 1 resp. 2. If both channels are working, the decoder reproduces the source output within an average distortion (error frequency) d_0; if channel t breaks down the decoder reproduces the source output within average distortion d_t, for t = 1 or 2. The problem is to determine the quintuple (R_1,R_2,d_0,d_1,d_2) of achievable (in the usual Shannon sense)

---

* Eindhoven University of Technology,
Department of Mathematics and Computing Science,
P.O. Box 513, 5600 MB Eindhoven, The Netherlands.

rates and distortions, often referred to as the achievable (rate-distortion) region. When we impose the restriction $R_1 + R_2 = 1 - h(d_0)$ (where $h(x) = -x \log x - (1-x)\log(1-x)$), i.e. the sumrate equals the Shannon rate-distortion function, we speak of a coding system without excess rate. If $R_1 + R_2 > 1 - h(d_0)$, excess rate is present.

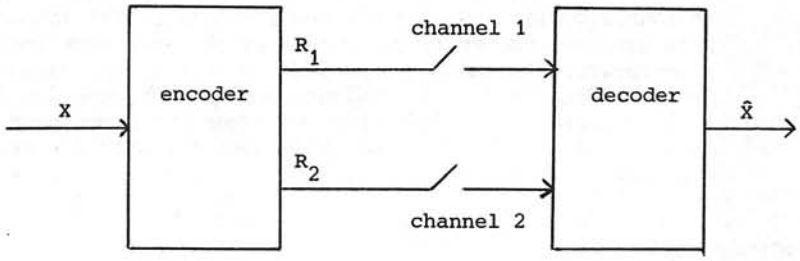


Figure 1. The 2-descriptions communication system.

In the no excess rate case the problem is completely solved for general sources and distortion measures [1]. Despite this result, it remains worthwhile to study the binary case, in order to obtain more knowledge of the excess rate problem. Zhang and Berger [2] used the binary case as example in showing that the achievable region of Cover and El Gamal [3] is not the complete region in the excess rate case. We simplify the k-descriptions problem by letting the distortions d_i only depend on the number i of channels that are received. So the problem is now to determine the achievable 2k-tuples $(R_1,\ldots,R_k; d_1,\ldots,d_k)$. Furthermore we define, for all $i = 1,2,\ldots,k-1$, the minimum breakdown degradation $d_i^{(k)} := \inf\{d_i \mid (\frac{1}{k},\ldots,\frac{1}{k}; d_1,\ldots,d_i,\ldots,0)$ is achievable$\}$, i.e. the minimum achievable distortion, if only i of the k channels are working.

In [4] and [5] H. Witsenhausen determined lower bounds for $d_1^{(k)}$ and $d_{k-1}^{(k)}$, under the strong restriction that the decoder reproduces the source without error if all channels are working. These distortions are in general not achievable. We will show that in the Shannon set-up these zero-error lower bounds in fact are achievable. Furthermore, the lower bounding technique also works in this case. We illustrate this

by determining $d_1^{(2)}$ in this way, obtaining an easier proof than Zhang and Berger [2], who used properties of Hamming spheres. Finally we show that this method does not work in the excess rate case.

THEOREMS AND PROOFS

The first rather standard result gives an inner bound for the achievable region for the k-descriptions problem in the case $d_k = 0$.

THEOREM 1: $(R_1,\ldots,R_k ; d_1,\ldots,d_{k-1},0)$ is achievable if there exist mutually independent random variables $Y_1,Y_2,\ldots,Y_k$, jointly distributed with source variable X, such that:

$$\sum_{i=1}^{k} R_i = 1 ,$$

$$\sum_{j=1}^{s} R_{i_j} \geqslant I(X ; Y_{i_1},\ldots,Y_{i_s}) , \quad \text{for all } s \in \{1,2,\ldots,k-1\} \text{ and for every s-subset } \{i_1,\ldots,i_s\} \text{ of } \{1,2,\ldots,k\} ,$$

and there exist decoding functions ϕ_s for all $s \in \{1,2,\ldots,k\}$, such that $E\, d(X,\phi_s(Y_{i_1},\ldots,Y_{i_s})) \leqslant d_s$, for every s-subset $\{i_1,\ldots,i_s\}$ of $\{1,2,\ldots,k\}$.

The proof, which we omit, can be given using properties of typical sequences [6]. We remark that the distortions, determined by Witsenhausen in the zero-error case for $d_1^{(k)}$ and $d_{k-1}^{(k)}$ are achievable. In particular we state the following corollary, which we will prove as a preparation for our converse result:

COROLLARY: $d_1^{(2)} \leqslant \frac{1}{\sqrt{2}} - \frac{1}{2}$, this is the smallest possible distortion in Theorem 1.

Proof: Without loss of generality the best thing a decoder can do in case of a breakdown is to "trust" the received symbol. If we look at the joint distribution of Y_1 and Y_2 in Figure 2 and recall that the

marginal distributions of Y_1 and Y_2 are the same, since our problem is symmetric, we readily see that we have to avoid the cases that $X = 0$ while $Y_1 = Y_2 = 1$, and $X = 1$ while $Y_1 = Y_2 = 0$. The distortion $d_1 = a(1-a)$ is minimized if a is maximal, i.e. if $a^2 = \frac{1}{2}$. We conclude in this case: $d_1^{(2)} \leqslant \frac{1}{\sqrt{2}} - \frac{1}{2}$. □

In the same way we can prove the following

LEMMA: In the zero-error case we have $d_1^{(2)} \geqslant \frac{1}{\sqrt{2}} - \frac{1}{2}$.

Proof: Without loss of generality we can assume that the source bits are taken in blocks of length 2n. We write all possible binary 2n-tuples in an $2^n \times 2^n$ array. For every row (resp. column) a codeword is made (see Figure 3). If the source has produced 2n bits the corresponding codewords, say c_i and d_j, are transmitted over channels 1 and 2. Note that indeed the decoder achieves errorfree reconstruction if both channels are working. In every coordinate there are 2^{2n-1} ones and 2^{2n-1} zeros. Analogous to the proof of the above corollary we conclude that in case of a breakdown the error frequency is at least $\frac{1}{\sqrt{2}} - \frac{1}{2}$ per coordinate. □

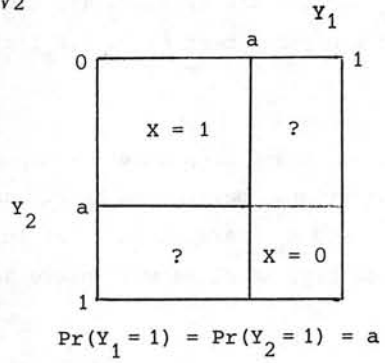


Figure 2. The joint distribution of Y_1 and Y_2.

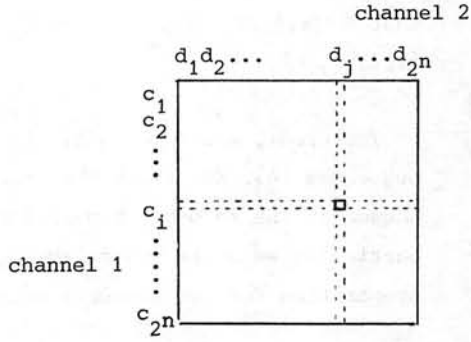


Figure 3. The array containing all 2^{2n}-tuples.

THEOREM 2: $d_1^{(2)} = \frac{1}{\sqrt{2}} - \frac{1}{2}$.

The proof is in essence the same as the proof of the above lemma. Given the achievable quintuple $(\frac{1}{2}, \frac{1}{2}, 0, d_1^{(2)}, d_1^{(2)})$, we can for some wordlength N construct again an array, in which almost all binary N-tuples are written. Due to the Shannon formulation there are a little more rows and columns. An important feature is that we may assume that the "empty" entries are randomly distributed over the array. If we analyze this carefully, we conclude that again we find that $d_1^{(2)} \geq \frac{1}{\sqrt{2}} - \frac{1}{2}$. We leave the technical details.

The reason that this method works in the no excess rate case is that we can assume that only a very small fraction of the entries in the array is empty, and that those entries are equally distributed. Suppose we have $R_1 = R_2 = \frac{1}{2} + r$ ($r > 0$, fixed) and the decoder reproduces the source without error if both channels are working. By writing all binary N-tuples in an $2^{(\frac{1}{2}+r)N} \times 2^{(\frac{1}{2}+r)N}$ array, we can arrange the empty entries in such a way that the error frequency tends to zero as N becomes larger. So we will not find useful lower bounds in this way for the excess rate case.

CONCLUSION

We have outlined a converse proof for the determination of the minimum breakdown degradation for the binary 2-descriptions problem without excess rate, parallelling the proof of the zero-error case. The minimum breakdown degradations $d_1^{(k)}$ and $d_{k-1}^{(k)}$ can now be determined using Witsenhausens results ([4], [5]). The problem of finding $d_i^{(k)}$ for all $i \in \{1,2,\ldots,k-1\}$ is reduced to the corresponding zero-error optimization problem.

REFERENCES

[1] R. AHLSWEDE, The rate-distortion region for multiple descriptions without excess rate, IEEE Trans. Info. Theory 31 (1985) 721-726.

[2] Z. ZHANG & T. BERGER, New results in binary multiple descriptions, preprint, August 1984.

[3] T. COVER & A. EL GAMAL, Achievable rates for multiple descriptions, IEEE Trans. Info. Theory 28 (1982) 851-857.

[4] H.S. WITSENHAUSEN, On team guessing with independent information, Math. Oper. Res. 6 (1981) 293-304.

[5] H.S. WITSENHAUSEN, Team guessing with lacunary information, Math. Oper. Res. 8 (1983) 110-121.

[6] T. BERGER, Multiterminal source coding, in *The Information Theory Approach to Communications*, G. Longo ed., CISM Courses and Lectures No. 229, Springer Verlag, Wien-New York, 1977, pp. 171-231.

KEY SIGNATURE SCHEMES

C.J.A. Jansen*

ABSTRACT: For the purpose of automatic key selection or key identification a bitstring of certain length is often derived from the secret key at hand. This bitstring which we call key signature, is usually derived from the key by means of a well known one-way function like the DES. In this paper we have considered simple functions for obtaining key signatures and their security. As a figure of merit for the security of key signature schemes we have introduced the notion of given-away information of the key. For random permutations on the key the given-away information per keybit decreases monotonicaly with increasing keylength. A random selection of keybits gives away very little information on the average.

1. INTRODUCTION

In order to identify a secret key, which is assumed to be a randomly chosen bitstring of certain length, one often derives another bitstring from this secret key by means of a one-way function. As an example this bitstring, called key signature, may be the result of encrypting a fixed message with a cryptographic algorithm that uses this key, like the Data Encryption Standard [1].

However for practical reasons it may be undesirable to use a complex cryptographic algorithm. Therefore simple key signature schemes need to be considered, e.g. a pseudo random selection of keybits.

The interesting knowledge about key signature schemes is of course the amount of information the signature reveals about the key itself.

* Philips Usfa B.V., P.O. Box 218
5600 MD Eindhoven, The Netherlands

As the key itself is secret but its signature is not, this amount of information should be as small as possible. In the sequel this amount of information revealed about the key is called given-away information ε. It is obvious that a linear deterministic scheme such as a parity check scheme is unsuitable as in principle it reveals as much bits as there are parity check bits.

How many bits the key signature should comprise is the subject of section 2. Section 3 establishes the amount of given-away information if a random permutation of the key bits is taken as a signature. Moreover an upper- and a lowerbound to the given-away information are derived. In sections 4 and 5 the given-away information is determined for a random extraction and a random selection respectively. Finally in section 6 a practical pseudo-random key signature scheme is shown.

2. KEY SIGNATURE LENGTH

In order to determine the length of a key signature, assume that signatures are bitstrings with every bit drawn independantly at random and the probabilities of a 1 and a 0 are equal. Suppose there are N signatures each of m bits length. We are interested in the probability that they are all different. This is the well known birthday problem. The probability of unique signatures can be determined as follows:

$$P_{unique} = \frac{M}{M} \cdot \frac{M-1}{M} \cdot \frac{M-2}{M} \cdot \ldots\ldots \frac{M-(N-1)}{M}$$

with $M = 2^m$

$$P_{unique} = 1 \cdot (1-\frac{1}{M}) \cdot (1-\frac{2}{M}) \ldots (1-\frac{(N-1)}{M})$$

If $\frac{N-1}{M} << 1$ then:

$$(0) \qquad P_{unique} \doteq 1 - \frac{1}{2} N(N-1)\, 2^{-m}$$

If for example N=100 and m=16, there is a probability of 7.6 % that one or more signatures will not be unique, and if for example we demand a probability of non-uniqueness of 10^{-6} for N=100, the signature length will be 34 bits.

3. RANDOM PERMUTATIONS

In the sequel let K denote the secret key, S its signature, W(K) the Hamming weight of K, and L(K) the length of K. Suppose that S is a random permutation of K, then one only has to try all permutations of S in order to guess K.
This number of permutations equals: $\binom{n}{k} < 2^n$, with n = L(K) and k = W(K).
Therefore the amount of given away information of K is:

$$(1) \qquad e_{n,k} := n - \log\binom{n}{k} \qquad \text{bits}$$

In order to determine the average given away information per key, note that the distribution of the weight of a key is binomial as in (2), so that the average given away info is given by (3).

$$(2) \qquad Pr\{W(K)=k\} = \binom{n}{k}\, 2^{-n}$$

$$(3) \qquad \bar{\varepsilon}_n = n - 2^{-n} \sum_{k=0}^{n} \binom{n}{k} \log\binom{n}{k} \qquad \text{bits}$$

As can easily be verified expression (3) shows that the average given away info is equal to the entropy of a binomial source with an alphabet of n letters and probability given by (2).

One can readily obtain upper and lower bounds for expression (3). A lower bound is obtained by replacing the rightmost binomial coefficient in (3) :

$$\binom{n}{k} \leqslant \binom{n}{\frac{1}{2}n}.$$

An upperbound follows immediately from the fact that the entropy of a memoryless source with any probability distribution over the output alphabet is always less than that of a source with a homogeneous probability distribution. So we have:

$$(4) \qquad \bar{\varepsilon}_n \geqslant n - \log\binom{n}{\frac{1}{2}n}$$

$$(5) \qquad \bar{\varepsilon}_n \leqslant \log(n+1)$$

Figure 1 shows the average given away information per keybit as a function of the length of the key. It also shows the normalized upper- and lowerbounds. It is clearly demonstrated that for a random permutation the amount of given away info per keybit goes to zero with increasing keylength. For example a randomly permuted DES key (56 bits) gives away 3.95 bits about the key itself or either 0.07 bits per keybit.

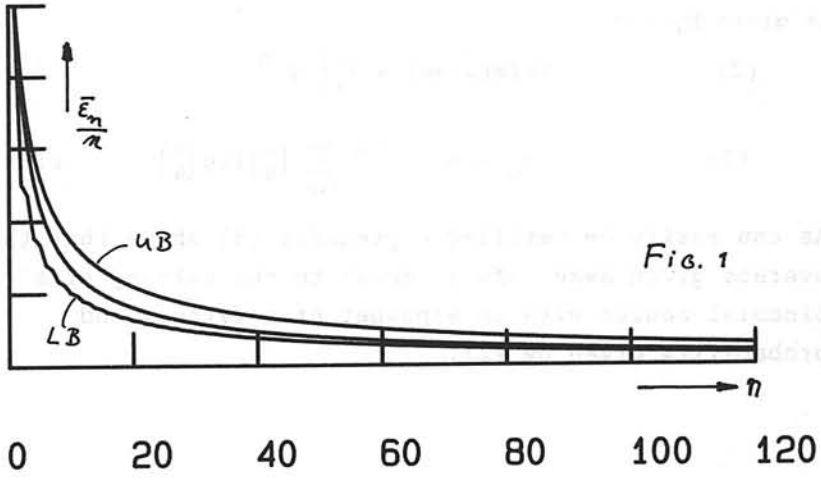


Fig. 1

4. RANDOM EXTRACTIONS

In this section we consider a random extraction of a number of keybits as a signature, so if a keybit has been selected, it cannot be selected again. This means that all signature bits represent distinct keybits. The signature length can therefore be at most equal to the keylength and if this is the case, we have a random permutation and are back at section 3. However a random extraction is more realistic than a random permutation, because most keysignatures used, have less bits than the key itself. In order to guess the key K one now has to try all bitstrings of length L(K) that have weight at least W(S) and contain at least L(S)-W(S) zeroes. Let N denote this number of trials, n=L(K), l=L(S) and w=W(S), then we have:

$$N = \sum_{i=w}^{n-(l-w)} \binom{n}{i} \quad \text{with } l \leqslant n \tag{6}$$

The given away info about the key K amounts to:

$$\varepsilon_{n,l,w} = n - \log N \quad \text{bits} \tag{7}$$

For random keys the weight distribution of S will also be binomial, analogous to (2). Therefore the average given away information of K is given by expression (8).

$$\bar{\varepsilon}_{n,l} = n - 2^{-l} \sum_{w=0}^{l} \binom{l}{w} \log \left[\sum_{i=w}^{n-l+w} \binom{n}{i} \right] \tag{8}$$

Substituting l=n in (8) yields expression (3), as can easily be verified. Figure 2 shows the behaviour of (8) for n=56. It also shows how little information is given away if the signature length is only a few bits shorter than the keylength. For instance a random extraction of 48 bits out of the 56, gives away only 0.7646 bits of info about the key on the average.

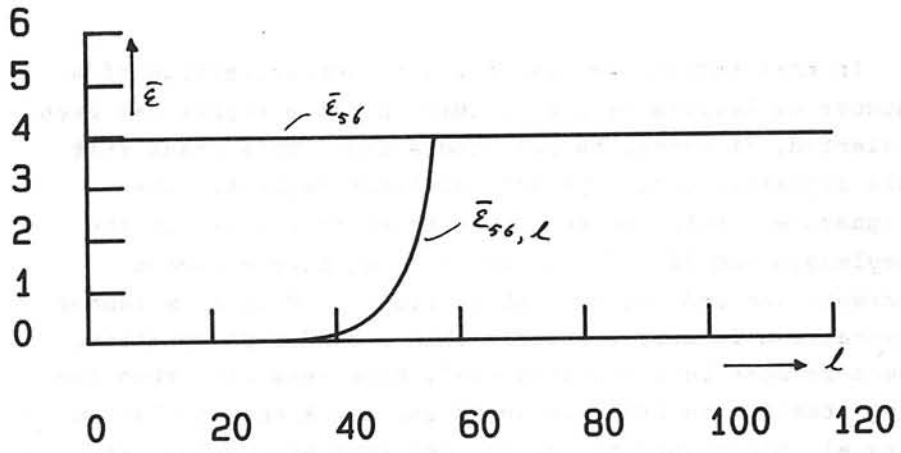


Fig. 2

5. RANDOM SELECTIONS

A random selection of keybits is taken as a key signature in this section. So in fact we throw l times with an n-sided dice and collect the l addressed bits to form the keysignature. In this case there is only a certain probability that all the signature bits represent different keybits, and obviously one or more keybits may be selected more than once. Observe that as the signature becomes longer an increasingly more acurate estimate of the weight of K can be made. If the signature comprises infinitely many bits, W(K) is known and we are back at section 3 with regard to the amount of given away info about key K. For a finite length signature suppose there are m different keybits. If one knows the weight of these m bits, we are back at section 4, where expression (6) gives the number ofguesses to make for the key. As one only knows W(S), more guesses will be necessary with strings that have minimum weight W(S) - (L(S)-m) and maximum weight L(K) - (m-W(S). It therefore makes sense to define as an upperbound to the given away info for a random selection :

$$(9) \qquad \bar{\varepsilon}'_{n,l} \leq \sum_{m=1}^{l} \Pr(m \text{ different} | l,n)\ \bar{\varepsilon}_{n,m}$$

Where $\bar{\varepsilon}_{n,m}$ is given by (8) with l=m and Pr(m different|l,n) is the probability of obtaining exactly m different outcomes when throwing l times with a n-sided (unbiased) dice. With the aid of a good textbook on combinatorial analysis, e.g. [2], one can find the following expression for this probability distribution :

$$(10) \qquad \Pr(m \text{ diff.} | l,n) = m!\ n^{-l} \binom{n}{m} \mathcal{S}_l^{(m)}$$

Where $\mathcal{S}_l^{(m)}$ is a Sterling number of the second kind, which satisfies the following recursion relation :

$$(11) \qquad \mathcal{S}_{n+1}^{(m)} = m\ \mathcal{S}_n^{(m)} + \mathcal{S}_n^{(m-1)} \qquad n \geq m \geq 1$$

Expression (9) is evaluated for n=16 and depicted in figure 3. Also for n=56 and l=100 the average given away information is less than 0.6523 bits

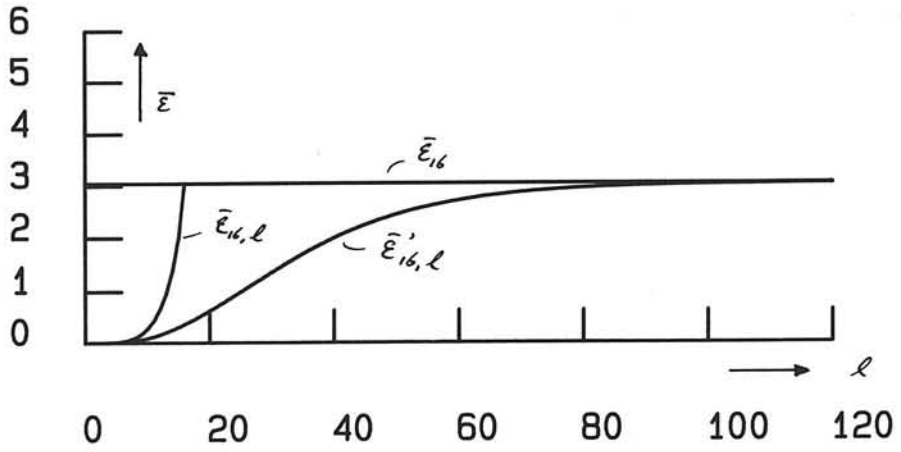


Fig. 3

6. A PRACTICAL KEY SIGNATURE SCHEME

In the previous sections random permutations, extractions and selections were considered for deriving a signature of a secret key. However in order to be reproducible a signature should not be determined by a random operation. In such cases it is a good practice to use some kind of pseudo-random mechanism to imitate real randomness. In the key signature scheme presented here a linear feedback shiftregister is used to produce a pseudo random sequence of addresses. These addresses in turn are used to select a number of keybits, which form the signature. The key itself determines the actual shift register sequence. Figure 4 depicts this scheme.

We argue that this scheme is probably as good as a random selection, because the selection sequence is determined by the random, secret key and therefor unpredictable a priori. But also because of the randomness of the selected keybits, the linearity of the shift register will be hidden.

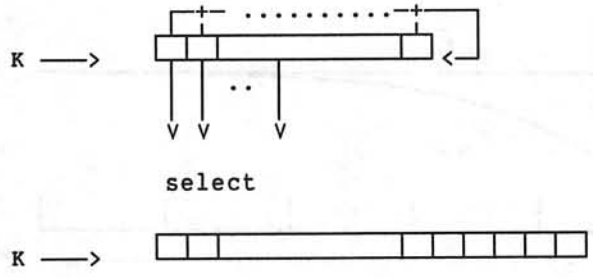


Fig. 4

7. CONCLUSIONS

In this paper it was shown that simple random operations on a secret key in order to obtain its signature reveal very little about this key. For the case of random permutations and random extractions expressions were found for the amount of given away information about the key, and for random selections a usefull upperbound was derived. It was clearly demonstrated that very little information is given away in the case of a random selection of keybits. Moreover a practical pseudo-random selection key signature scheme was shown which can be implemented very easily. The theory presented here also provides an insight into the information theoretic aspects of shift register devices used for encryption of messages.

REFERENCES

[1] C.H. MEYER & S.M. MATYAS, 'Cryptography'

[2] M. HALL JR., 'Combinatorial Theory'

THE Pe-SECURITY DISTANCE AS A GENERALIZED UNICITY DISTANCE

Johan van Tilburg*, Dick E. Boekee**

1. INTRODUCTION

The user of cipher systems makes it possible to send secret messages via public insecure channels. However, the secrecy of the message depends highly on the cryptographic performance of the cipher system used. When evaluating the theoretical strength of cipher systems with a probabilistic model, it is assumed that the cryptanalyst behaves rationally, that he or she at least knows the set of transformations, the statistics of the message and the key source.

In Shannon's paper [1] it is pointed out that if the cryptanalyst intercepts a cryptogram, that he or she is able to calculate the a-posteriori probabilities of the various possible messages and keys which might have produced this cryptogram. This set of a-posteriori probabilities describes how the cryptanalyst's knowledge of the message and the key gradually becomes more precise as more enciphered text is intercepted. Shannon used as a measure of theoretical strength the equivocation, which deals with a simplified description of the set of a-posteriori probabilities. Shannon's approach has led to the so-called (classical) unicity distance and will be described in section 2.

Although Shannon's information measure leads to easy manipulation in a natural and intuitive way between different probability distributions, still the underlying relevant parameter is the error probability (or probability of incorrect identification) Pe faced by the cryptanalyst.

In cases where determining the error probability in a direct manner is quite involved, bounds on Pe can be considered. By bounding Pe with information measures and/or distance measures, a region is determined

---

* Datacommunication Section, PTT - Dr. Neher Laboratories
P.O. Box 421, 2260 AK Leidschendam, the Netherlands
** Information Theory Group, Delft University of Technology
P.O. Box 5031, 2600 GA Delft, the Netherlands

in which the actual Pe can be found. The uncertainty in the value of Pe is resolved only in limiting cases where the bounds are tight. In this context it seems to be a natural way to make use of the concept of distance measures since the error probability is actually a distance measure itself. This approach, which can be found in Van Tilburg and Boekee [2], has led to the introduction of the Pe-security distance and will be described in section 3. Finally, in section 4 conclusions are drawn.

2. THE CLASSICAL UNICITY DISTANCE

As it appears from the literature, the Unicity Distance (UD) is often linked to the random cipher model and/or the key equivocation. As a result of this several authors have given definitions of the unicity distance which are incomplete, biased and more restrictive than necessary. As a consequence of this the UD is easily given a wrong interpretation. To clarify this confusion let us first consider the UD as derived by Shannon [1, p. 693]. Shannon defined the (classical) UD for the message based on a ciphertext-only-attack, $UD_{RC}(M^L/E^L)$, by evaluating the key equivocation and the key appearance characteristic in a Random Cipher (RC). As a result he obtained

$$UD_{RC}(M^L/E^L) = H(K)/D(M^L), \tag{1}$$

where $D(M^L) = \log |M| - H(M^L)/L$ is the average redundancy per message source symbol in a sequence M^L of L message source symbols, $H(K) = \log |K|$ is the entropy of the key source and E^L is the enciphered message of length L.

Unfortunately this UD is sometimes confused with the UD for the key based on a ciphertext-only-attack. It trivially holds that

$$UD_{RC}(K/E^L) \geq UD_{RC}(M^L/E^L), \tag{2}$$

so that (1) and (2) yields

$$UD_{RC}(K/E^L) \geq H(K)/D(M^L), \tag{3}$$

Hellman [3] has proved that the RC-model actually defines a lower bound on the existence of good cipher systems. For this reason (1) and (3) give a worst case indication of the strength of a cipher system. However, these results are not precise and the interpretation depends highly on the size of the key space used and the message source used.

A general relation between the key equivocation and the message equivocation is given by

$$H(K/E^L) - H(M^L/E^L) = H(K/M^L E^L), \tag{4}$$

in which $H(K/M^L E^L)$ is the key appearance equivocation. The left hand side of the equality is based on a ciphertext-only-attack, while the right hand side is based on a known-plaintext-attack. Hence Dunham [4] concludes that there is a fundamental trade-off between protecting the key under a known-plaintext-attack and protecting the message under a ciphertext-only-attack when the size of the key space is fixed. And also, when designing a cipher system which is to be strong under a ciphertext-only-attack on the message, (4) suggests that it consequently will be weak under a known-plaintext-attack.

From (4) it also follows that

$$H(K/E^L) \geq H(M^L/E^L) \tag{5}$$

and thus

$$UD(K/E^L) \geq UD(M^L/E^L), \tag{6}$$

with equality if the key appearance equivocation is zero, which is in agreement with (2). In general, the key equivocation is given by

$$H(K/E^L) = H(KE^L) - H(E^L). \tag{7}$$

If the message source and the key source are stochastically independent, then the key equivocation becomes

$$H(K/E^L) = H(K) + H(M^L) - H(E^L). \tag{8}$$

Using the inequality $H(E^L) \leq \log |E^L| = L.\log |E|$ and the fact that $|E| = |M|$, we easily find that $H(K/E^L)$ in (8) can be lower bounded by

$$H(K/E^L) \geq H(K) + H(M^L) - L.\log |E|$$
$$= H(K) - L.D(M^L). \qquad (9)$$

If we define the unicity distance $UD(K/E^L)$ for the key based on a ciphertext-only-attack as the distance where $H(K/E^L)$ is zero, then from (9) it follows that

$$UD(K/E^L) \geq H(K)/D(M^L). \qquad (10)$$

It is tempting to say that the RC reaches this lower bound, i.e. $UD(K/E^L) \geq UD_{RC}(K/E^L)$. The next Lemma may help to make this statement clear.

Lemma 1. The average probability of error (or probability of incorrect key identification) in a random cipher model at classical unicity distance is given by

$$Pe_{RC}(K/E^{UD}) = (|K| - 1)/|K|^2. \qquad (11)$$

Proof. Suppose there are $|K|$ different and independent keys in the RC so that $Pe_{RC}(K/E^L) = \bar{n}_k/|K|$ in which $\bar{n}_k$ is the average number of spurious key decipherments. According to Hellman [3, Theorem 1] we have $\bar{n}_k = (|K| - 1).2^{-L.D(M^L)}$. Substitution yields

$$Pe_{RC}(K/E^L) = (1-|K|^{-1}).2^{-L.D(M^L)}. \qquad (12)$$

At classical UD it holds that $L = H(K)/D(M^L)$. In addition, the keys are equiprobable so that $H(K) = \log |K|$. Substitution yields the Lemma. □

This Lemma tells us that the cryptanalyst is faced with an error probability (unequal to zero) at the classical UD. For this reason $H(K/E^{UD})$ can not be zero and the lower bound (10) does not hold in general. This

also shows that Blom's general derivation [5, p. 9] of Hellman's result is not as general as suggested. Actually (10) is restricted to the limiting case where $H(K/E^L) = 0$ can be obtained.

Finally, to illustrate the difference between $UD(K/E^L)$ and $UD(M^L/E^L)$ with an extreme example we mention that for a simple substitution cipher using the English language we may obtain $UD(K/E^L) = 1500$ and $UD(M^L/E^L) = 25$ respectively.

3. THE Pe-SECURITY DISTANCE

To understand the introduction and the interpretation of the Pe-security distance (Pe-SD) as a generalized unicity distance it is necessary to formalize the UD.

Definition 1. The unicity distance of a cipher model (including the message source) is the minimal expected length of ciphertext, generated by this model, after which the enciphered text (cryptogram) can be broken on the average.

□

This definition of the UD covers at least five important aspects. The first one is that the UD is a minimal expected length. For an accurate interpretation of the UD it might be important to consider higher order statistics too. The second aspect follows from the fact that the cipher model includes the message source also. It is evident that the message source greatly influences the UD. Generally speaking, it is important to know the proces which has generated the enciphered text. The third aspect is inherently related to the plaintext, i.e. the text generated by the message source. If the plaintext is known, then we speak of a UD based on a known-plaintext-attack. If the plaintext is unknown, then we speak of a UD based on a ciphertext-only-attack. The fourth aspect has a strong affinity with the previous one. What is our object: the key or the message? As illustrated in section 2 they might be quite different. Finally, the fifth aspect and this might be the most important one: what is the meaning of "can be broken on the average".

Most of the definitions in the open literature approach this problem by introducing the key equivocation and adverbs like almost and near-

ly. Jürgensen and Matthews [6] addressed this problem by defining the β-UD as $MIN\{L \mid H(K/E^L) \leq \beta\}$. However, the key equivocation defines an upper bound on the error probability and is usually only tight for large L. Consequently, this contradicts the "minimal expected length" in definition 1 and the worst case approach in general. Moreover, the interpretation of the β-UD is not unique and depends highly on the size of the key space used. To avoid these problems one can link a probability function to "can be broken on the average".

For example, if the error probability (or probability of incorrect identification) faced by the cryptanalyst is used, then the cryptogram space is divided into equivalence classes, one of which has an unique average error probability Pe for a given cipher model. If we do this, then it follows from (11) that the classical UD is directly related to an average error probability which is inversely proportional to the cardinality of the key space used. As a result, the meaning of the UD for different sizes of the key space is also different, in the sense of Pe. Actually, that is not what one prefers. For this reason a constant average error probability is taken as a starting point and definition 1 can be restated as a security distance.

Definition 2. The Pe-security distance of a cipher model (including the message source) is the minimal expected length of the ciphertext, generated by this model, necessary in order to be able to break the enciphered text (cryptogram) with an average error probability (or probability of incorrect identification) of at most Pe.

□

This definition provides a theoretically attractive measure of cryptographic performance of a cipher system. In order to give a mathematically suitable definition it is necessary to restrict ourselves to a specific attack, for example as is done in the next definition [2, Definition 4.2].

Definition 3. The Pe-security distance for the key based on a ciphertext-only-attack is defined by

$$\gamma\text{-SD}(K/E^L) = \text{MIN}\{L \mid Pe_m(K/E^L) \leq \gamma\}$$

where

m is the actual cipher model, and

γ is a value of the error probability Pe.

□

Remark. Depending on what ones object is i.e. the key or the message, the Pe-SD can be based on $Pe_m(K/E^L)$ or on $Pe_m(M^L/E^L)$. If a known-plaintext-attack is used one may use $Pe_m(K/M^LE^L)$. From the definition it also follows that the Pe-SD depends on the cipher model m used and the desirable value γ of Pe.

The next corollary [2, Corollary 4.2] shows that the Pe-security distance can be considered as a generalized unicity distance.

Corollary 1. The Pe-security distance includes the classical unicity distance as a special case.

Proof. For an RC-model we have (12):

$$Pe_{RC}(K/E^L) = (1-|K|^{-1}) \cdot 2^{-L \cdot D(M^L)}.$$

If we choose $\gamma = (|K|-1)/|K|^2$, one easily obtains

$$\text{MIN}\{L \mid L \geq H(K)/D(M^L)\},$$

which is the classical unicity distance.

□

Whereas determining the error probability (and thus the Pe-SD) in a direct manner is quite involved, one can make use of lower bounds only. This is in agreement with the worst case approach. A natural way to obtain lower bounds is to make use of the concept of distance measures, as shown in Van Tilburg and Boekee [2], since the error probability is actually a distance measure itself. However, if the key equivocation is used one must realize that this measure defines an upper bound on the error probability. For this reason, Fano's inequality

$H(K/E^L) \leq H(Pe) + Pe.\log(|K|-1)$ has to be used to obtain the lower bound.

4. CONCLUSION

Shannon obtained a unicity distance for the message based on a ciphertext-only-attack in a random cipher model, which is referred to as the classical unicity distance. Hellman has shown that Shannon's random cipher result actually defines a lower bound on the existence of good ciphers. Later on, Blom generalized this result in terms of key equivocation. However, Blom's result is not as general as suggested.

After formalizing the unicity distance a potential ambiguity can be found in most definitions in the literature. This ambiguity can be resolved by introducing a probability function.

A natural probability function is one based on the expected error probability (or probability of incorrect identification) faced by the cryptanalyst. As a direct result the Pe-security distance is introduced as the minimal expected amount of enciphered text necessary to make an average probability of incorrect identification of at most Pe.

Finally, if the expected error probability Pe is set equal to $(|K|-1)/|K|^2$, then the classical unicity distance is obtained, which shows that the Pe-security distance can be considered as a generalized unicity distance.

REFERENCES

[1] C.E. Shannon, Communication theory of secrecy systems, Bell. Syst. Tech. J. 28, pp. 656-715, 1949.

[2] J. van Tilburg and D.E. Boekee, Divergence Bounds on Key Equivocation and Error probability. To be published in the proceedings of Crypto'85.

[3] M.E. Hellman, An Extension of the Shannon Theory Approach to Cryptography, IEEE Trans. Inform. Theory IT-23, pp. 289-294, 1977.

[4] J.G. Dunham, Bounds on Message Equivocation for Simple Substitution Ciphers, IEEE Trans. Inform. Theory IT-26, pp. 522-527, 1980.

[5] R. Blom, Bounds on Key Equivocation for Simple Substitution Ciphers, IEEE Trans. Inform. Theory IT-25, pp. 8-18, 1979.

[6] H. Jürgensen and D.E. Matthews, Some result of the information theoretic analysis of cryptosystems, Proceedings of Crypto'83, Santa Barbara, California, August 1983, pp. 303-356.