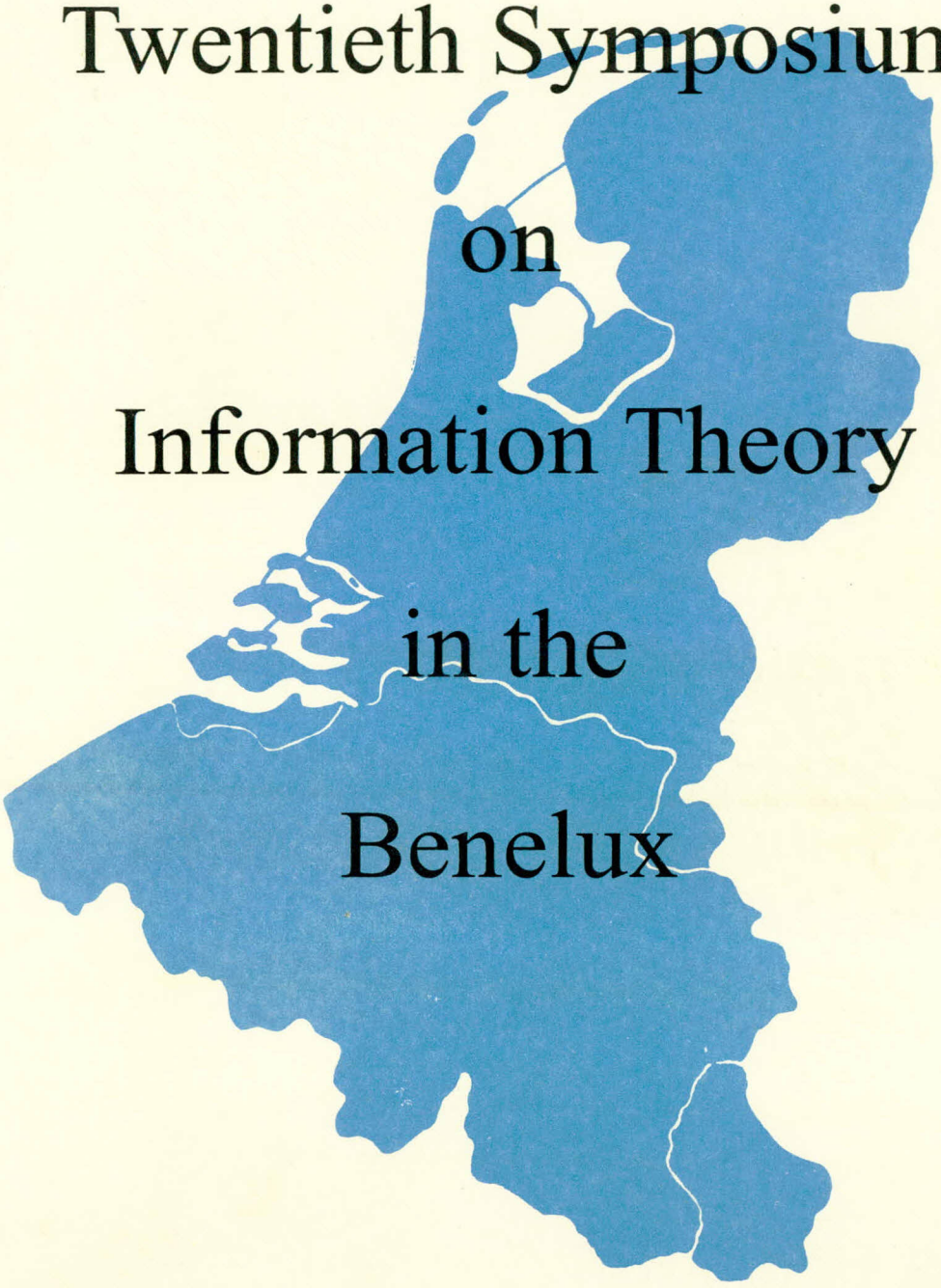
Twentieth Symposium on Information Theory in the Benelux



Haasrode, Belgium
May 27 & 28, 1999

L. Tolhuizen

PROCEEDINGS OF THE

20TH SYMPOSIUM

ON

INFORMATION THEORY

IN THE

BENELUX

PROCEEDINGS OF THE 20TH SYMPOSIUM ON INFORMATION THEORY IN THE BENELUX

May 27-28, 1999
Haasrode, Belgium

by A. Barbé
E. van der Meulen
P. Vanroose

Published by :
Werkgemeenschap voor Informatie- en Communicatietheorie,
Enschede, The Netherlands, 1999

The 20th Symposium on Information Theory in the Benelux has been organised by :

Prof. A. Barbé, Prof. E.C. van der Meulen, Dr. P. Vanroose

from the Katholieke Universiteit Leuven, Belgium, Department of Mathematics and Department of Electrical Engineering,
on behalf of the Werkgemeenschap voor Informatie- en Communicatietheorie and the IEEE Benelux Chapter on Information Theory.

Program Committee : A. Barbé, B. Preneel, E.C. van der Meulen, J. Vandewalle, L. Van Eycken, P. Vanroose, P. Wambacq

Financial support has been received from the Gauss Foundation

CIP-gegevens Koninklijke Bibliotheek, Den Haag (NL)

Proceedings

Proceedings of the ... symposium on information theory in the Benelux.
- Enschede : Werkgemeenschap Informatie- en Communicatietheorie
Proceedings of the 20th symposium : May 27-28, 1999,
Haasrode, Belgium / by A. Barbé, E.C. van der Meulen, P. Vanroose, eds.

Met lit. opg.
ISBN : ISBN 90-71048-14-4
SISO
Trefw. : informatietheorie

Previous Symposia :

1.	1980	Zoetermeer, The Netherlands	Department of Electrical
2.	1981	Zoetermeer, The Netherlands	Engineering, Technical
3.	1982	Zoetermeer, The Netherlands	University Delft
4.	1983	Haasrode, Belgium	ISBN 90-334-0690-X
5.	1984	Aalten, The Netherlands	ISBN 90-71048-01-2
6.	1985	Mierlo, The Netherlands	ISBN 90-71048-02-0
7.	1986	Noordwijkerhout, The Netherlands	ISBN 90-6275-272-1
8.	1987	Deventer, The Netherlands	ISBN 90-71048-03-9
9.	1988	Mierlo, The Netherlands	ISBN 90-71048-04-7
10.	1989	Houthalen, Belgium	ISBN 90-71048-05-5
11.	1990	Noordwijkerhout, The Netherlands	ISBN 90-71048-06-3
12.	1991	Veldhoven, The Netherlands	ISBN 90-71048-07-1
13.	1992	Enschede, The Netherlands	ISBN 90-71048-08-X
14.	1993	Veldhoven, The Netherlands	ISBN 90-71048-09-8
15.	1994	Louvain-la-Neuve, Belgium	ISBN 90-71048-10-1
16.	1995	Nieuwerkerk a/d IJssel, The Netherlands	ISBN 90-71048-11-X
17.	1996	Enschede, The Netherlands	ISBN 90-365-0812-6
18.	1997	Veldhoven, The Netherlands	ISBN 90-71048-12-8
19.	1998	Veldhoven, The Netherlands	ISBN 90-71048-13-6
20.	1999	Haasrode, Belgium	ISBN 90-71048-14-4

TABLE OF CONTENTS

Page

- **Guest lecture : Automatic sequences**
  Fritz von Haeseler 1

Session 1 : **Data compression, video coding**

- Nonasymptotic lower bound on the maximal cumulative redundancy of universal coding 17
  Vladimir B. Balakirsky, Frans M.J. Willems
- Complexity reducing techniques for the CTW algorithm 25
  Paul A.J. Volf, Frans M.J. Willems, Tjalling J. Tjalkens
- Stochastic language modelling using context tree weighting 33
  Peter Vanroose
- A fast full search equivalent for mean-shape-gain vector quantizers 39
  Jean Cardinal
- Efficient coding of non-static texture in 3D scenes 47
  Stef Desmet, Bert DeKnuydt, Luc Van Gool, Luc Van Eycken
- High-quality embedded compression for digital TV 55
  Mihaela v.d. Schaar-Mitrea, Peter H.N. de With

Session 2 : **Cryptography 1**

- Timing attack: what can be achieved by a powerful adversary ? 63
  Gaël Hachez, François Koeune, Jean-Jacques Quisquater
- The digital timestamping problem 71
  Bart Van Rompay, Bart Preneel, Joos Vandewalle
- Design of a secure timestamping service with minimal trust requirement 79
  H. Massias, X. Serret Avila, J.-J. Quisquater

Session 3 : **Cryptography 2**

- Characterization of the secrecy of a common key constructed via data transmission over the two-way « AND » channel
  Vladimir B. Balakirsky 87

- An attack against the Alabbadi-Wicker scheme
  Sheng-bo Xu, Jeroen Doumen 95

- Diffusion analysis of Feistel networks
  Jorge Nakahara Jr, Joos Vandewalle, Bart Preneel 101

- Anonymity controlled electronic payment systems
  Joris Claessens, Bart Preneel, Joos Vandewalle 109

Session 4 : **Speech processing, communication networks**

- Improved noise robustness for speech recognition by adaptive SVD-based filtering
  Kris Hermus, Patrick Wambacq and Dirk Van Compernolle 117

- On the objective speech quality of the TETRA system
  C.H.Slump, T. de Bont, A.M. Mertens, K. Verwey 125

- Proposal for an adaptive multi-rate coder for GSM
  A. Gerrits, A. Koppelaar, R. Taori, R. Sluijter, C. Baggen, E. Hekstra-Nowacka 133

Session 5 : **Classification, pattern recognition, neural networks**

- Linear feature transformations based on MCE and MMI
  Kris Demuynck, Patrick Wambacq 141

- Generalization capacity of neural networks for the classification of ovarium tumours
  Ellen Lerouge, Sabine Van Huffel 149

- Generalized moment invariants for viewpoint and illumination independent color pattern recognition
  Florica Mindru, Theo Moons, Luc Van Gool 157

Session 6 : **Shannon theory, detection, signal processing**

- On certain channels with a random parameter 165
  Mark S.Pinsker, Vyacheslav V.Prelov, Edward C. van der Meulen

- On error correction in information feedback schemes 173
  M.S. Bargh, J.P.M. Schalkwijk

- More on the duality between source and channel coding 181
  V.N. Koshelev, E.C. van der Meulen

- An efficient algorithm for selecting optimal configurations 189
  of AR-coefficients
  Rudy Moddemeijer

- Optimal resource management algorithm for adaptive modelling 197
  J. Levendovszky, L. Kovács, I. Koller, E.C. van der Meulen

Session 7 : **Channel coding, zero-error information theory**

- Multirate block codes 205
  J.B.H. Peek

- The binary multiplying channel without feedback : New rate 215
  pairs in the zero-error capacity region
  Ludo Tolhuizen

- Error correction capabilities of concatenated coding schemes 219
  with single-trial bounded distance decoding and optimized erasing
  Jos H. Weber, Khaled A.S. Abdel-Ghaffar

- Coding for random access communications 227
  A.J. Han Vinck

AUTHORS INDEX

Abdel-Ghaffar K.A.S.	219
Baggen C.	133
Balakirsky V.B.	17, 87
Bargh M.S.	173
Cardinal J.	39
Claessens J.	109
de Bont T.	125
de With P.H.N.	55
DeKnuydt B.	47
Demuynck K.	141
Desmet S.	47
Doumen J.	95
Gerrits A.	133
Hachez G.	63
Hekstra-Nowacka E.	133
Hermus K.	117
Koeune F.	63
Koller I.	197
Koppelaar A.	133
Koshelev V.N.	181
Kovács L.	197
Lerouge E.	149
Levendovszky J.	197
Massias H.	79
Mertens A.M.	125
Mindru F.	157
Moddemeijer R.	189
Moons T.	157
Nakahara J.	101
Peek J.B.H.	205
Pinsker M.S.	165
Prelov V.V.	165
Preneel B.	71,101,109
Quisquater J.J.	63,79
Schalkwijk J.P.M.	173
Serret Avila X.	79
Sluijter R.	133
Slump C.H.	125
Taori R.	133
Tjalkens T.J.	25
Tolhuizen L.	215
v.d. Schaar-Mitrea M.	55
Van Compernolle D.	117
van der Meulen E.C.	165,181,197
Van Eycken L.	47
Van Gool L.	47,157
Van Huffel S.	149
Van Rompay B.	71
Vandewalle J.	71,101,109
Vanroose P.	33
Verwey K.	125
Vinck A.J.	227
Volf P.A.J.	25
von Haeseler F.	1
Wambacq P.	117,141
Weber J.H.	219
Willems F.M.J.	17,25
Xu S.	95

PREFACE

Since 1980 an Annual Symposium on Information Theory in the Benelux has been held. This sequence originated as an idea of Prof. Y. Boxma (T.U.Delft) to provide a platform where researchers in Information Theory from the various universities and other institutions in the Benelux could present their recent research results and have interaction. The response was so enthusiastic that in 1984 an organisation was formed, called the "Werkgemeenschap voor Informatie- en Communicatietheorie" (WIC). It is this Working Community for Communication and Information Theory that since 1984 formally organises this Annual Symposium.

In 1989 the Information Theory Chapter in the Benelux Section of IEEE was formed. Since then, the IT Chapter cosponsors the organisation of this Annual Benelux Symposium.

The location of the Information Theory Symposium usually rotates over various sites in the Benelux, depending on which group takes the responsibility to carry out the local organisation. This year, the 20th Symposium on Information Theory in the Benelux is held in Belgium. It is organised locally by members of the Department of Electrical Engineering and the Department of Mathematics of the Katholieke Universiteit Leuven.

Thus, this year we celebrate the 4th lustrum of the Annual Benelux Symposium on Information Theory, the 15th anniversary of the WIC and the 10th anniversary of the IEEE IT Chapter. For this occasion we have invited a special guest speaker, PD. Dr. Fritz von Haeseler, whose talk on automatic sequences will hopefully inspire all participants.

These proceedings contain the papers submitted to and to be presented at this Symposium. Altogether there are 1 invited and 28 contributed papers. The

topics channel coding, classification, communication networks, cryptography, data compression, neural networks, pattern recognition, sequences, Shannon detection theory, signal processing, speech processing, video coding and zero-error information theory are all represented in the program.

The fact that a relatively small group of researchers is able to come up every year with about 30 papers in a wide range of topics within Information Theory demonstrates the strength and vitality of this group and of the WIC in particular. This success augurs well for the performance and well-being of our research community in the future. Whereas in the next millennium the emphasis will undoubtedly be on more technologically oriented topics, we are confident that the research presented here will provide an important contribution towards future developments in information technology.

We would like to express our sincere thanks to Mrs. Ida Tassens from ESAT for the skilful assistance in putting these proceedings together.

André Barbé
Edward C. van der Meulen
Peter Vanroose

Leuven, May 1999

Automatic sequences (Selected topics)

Fritz von Haeseler[1]
CeVis Universität Bremen
Universitätsallee 29
28359 Bremen
Germany

1 Introduction

The goal of this note is to provide a glimpse into the realm of automatic sequences. The notion of automaticity comes in several disguises, all equivalent in a strict mathematical sense. In the course of this note we are going to know all these disguises.

It is difficult to trace back all the contributions which have been made in the vast field of automatic sequences (and its equivalents). But it is beyond doubt that the pioneering work of A. Thue [56, 57] had a strong impact on the development of the theory of automatic sequences, where automatic sequences were investigated within the framework of combinatorics on words.

Nowadays automatic sequence appear in several branches of mathematics, physics and computer science. In this short note we will sketch some of the fields where automatic sequences play a rôle. This article is by no way exhaustive, it covers only a small part of the world of automatic sequences – as does the list of references. Some further introductionary articles which emphasize and stretch out in directions different to ours are [3] and [1].

We begin our travel through the world of automatic sequences with some well known 'classic' examples. Each of these examples exemplifies different aspects of automatic sequences.

2 A bouquet of examples

2.1 The Thue-Morse sequence

It is not exaggerated to state that the (Prouhet)-Thue-Morse sequence is 'the' automatic sequence, see [9] where almost all different facets of the Thue-Morse sequence are discussed.

Around the years 1906–1912 A. Thue was wondering whether it is possible to construct an infinite binary sequence such that the sequence contains no cube, i.e., no subword of the form $\omega\omega\omega$, where ω is a finite sequence (word) of zeros and ones. It is plain that it is impossible to construct an infinite binary sequence which contains no square. In fact, the reader should try to prove that there is no such finite sequence of length greater than or equal to 4. A. Thue finally found a solution which can be generated in the following way. We consider the set of words formed by the letters 0 and 1 and define a substitution by $\sigma(0) = 01$, $\sigma(1) = 10$. If ω is any word with letters 0, 1, we define $\sigma(\omega)$ by replacing each 0 by 01 and 1 by 10, respectively. Starting with 0 and iterating the map σ we obtain a

[1]Current Address: ESAT/SISTA, KU Leuven, Kardinaal Mercierlaan 94, 3001 Heverlee, Belgium

sequence of words

$$0 \mapsto 01 \mapsto 0110 \mapsto 01101001 \mapsto 0110100110010110 \mapsto \dots$$

and we see that the beginning of the words stabilizes. Therefore it is justified to speak of the limit of this substitution. The limit is an infinite sequence $\underline{t} = (t_n)_{n\geq 0}$ the *Thue-Morse sequence.* It is an example of an infinite binary sequence without subword that is a cube.

By its construction, we see immediately that the Thue-Morse sequence can be constructed also recursively, i.e., if we set $t_0 = 0$, then t_n is given by

$$\begin{aligned} t_{2n} &= t_n \\ t_{2n+1} &= 1 - t_n \end{aligned}$$

for all $n \in \mathbb{N}$. Since the sequence has only the values 0 and 1, it is natural to replace the definition of t_{2n+1} by $t_{2n+1} = t_n + 1$, where the addition is carried out modulo 2.

2.2 Paperfolding sequence

The next sequence, again an automatic sequence, is a species in the huge zoo of paperfolding sequences, see, e.g., [27, 45, 46, 50] and the literature cited there.

Take a long strip of paper, mark its right end and fold repeatedly such that the marked end is always the right end of the folded strip. Unfolding the strip in such a way that the folds form right angles and moving from the marked end towards the unmarked end along the strip produces a sequence of left- and right turns.

Folding the paper once we obtain one left turn denoted by L. Folding the paper two times, the unfolding yields LLR, a third fold gives the word $LLRLRR$. Thus we have a sequence of words

$$L \mapsto LLR \mapsto LLRLLRR \mapsto LLRLLRRLLLRRLRR \dots,$$

and, as for the Thue-Morse sequence, the n-th word is the initial word of the $n+1$-st sequence. We may therefore speak of the limit of this process. A careful analysis of the folding procedure reveals that we can obtain the $n+1$-st word ω_{n+1} in the following way: if $\omega_n = a_0 a_1 \dots a_N$, then is given by

$$\omega_{n+1} = a_0 a_1 \dots a_N \; L \; \hat{a_N} \dots \hat{a_0},$$

where $\hat{L} = R$ and vice versa. The limit sequence $\underline{pf} = (p_n)_{n\geq 0}$ of this process is called a *paperfolding sequence.*

2.3 Baum-Sweet sequence

The Baum-Sweet sequence [16] is a binary sequence $\underline{bs} = (b_n)_{n\geq 0}$, where the entries are determined by a condition imposed on the binary expansion of n. We set $b_0 = 1$ and define b_n for $n \geq 1$ by

$$b_n = \begin{cases} 0 & \text{the binary expansion of } n \text{ contains an odd number off successive zeros} \\ 1 & \text{otherwise} \end{cases}$$

The first few terms of the Baum-Sweet sequence are given by

$$1101101101001\dots$$

Note that we have that $u_{2n+1} = u_n$ (multiplication by two and addition of one does not change the number of successive zeros in the binary expansion of n), $u_{4n} = u_n$ (multiplication by 4 does not change the parity of successive zeros), and $u_{4n+2} = 0$ (multiplication by 4 and addition of two creates an odd number of successive zeros). As we shall see later the Baum-Sweet sequence is also an example of an automatic sequence.

2.4 The Cantor set

Let $c(x)$ be a power series with coefficients in the real numbers $\mathbb{R}$. We are interested in the solutions of the following functional equation

$$c(x) = (1 + x^2)c(x^3).$$

It is clear that if $c(x)$ is a solution of the above equation, so is $rc(x)$, where $r \in \mathbb{R}$. Thus we may assume that $c(x) = 1 + c_1x + c_2x^2 + \dots$ and we find the solution

$$c(x) = \prod_{j=0}^{\infty}(1 + x^{2\cdot 3^j}) = 1 + x^2 + x^6 + x^8 + x^{18} + x^{20} + x^{24} + x^{26} + \dots$$

which is the unique solution satisfying $c_0 = 1$. The sequence $\underline{c} = (c_n)_{n\geq 0}$ is a binary sequence and automatic. It is related to the Cantor set $\mathcal{C} \subset [0, 1]$ in the following way

$$\mathcal{C} = \lim_{n\to\infty} \frac{1}{3^n} \{j \mid c_j = 1 \text{ and } 0 \leq j < 3^n\},$$

where the limit is taken w.r.t. the Hausdorff distance of compact non-empty sets, see, e.g., [29, 31, 47] for more details on fractal geometry.

3 Automatic sequences and finite automata

The goal of this section is to give the definition of an automatic sequence and to state their basic properties. In particular, we explain the relation between automatic sequences and finite automata, thus clarifying the expression automatic sequence.

A sequence $(f_n)_{n\geq 0}$, where $f_n \in A$ and A (the alphabet) is a finite set, is denoted by its generating function

$$f(x) = \sum_{n=0}^{\infty} f_n x^n,$$

or simply by $\underline{f}$. The set of all sequences is denoted by $A(x)$. The advantage of this notion becomes clear if the set A carries an additional structure like a ring structure. We can then speak of sums and products of sequences. The sum of two sequences $f(x)$, $g(x)$ is simply the sum of the two power series, and the product of $f(x)$ and $g(x)$ is the usual product of two power series.

From now on let k denote a natural number such that $k \geq 2$.

Definition 3.1 *Let $k \geq 2$ be a natural number and let A be a finite set. The map $\partial_{k,j} : A(x) \to A(x)$, $j = 0, 1, \dots, k-1$, defined as*

$$\partial_{k,j}(\underline{f}) = \partial_{k,j}\left(\sum_{n=0}^{\infty} f_n x^n\right) = \sum_{n=0}^{\infty} f_{kn+j} x^n$$

is called j-decimation (w.r.t. k).

If it is clear from the context, we simply speak of a j-decimation and denote it by ∂_j. Decimations provide the proper framework for the repeated selection of subsequences.

The set of all possible subsequences obtained by repeated decimations is called the *k-kernel* of a sequence. It is denoted by

$$\ker_k(\underline{f}) = \ker_k(f(x)) = \{\partial_{j_1} \circ \ldots \circ \partial_{j_n}(\underline{f}) \mid j_1, \ldots, j_n \in \{0, 1, \ldots, k-1\},\ n \in \mathbb{N}\}.$$

Note that for $n = 0$ we obtain the sequence $\underline{f}$.

Definition 3.2 *Let $f(x) \in A(x)$ be a sequence and let k be a natural number such that $k \geq 2$. The sequence $f(x)$ is called* k-automatic *if its k-kernel is finite.*

Examples

1. The Thue-Morse sequence is an example of a 2-automatic sequence. Indeed, due to the recursive definition one easily computes that the 2-kernel consists of the sequences $(t_n)_{n\geq 0}$ and $(t_n + 1)_{n\geq 2}$.

2. By analysing the process of generating the paperfolding sequence one obtain that the 2-kernel has for elements, which are $\underline{pf}$ and the periodic sequences $\underline{v} = (LRLRLR\ldots)$, $\underline{w} = (LLLL\ldots)$, $\underline{u} = (RRRR\ldots)$.

3. The Baum-Sweet sequence is 2-automatic. That is a consequence of the recusive character of the Baum-Sweet sequence we already established. The 2 kernel is given by
$$\ker_s(\underline{bs}) = \{\underline{bs},\ (b_{2n})_{n\geq 0},\ \underline{0}\},$$
where $\underline{0}$ denotes the constant sequence with values equal to 0.

4. The sequence $\underline{c}$ which is related to the Cantor set is a 3-automatic sequence. The 3-kernel contains two elements, namely $\underline{c}$ and $\underline{0}$.

5. A sequence $f(x)$ is called ultimately periodic if there exist K, $N \in \mathbb{N}$ such that
$$f_{n+N} = f_n$$
holds for all $n \geq K$. An ultimately periodic sequence is k-automatic for all $k \geq 2$.

6. The sequence
$$f(x) = \sum_{n=0}^{\infty} x^{n^2}$$
with values in $\{0, 1\}$ is not k-automatic for all $k \geq 2$. [2]

7. The sequence defined by
$$f_k(x) = \sum_{n=0}^{\infty} x^{k^n},$$
where $k \geq 2$, is k-automatic.

[2] It is not too difficult to prove this assertion for a particular value of k. The hard part is to prove it for all k.

Lemma 3.3 *A sequence $f(x) \in A(x)$ is k-automatic if and only if the sequence is k^α-automatic for all $\alpha \in \mathbb{N}$.*

The second of the above examples shows that there exist sequences which are k-automatic for all $k \geq 2$. Due to a theorem of Cobham [23] these are the only examples.

Theorem 3.4 *A sequence is ultimately periodic if and only if there exist k_1, $k_2 \in \mathbb{N}$ such that $\frac{\log k_1}{\log k_2}$ is irrational and $f(x)$ is k_1-automatic and k_2-automatic.*

An application of Cobham's Theorem for the proof of the non-automaticity of certain sequences is given in [5, 8].

As a next step, we introduce finite automata (in [24], finite automata are called uniform tag systems). A finite automaton is a finite 'device' which reads (either from the left to right or from the right to left) words formed with the letters $\{0, 1, \ldots, k-1\}$. During the successive reading the automaton changes its state depending on the current state and the letter just read (the transition map), the final state of the automaton determines the output of the automaton (output function). For a general definition of a finite automaton the reader should consult Eilenberg [30].

We restrict ourself to a definition which fits our needs. A finite (k-)automaton consists of

- a finite set S, the state set,
- a distinguished element $s_0 \in S$, the initial state,
- a map $\tau : \{0, \ldots, k-1\} \times S \to S$, the transition- or input map,
- a map $\phi : S \to A$, the output map and A is the output alphabet.

We denote a k-automaton by $\mathcal{A} = (S, s_0, \tau, A, \phi)$.

If the input word is, e.g., $i_0 i_1 i_2$, where i_0, i_1, $i_2 \in \{0, \ldots, k-1\}$, then the state of the automaton after reading the word (from left to right) the automaton is in the state

$$\tau(i_2, \tau(i_1, \tau(i_0, s_0))) = \tau_{i_2} \tau_{i_1} \tau_{i_0}(s_0)$$

and the corresponding output is $\phi(\tau(i_2, \tau(i_1, \tau(i_0, s_0)))) = \phi(\tau_{i_2} \tau_{i_1} \tau_{i_0}(s_0)) \in A$.

One should think of a finite automaton as a directed graph. The vertices correspond to the possible states of the automaton. There exists an arrow from s_1 to s_2 with label $j \in \{0, \ldots, k-1\}$ if $\tau(j, s_1) = s_2$ holds.

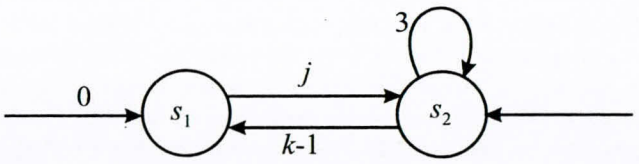


Figure 1: A part of a finite automaton as directed graph

To any finite k-automaton (S, s_0, τ, A, ϕ) we can define a sequence $(u_n)_{n\geq 0}$ with values in A. To this end we represent $n \in \mathbb{N}$, $n \neq 0$, by its k-adic representation

$$n = \sum_{i=0}^{m} j_i k^i,$$

where $j_i \in \{0, 1, \ldots, k-1\}$ **and** $j_m \neq 0$. Therefore n defines a finite word which in turn defines u_n by setting

$$u_n = \phi(\tau_{i_m} \tau_{i_{m-1}} \ldots \tau_{i_0}(s_0)).$$

For $n = 0$ we define $u_0 = \phi(s_0)$. The sequence $u(x) = \sum u_n x^n$ is *generated* by a finite k-automaton. The next Theorem, see Cobham [24], provides the link between k-automatic sequences and sequences generated by a k-automaton.

Theorem 3.5 *A sequence $\underline{f} \in A(x)$ is k-automatic if and only if there $\underline{f}$ is generated by a finite k-automaton.*

It is interesting to note that the construction of the k-automaton which generates a k-automatic sequence is a straightforward procedure. If $\underline{f} \in A(x)$ is a k-automatic sequence, then

$$\mathcal{A}(\underline{f}) = \left(\ker_k(\underline{f}), \underline{f}, \tau : \begin{array}{c} \{0, \ldots, k-1\} \times \ker_k(\underline{f}) \to \ker_k(\underline{f}) \\ (j, \underline{h}) \mapsto \partial_j(\underline{h}) \end{array}, A, \phi : \begin{array}{c} \ker_k(\underline{f}) \to A \\ \underline{h} \mapsto h_0 \end{array} \right)$$

defines a k-automaton that generates the sequence. In [37], the directed graph associated with the automaton $\mathcal{A}(\underline{f})$ is called the kernel graph of $\underline{f}$. The kernel graph allows to decide whether an automatic sequence is (ultimately) periodic or not.

Theorem 3.6 *If $\underline{f} \in A(x)$ is k-automatic, then the automaton $\mathcal{A}(\underline{f})$ is the smallest (with regard to the number of states) of all automata (S, s_0, τ, A', ϕ) that generate $\underline{f}$ and satisfy*

$$\phi(s) = \phi(s') \text{ if } \tau(0, s) = s'.$$

Although the result seems to be folklore, the author knows no reference, except [41].

In Figure 2, we present the kernel graph of the paperfolding sequence. The computation of the kernel graph for the remaining examples is left to the reader.

4 Automatic sequences and substitutions

In this section we explore the relation between automatic sequences and substitutions. [3] A substitution is basically a kind of rewriting rule. If $a_1 \ldots a_n$ is a word over the (finite) alphabet A, then a substitution σ replaces each letter a_i by a word (or letter) $\omega_{a_i} = \sigma(a_i)$. A substitution is a map σ from A to A^*, the set of all words with letters in A.

Of particular interest for our considerations are the so called constant length substitutions. Any letter $a \in A$ is replaced by a word of length equal to k, where $k \geq 2$, cf., the Thue-Morse sequence from Section 2.1. These substitutions are called *k-substitutions.*

Note that a substitution is also well defined on the set of sequences $A(x)$. A sequence $\underline{f} \in A(x)$ is called a fixed point of the substitution if

$$\sigma(\underline{f}) = \underline{f}.$$

[3] Similar terms are 'inflation rules' or 'morphism'

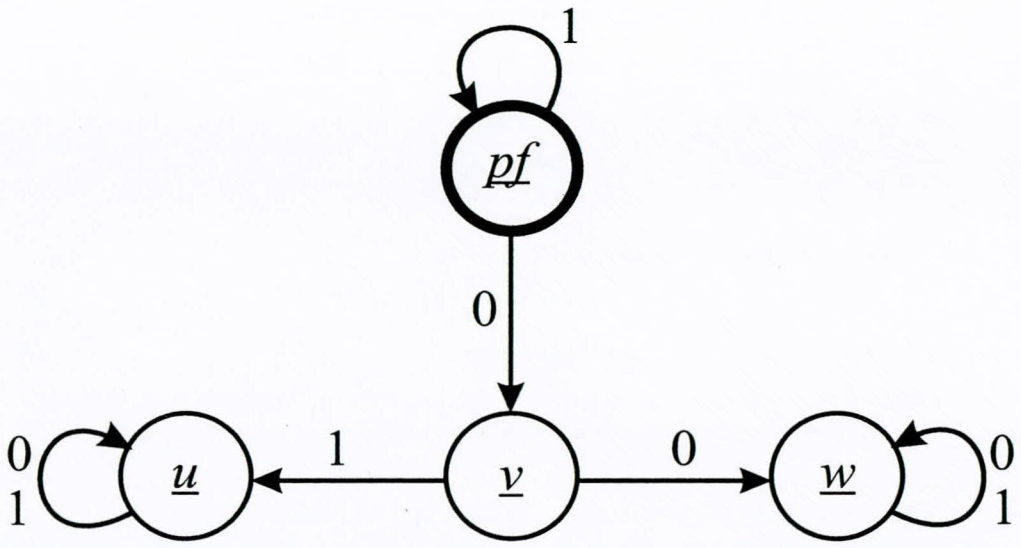


Figure 2: Kernel graph of the paperfolding sequence (see Example 2 on page 4)

Definition 4.1 *A sequence $f(x) \in A(x)$ is* generated by a k-substitution *if there exists a finite set B a k-substitution $\sigma : B \to B^*$ with fixed point $\underline{F} \in B(x)$ and a map $\phi : B \to A$ such that*

$$\phi(\underline{F}) = \phi\left(\sum F_n x^n\right) = \sum \phi(F_n) x^n = \underline{f}$$

holds.

The connection between k-automatic sequences and sequences generated by a k-substitution was again investigated by Cobham in [24].

Theorem 4.2 *A sequence $\underline{f} \in A(x)$ is k-automatic if and only if $\underline{f}$ is generated by a k-substitution.*

By Lemma 3.3, we have that a sequence which is generated by a k^α-substitution is k-automatic and therefore generated by a k-substitution.

Examples

1. The 2-substitution for the paperfolding sequence is given by

$$\sigma(0) = 01,\ \sigma(1) = 21,\ \sigma(2) = 03,\ \sigma(3) = 23.$$

The map $\phi : \{0, 1, 2, 3\} \to \{L, R\}$ is given by

$$\phi(0) = \phi(1) = L$$
$$\phi(2) = \phi(3) = R.$$

2. The 2-substitution for the Baum-Sweet sequence is given by

$$\sigma(0) = 01,\ \sigma(1) = 21,\ \sigma(2) = 13,\ \sigma(3) = 33,$$

and $\phi : \{0,1,2,3\} \to \{0,1\}$ is defined as

$$\begin{aligned}\phi(0) &= \phi(1) = 1\\ \phi(2) &= \phi(3) = 0.\end{aligned}$$

3. The 3-substitution for the sequence $\underline{c}$ is given by

$$\sigma(1) = 101,\ \sigma(0) = 000,$$

and $\phi : \{0,1\} \to \{0,1\}$ is the identity map.

Fixed points $\underline{f}$ of a non-constant length substitution prove to be much more difficult to handle, see [3] and the references cited there.

5 Automatic sequences and equations

The most fruitful relation of automatic sequences and other structures in mathematics was discovered some ten years ago. Of course, in order to explore this connection we have to assume that our set A has an additional structure, namely the structure of a finite field of characteristic p.[4]

To emphasize the fact that our sequence has values in a field we write $\underline{f} \in \mathbb{F}_p(x)$, where $\mathbb{F}_p$ denotes a field of characteristic p. Since we can add and multiply power series (sequences) the question arises whether these operations preserve the automaticity. The answer is in the affirmative.

Lemma 5.1 *Let $f(x), g(x) \in \mathbb{F}_p(x)$. If $f(x)$ and $g(x)$ are k-automatic, then the sequences*

$$\begin{aligned}&rf(x) + sg(x),\ r, s \in \mathbb{F}_p\\ &f(x)g(x)\end{aligned}$$

are also k-automatic.

The proof of the first assertion is an immediate consequence of the $\mathbb{F}_p$-linearity of the decimations. The second assertion is based on a multiplication formula for the decimation of a product, see, e.g., [40].

Note further that the assertion of Lemma 5.1 remains true if the field is replaced by an arbitrary (finite) ring. Also note that k is not related to the characteristic p of the field under consideration.

If k is related to the characteristic p, then we have an algebraic characterization of k-automatic sequences.

Theorem 5.2 *A sequence $f(x) \in \mathbb{F}_p(x)$ is p-automatic if and only if the power series satisfies*

$$r_0(x) + r_1(x)f(x) + r_2(x)f(x)^2 + \ldots + r_D(x)f(x)^D = 0,$$

where $r_j(x)$, $j = 0, \ldots, D$ are rational functions, i.e, quotients of polynomials.

[4] p is always a prime number, the number of elements is equal to p^α, where $\alpha \in \mathbb{N}$. The simplest finite fields are $\mathbb{Z}_p$ (addition and multiplication modulo the prime number p)

A proof may be found in [21, 22]. Since a p-automatic sequence $f(x) \in \mathbb{F}_p(x)$ satisfies an equation of the above type, one says that $f(x)$ is algebraic over $\mathbb{F}_p((x))$, the field of rational functions.

In a field of characteristic p we have the following identity

$$f(x)^p = f(x^p)$$

for all $f(x) \in \mathbb{F}_p(x)$. Therefore the above equation can be transformed into a Mahler equation, i.e., $f(x)$ satisfies

$$p_0(x)f(x) + p_1(x)f(x^p) + p_2(x)f(x^{p^2}) + \ldots + p_L(x)f(x^{p^{L+1}}) = 0,$$

where $p_j(x)$, $j = 0, \ldots, L$ are polynomials with coefficients in $\mathbb{F}_p$.

Examples

1. We show how to achieve an algebraic equation for the Thue-Morse sequence. We consider $t(x) = \sum_{n\geq 0} t_n x^n \in \mathbb{Z}_2(x)$ and obtain

   $$t(x) = \sum_{n\geq 0} t_n x^n = \sum_{n\geq 0} t_{2n} x^{2n} + \sum_{n\geq 0} t_{2n+1} x^{2n+1},$$

   the recurrence relations and $f(x)^2 = f(x^2)$ (in $\mathbb{Z}_2(x)$!) yield

   $$t(x) = t(x)^2 + xt(x)^2 + \frac{x}{1+x^2}$$

   as an algebraic equation for the Thue-Morse sequence.

2. In order to find the algebraic equation for the paperfolding sequence we replace L by 1 and R by 0, respectively. Then $pf(x)$ satisfies the following equation

   $$1 + (1+x)^4 f(x) + x(1+x)^4 f(x)^2 = 0.$$

3. The Baum-Sweet sequence $bs(x) \in \mathbb{Z}_2(x)$ satisfies the equation

   $$1 + xf(x) + f(x)^3 = 0.$$

4. The sequence $c(x) \in \mathbb{Z}_3(x)$ satisfies the equation used to define the sequence, i.e.,

   $$f(x) = (1+x^2)f(x^3) = (1+x^2)f(x)^3.$$

It is plain that the above result translates to arbitrary p-automatic sequences $\underline{f} \in A(x)$. Choose $\alpha \in \mathbb{N}$ such that $p^\alpha \geq \mathrm{Card}(A)$ and embed A into $\mathbb{F}_p$, a field with p^α elements.

It remains to discuss the situation if a sequence $f(x) \in \mathcal{R}(x)$, where $\mathcal{R}$ is a finite commutative ring (by the above embedding argument that is no restriction), is k-automatic and k is not a prime number. In [37] it is shown that a similar result can be achieved, see also [17, 28, 48, 49] for related results.

Theorem 5.3 *If $f(x) \in \mathcal{R}(x)$, where $\mathcal{R}$ is a commutative ring, is k-automatic, then $f(x)$ satisfies a Mahler equation, i.e.,*

$$p_0(x)f(x) + p_1(x^k) + \ldots + p_D(x)f(x^{k^D}) = 0.$$

Unfortunately, as shown by an example in [37] and by a more systematic study in [39], the converse is not true, i.e., there exist Mahler equations which posses non-k-automatic solutions.

6 Automatic sequences in higher dimensions

It is quite natural to generalize the concept of automaticity to higher dimensions, i.e., to sequences which are multi-indexed. In [51, 52], the author considers sequences of the form

$$f(x,y) = \sum_{n,m=0}^{\infty} f_{n,m} x^n y^m,$$

where $f_{n,m} \in A$. The analogous notion of k-decimations are maps $\partial_{k,(i,j)}$, or $\partial_{(i,j)}$ for short, defined as

$$\partial_{(i,j)}(\underline{f}) = \sum_{m,n=0}^{\infty} f_{kn+i,km+j} x^n y^m,$$

where $(i,j) \in \{0,\ldots,k-1\}^2$. The k-kernel of a sequence is defined in the same way as for one-dimensional sequences. A double-indexed sequence is called k-automatic if the k-kernel is finite. As for one-dimensional sequences we have the same characterizations of k-automatic sequences, i.e., a k-automatic sequence is generated by a finite automaton (the labels assigned to the arrows are of the form (i,j)), it is a fixed point of a substitution (a letter is replaced by a $k \times k$-square of letters), and a sequence $f(x,y) \in \mathbb{F}_p(x,y)$ is p-automatic is p-automatic if and only if $f(x,y)$ satisfies a Mahler equation (replace the x by x,y in Theorem 5.2).

Of course, all results mentioned so far readily generalize to multi-indexed sequences $(f_{n_1,n_2,\ldots,n_d})$ with values in a finite set. The same holds for Theorem 5.3.

In certain applications, see the next section, it may be necessary to study bi-infinite sequences, i.e., $(f_l)_{l\in\mathbb{Z}}$ with values in a finite set, or even sequences of the form $(f_{(n)})_{(n)\in\mathbb{Z}^d}$. In [37] a general framework is developed to deal with sequences of this type.

By Cobham's Theorem, we know that the only one-dimensional sequences which are k-automatic for all k are the ultimately periodic sequences. In [19, 20], one can find a discussion of the situation for multi-indexed sequences. Using tools from the theory of first order logic they present the celebrated Cobham-Semenov-Theorem, which gives a complete description of multi-indexed sequences that are k-automatic for all k. In [39] multi-indexed sequences with values in a finite ring and which are k-automatic for all k where characterized in complete algebraic terms.

Besides moving into higher dimensions there are other kinds of generalizations of automaticity, like k-regular sequences [2, 17, 53] or (D,b)-sequences [7].

7 Applications of automatic sequences

In this section we touch upon same applications of automatic sequences.

7.1 Number tables

One well known number table is given by the binomial coefficients, i.e., the double sequence $\left(\binom{n}{m}\right)_{n,m\geq 0}$, and it has been observed by several authors, see [18, 54], that the tables reduced modulo a number l exhibit certain patterns. The proper framework for a description of these patterns is the notion of automatic double sequences. In [4, 33] the ob-

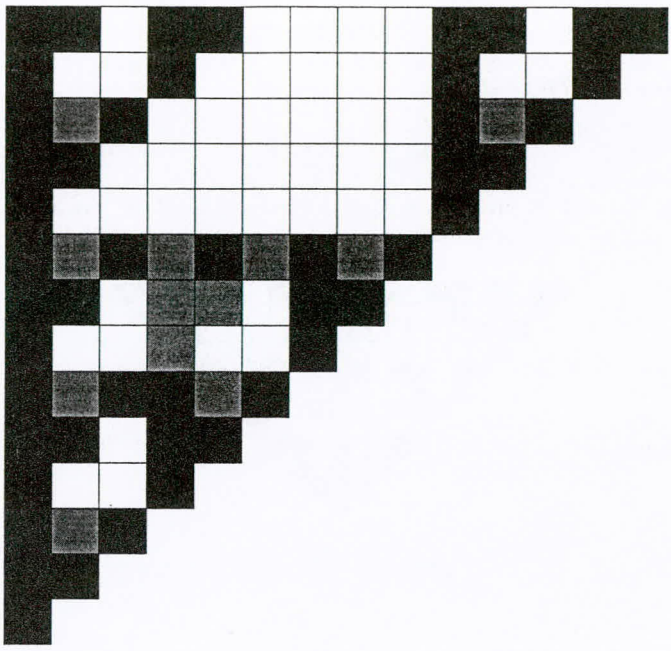


Figure 3: Binomial coefficients mod 3 (white= 0, black= 1, grey= 2)

served patterns for the binomial coefficients are classified using the notion of automaticity. The main result is

Theorem 7.1 *The double sequence* $\left(\binom{n}{m} \bmod l\right)_{n,m\geq 0}$ *is k-automatic if and only if l and k are powers of the same prime number.*

Similar results for other number tables like Gaussian and Sterling numbers may be found in [6, 42, 43]. It is worthwhile to mention that these patterns lead to fractal objects, e.g., [29, 31], which can be described as attractors of a hierarchical iterated function system, [47], or as a graph directed construction, [10, 44].

7.2 Automatic sequences and cellular automata

Cellular automata, see [60], are discrete dynamical systems. A one dimensional cellular is a map that assigns to a bi-infinite string of cells, where each cell has states in a fixed set A, a new string in such the way that the new entry of the cell at position i is completely determined by the states of finitely many neighbors.

We denote the set of bi-infinite sequences with values in A by $A_{\pm\infty}(x)$ and the elements are written as $f(x) = \sum_{n\in\mathbb{Z}} f_n x^n$ or $\underline{f} = (f_n)_{n\in\mathbb{Z}}$. A cellular automaton is a map $C : A_{\pm\infty} \to A_{\pm\infty}$ such that there exists a function (the local rule) $\phi : A^{2M+1} \to A$ with

$$C(\underline{f})(i) = \phi(f_{i-M}, f_{i-M+1}, \ldots, f_{-1}, f_i, f_{i+1}, \ldots f_{i+M})$$

holds for all $i \in \mathbb{Z}$.

Now it becomes clear how automaticity questions enter the stage. Given an initial configuration $\underline{f}_0 \in A_{\pm\infty}$ and a cellular automaton C, we consider the double sequence, the *orbit* of $\underline{f}$ (w.r.t. C, $O(\underline{f}) = (u_{n,t})_{n\in\mathbb{Z},t\geq 0}$ defined as

$$u_{n,t} = C^t(\underline{f})(n),$$

i.e., the value of the cell at position n after t time steps. That clearly defines a double sequence which can be investigated in terms of automaticity.

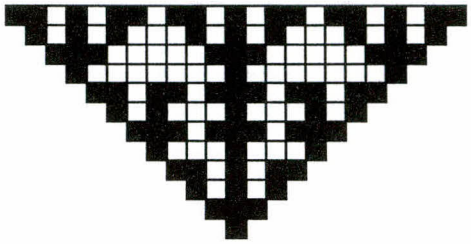


Figure 4: Orbit of a CA

For linear cellular automata, i.e., $A = \mathcal{R}$ is a finite ring and the local rule $\phi : \mathcal{R}^{2M+1} \to \mathcal{R}$ is a linear map, a thorough study of the orbits has been made in [5].

Since the orbits of cellular automata can be considered as special number tables, there is also a connection between orbits of cellular automata and fractals. This connection was observed by [55, 58, 59] and further investigated in [32, 34, 35, 36].

Finally we mention number tables or orbits of linear cellular automata which exhibit a particular pattern, namely being invariant under a certain renormalization operation, the so called coarse graining invariant tables, cf., [11, 12, 13, 14]. They provide further examples of automatic double sequences.

7.3 Automatic sequences and images

As already mentioned in the two subsections above, for certain automatic number tables or orbits of cellular automata one we can associate a (fractal) subset of the plane with the number table. The fractal set itself is generated by a (hierarchical) iterated function system ([10, 15, 15, 44] to name but a few).

It was the author of [15] who used the methods of (hierarchical) iterated function systems as a way for data compression of images. In the same vein is the work presented in [26, 25], where finite automata are used to generate and process images.

References

[1] J.-P. Allouche. Automates finis en théorie des nombres, *Expo. Math.* **5**(1987), 239–266.

[2] J.-P. Allouche, J. Shallit. The ring of k-regular sequences, *Theoret. Comput. Sci.* **98**(1992), 163–197.

[3] J.-P. Allouche, M. Mendès France. Automata and automatic sequences, in F. Axel et al. (eds.), *Beyond Quasicristals*, Les Editions de Physique, Springer, Berlin, 1994, 293–367.

[4] J.-P. Allouche, F. von Haeseler, H.-O. Peitgen, G. Skordev. Linear cellular automata, finite automata and Pascal's triangle, *Discrete Appl. Math.* **66**(1996), 1–22.

[5] J.-P. Allouche, F. von Haeseler, H.-O. Peitgen, A. Petersen, G. Skordev. Automaticity of double sequences generated by one-dimensional linear cellular automata, *Theoret. Comput. Sci.* **188** (1997), 195-209.

[6] J.-P. Allouche, G. Skordev. Schur congruences, Carlitz sequences of polynomials and automaticity, Inst. Dyn. Syst., Universität Bremen, Rep. 399, 1997.

[7] J.-P. Allouche, E. Cateland, W. J. Gilbert, H.-O. Peitgen, J. O. Shallit, G. Skordev. Automatic maps in exotic numeration sequences, *Theory of Computing Systems* **30**(1997), 285–331.

[8] J.-P. Allouche, F. von Haeseler, E. Lange, A. Petersen, G. Skordev. Linear cellular automata and automatic sequences, *Parallel Computing* **23**(1997), 1577-1592.

[9] J.-P. Allouche, J. Shallit. The ubiquitous Prouhet-Thue-Morse sequence, Proceedings of the "International Conference on Sequences and their Applications – SETA '98", Singapore, to appear.

[10] Ch. Bandt. Self-similar sets I. Topological markov chains and mixed self-similar sets, *Math. Nachr.* **142**(1989), 107–123.

[11] A. Barbé, F. von Haeseler, H.-O. Peitgen, G. Skordev. Coarse-graining invariant orbits of one-dimensional two-state linear cellular automata, *Internat. J. Bifurcation and Chaos* **5**(1995), 1611–1631.

[12] A. Barbé. Coarse-graining invariant orbits of one-dimensional $\mathbb{Z}_p$-linear cellular automata, *Internat. J. Bifurcation and Chaos* **6**(1996), 2237–2297.

[13] A. Barbé. Complex order from disorder and from simple order in coarse-graining invariant orbits of certain two-dimensional linear cellular automata, *Internat. J. Bifurcation and Chaos* **7**(1997), to appear.

[14] A. Barbé, G. Skordev. Decimation invariant sequences, SISTA/COSIC report nr. 97-106, Katholieke Universiteit Leuven, 1997.

[15] M. Barnsley. *Fractals everywhere*, Boston MA, Academic Press Inc., 1988.

[16] L. Baum, M. Sweet. Continued fractions of algebraic power series in characteristic 2, *Ann. Math.* **107**(1976), 593–610.

[17] P.-G. Becker. k-regular power series and Mahler-type functional equations. *J. Number Theory* **49**(1994), 269–286.

[18] B. Bondarenko, *Generalized Triangles and Pyramids of Pascal, Their Fractals, Graphs and Applications*, Tashkend, Fan, 1990 (in russian).

[19] V. Bruyère, G. Hansel, C. Michaux, R. Villemaire. Logic and p-recognizable sets of integers, em Bull. Belg. Math. Soc. **2**(1994), 191–238.

[20] V. Bruyère, G. Hansel, C. Michaux, R. Villemaire. Correction to: Logic and p-recognizable sets of integers, em Bull. Belg. Math. Soc. **4**(1994), 577.

[21] G. Christol. Ensembles presque périodiques k-reconnaissables, *Theoret. Comput. Sci.* **9**(1979), 141–145.

[22] G. Christol, T. Kamae, M. Mendès France, G. Rauzy. Suites algébriques, automates et substitutions, *Bull. Soc. Math. France* **108**(1980), 401–419.

[23] A. Cobham. On the base-dependence of sets of numbers recognizable by finite automata, *Math. Systems Theory* **3**(1969), 186–192.

[24] A. Cobham. Uniform tag sequences, *Math. Systems Theory* **6**(1972), 164–192.

[25] K. Culik II, S. Dube. Automata-theoretic techniques for image generation and compression, in *Foundations of software technology and theoretical computer science*, Lect. Notes Comput. Sci. 472, 1990, 76–90.

[26] K. Culik II, I. Fris. Weighted finite transducers in image processing, *Discrete Appl. Math.* **58**(1995), No.3, 223–237.

[27] F.M. Dekking, M. Mendès France, A. van der Poorten. FOLDS!, *Math. Intell.* **4**(1982), 130–138, 173–181 and 190–195.

[28] P. Dumas. Algebraic aspects of B-regular sequences, in A. Lingas, R. Karlsson and S. Carlsson (eds), *Automata, Languages and Programming*, Lect. Notes Comput. Sci., Springer-Verlag, 1993, 457–468.

[29] G. Edgar. *Measure, Topology and Fractal Geometry*, Springer-Verlag, New York, 1990.

[30] S. Eilenberg. *Automata, Languages and Machines*, Vol.A, Academic Press, New York, 1985.

[31] K. Falconer. *The Geometry of Fractal Sets*, Cambridge Univ. Press, New York, 1985.

[32] F. von Haeseler, H.-O. Peitgen, G. Skordev. On the fractal structure of rescaled evolution sets of cellular automata and attractors of dynamical systems, Inst. Dyn. Syst., Univ. Bremen, Rep. **278**, 1992.

[33] F. von Haeseler, H.-O. Peitgen, G. Skordev. Pascal triangle, dynamical systems and attractors, *Erg. Th. Dyn. Syst.* **12**(1992), 479–486.

[34] F. von Haeseler, H.-O. Peitgen, G. Skordev. Cellular automata, matrix substitutions and fractals, *Ann. Math. and Art. Intell.* **8** (1993), 345–362.

[35] F. von Haeseler, H.-O. Peitgen, G. Skordev. Multifractal decomposition of rescaled evolution sets of equivariant cellular automata,*Random and Comput. Dynamics* **3**(1995), 93–119.

[36] F. von Haeseler, H.-O. Peitgen, G. Skordev. Global analysis of self-similarity features of cellular automata: selected examples, *Physica D* **86**(1995), 64–80.

[37] F. von Haeseler. On algebraic properties of sequences generated by substitutions over a group, Habilitationsschrift, Universität Bremen, 1995.

[38] F. von Haeseler. A solution of the periodicity problem for automatic sequences, Inst. Dyn. Syst., Universität Bremen, Rep. 416, 1997, *Journal of Automata, Languages, and Combinatorics* to appear.

[39] F. von Haeseler, A. Petersen. Automaticity of Rational Functions, *Beiträge zur Algebra und Geometrie* **39**(1998), No. 1, 205-215.

[40] F. von Haeseler, W. Jürgensen. Automaticity of Solutions of Mahler equations, Universität Bielefeld, Forschungsschwerpunkt Mathematisierung – Strukturbildungsprozesse, Preprint 127, 1998. Proceedings of the "International Conference on Sequences and their Applications – SETA '98", Singapore, to appear.

[41] F. von Haeseler. *Aspects of automatic sequences*, de Gruyter, Berlin, to appear.

[42] F. von Haeseler, H.-O. Peitgen, G. Skordev. On the fractal structure of rescaled evolution set of Carlitz sequences of polynomials, *Discrete Appl. Math.*, to appear.

[43] E. Lange, H.-O. Peitgen, G. Skordev. Fractal patterns in Gaussian and Stirling number tables, *Ars Combinatoria* **48**(1998), 3–26.

[44] R. Mauldin, S. Williams. Hausdorff dimension in graph directed construction, *Trans. Amer. Math. Soc.* **309** (1989), 811–829.

[45] M. Mendès France. Paper folding, space-filling curves and Rudin-Shapiro sequences, *Contemporary Mathematics* **9**(1982), 85–95.

[46] M. Mendès France, A. van der Poorten. Arithmetic and analytic properties of paperfolding sequences, *Bull. Austr. Math. Soc.* **24**(1981), 123–131.

[47] H.-O. Peitgen, H. Jürgens, D. Saupe. *Chaos and Fractals*, Springer-Verlag, New York, 1992.

[48] A.J. van der Poorten. Substitution automata, functional equations and functions algebraic over a finite field, *Contemp. Math.* **9**(1982), 307–312.

[49] B. Randé. Équations fonctionelles des Mahler et fonctions élémentaires, Thèse, Univeristé de Bordeaux, 1992.

[50] A. Rodenhausen. Paperfolding, generalized Rudin-Shapiro sequences, and the Thue-Morse sequence, *Fractals* **3(4)**(1995), 679–688.

[51] O. Salon. Suites automatiques á multi-indices, *Séminaire de Théorie des nombres* Bordeaux, Exp. 4 (1986–1987), 4-01 – 4-27.

[52] O. Salon. Suites automatiques á multi-indices et algébricité, *C. R. Acad. Sci. Paris, Série I* **305**(1987), 501–504.

[53] J. Shallit. A generalization of automatic sequences, *Theoret. Comput. Sci.* **61**(1988), 1–16.

[54] M. Sved, J. Pitman. Divisibility of binomial coefficients by prime powers a geometrical approach, *Ars Combinatoria* **26A**(1988), 197–222.

[55] S. Takahashi. Self-similarity of linear cellular automata,*J. Comput. Sci.* **44**(1992), 114-140.

[56] A. Thue. Über unendliche Zeichenreihen, *Norske vid. Selsk. Skr. I. Mat. Kl. Chritiana* **7**(1906), 1–22.

[57] A. Thue. Lage gleicher Teile gewisser Zeichenreihen, *Norske vid. Selsk. Skr. I. Mat. Kl. Chritiana* **1**(1912), 1–67.

[58] S. Willson. Cellular automata can generate fractals,*Discr. Appl. Math.* **8**(1984), 91–99.

[59] S. Willson. Calculating growth rates and moments for additive cellular automata,*Discr. Appl. Math.* **35**(1992), 47–65.

[60] S. Wolfram. *Theory and application of cellular automata*, World Scientific Publ. Co. Pte. Lid., Singapore, 1996.

NONASYMPTOTIC LOWER BOUND ON THE MAXIMAL CUMULATIVE REDUNDANCY OF UNIVERSAL CODING

Vladimir B. Balakirsky and Frans M. J. Willems

Eindhoven University of Technology, Electrical Engineering Department
P.O.Box 513, 5600 MB Eindhoven, the Netherlands
v.b.balakirsky,f.m.j.willems@ele.tue.nl

We address the problem of estimating the maximal cumulative redundancy of universal coding and present a nonasymptotic lower bound for the Bernoulli sources with the unknown parameter that can be used for different assignments of the length distributions of both the messages and codewords; this includes the variable-to-variable length coding. The approach can be directly generalized to the other source models.

INTRODUCTION, NOTATION, AND STATEMENT OF THE PROBLEM

Let us consider *a Bernoulli source* with the parameter $\theta \in (0,1)$, i.e., the probability that the source generates a binary sequence u of length $a(u)+b(u)$ consisting of $a(u)$ zeroes and $b(u)$ ones is equal to

$$P_\theta(u) = (1-\theta)^{a(u)}\theta^{b(u)}. \tag{1}$$

Suppose also that $\mathcal{S} \subset \{0,1\}^*$ is a finite set of binary sequences chosen in such a way that $\{P_\theta(u), u \in \mathcal{S}\}$ is the probability distribution on $\mathcal{S}$ for any $\theta \in (0,1)$,

$$\sum_{u\in\mathcal{S}} P_\theta(u) = 1. \tag{2}$$

The set $\mathcal{S}$ will be referred to as *the source segment set*, and we represent it as a union of pairwise disjoint subsets

$$\mathcal{S}(a,b) = \{\, u \in \mathcal{S} :\ a(u) = a,\ b(u) = b \,\},\quad a,b \geq 0. \tag{3}$$

Then (1)–(3) imply

$$\sum_{a,b} |\mathcal{S}(a,b)|(1-\theta)^a\theta^b = 1. \tag{4}$$

For any *encoding* $f : \mathcal{S} \to \{0,1\}^*$ such that

$$\sum_{u \in \mathcal{S}} 2^{-|f(u)|} \leq 1 \tag{5}$$

where $|f(u)|$ denotes the length of the binary sequence $f(u)$, let us introduce the function

$$\overline{\varrho}_\theta(f, \mathcal{S}) = \sum_{u \in \mathcal{S}} P_\theta(u) \cdot |f(u)| - \sum_{u \in \mathcal{S}} P_\theta(u) \log \frac{1}{P_\theta(u)} \tag{6}$$

which is known as *the cumulative redundancy* of coding the sequences belonging to the source segment set ("coding" is understood as a pair (encoding, decoding) where the decoding is assumed to be noiseless). The function $\overline{\varrho}_\theta(f, \mathcal{S})$ is non–negative, as it follows from the Shannon's coding theorem, and can serve as a measure of the coding performance. There are several explicit constructions of codes for the case when the parameter θ is known. In particular, one can assign the Huffman or arithmetic code and get the redundancy which is not greater to 1 and 2 bits, respectively. The case when f should be assigned without knowing the θ is regarded as *universal coding*. Then a proper measure of the performance is the maximal value of the function $\overline{\varrho}_\theta(f, \mathcal{S})$ for all $\theta \in (0,1)$ distinguished as *the maximal cumulative redundancy of universal coding*,

$$\overline{\varrho}(f, \mathcal{S}) = \max_{0<\theta<1} \overline{\varrho}_\theta(f, \mathcal{S}). \tag{7}$$

One of the explicit constructions of universal codes is based on the context tree weighting method [5] which guarantees the cumulative redundancy not greater to the function depending on the average source segment length,

$$L(\mathcal{S}) = \sum_{u \in \mathcal{S}} P_\theta(u) \cdot |u|$$

as $\log \sqrt{L(\mathcal{S})}$. This means that $\overline{\varrho}(f, \mathcal{S})$ is not bounded from above by a constant like for non–universal codes. It is also known [2] that the cumulative redundancy of any universal code of a fixed length (the fixed-to-variable length coding) is not less than a function proportional to $\log \sqrt{L(\mathcal{S})}$. However, as it follows from the arguments used in the proof, this lower bound can be applied only in the asymptotics when $L(\mathcal{S})$ is very big and, for the practical cases, we could only say that the maximal redundancy is non–negative. Another important class of source coding problems concerns different assignments of the length distributions

of the sequences belonging to the source segment set and the code sequences, and one usually distinguishes between the fixed-to-variable, variable-to-fixed, and variable-to-variable length coding. Therefore we address the problem : *derive a nonasymptotic lower bound on the maximal cumulative redundancy of universal coding that can be used for any source segment set satisfying (2) and any encoding function satisfying (5).*

THE KRICHEVSKY–TROFIMOV ESTIMATOR AND ITS PROPERTIES

The universal coding technique developed in [3], [4], [5] and other papers essentially uses the properties of the function introduced in [1] and known as *the Krichevsky–Trofimov estimator*,

$$P_e(a,b) = \int_0^1 \frac{(1-\theta)^a \theta^b}{\pi\sqrt{(1-\theta)\theta}}\, d\theta, \quad \text{for all } a,b \geq 0. \tag{8}$$

The function $P_e(a,b), a,b \geq 0$, has the following remarkable properties :

(**P0**) $P_e(0,0) = 1$;

(**P1**) $P_e(a,b) = P_e(a+1,b) + P_e(a,b+1)$, *for all* $a,b \geq 0$;

(**P2**) *for all* $a,b \geq 0$ *such that* $a+b>0$, *the value of* $P_e(a,b)$ *can be recurrently computed based on* (**P0**) *and equations*

$$P_e(a+1,b) = \frac{a+1/2}{a+b+1} P_e(a,b), \quad P_e(a,b+1) = \frac{b+1/2}{a+b+1} P_e(a,b);$$

(**P3**) *for all* $a,b \geq 0$ *such that* $a+b>0$,

$$P_e(a,b) = \frac{\prod_{i=1}^{a}(i-1/2)\prod_{j=1}^{b}(j-1/2)}{(a+b)!}$$

where the product is understood as 1 when the lower limit is greater to the upper one and

$$\frac{1}{2}\cdot\frac{1}{\sqrt{a+b}} \leq \frac{P_e(a,b)}{\left(\frac{a}{a+b}\right)^a \left(\frac{b}{a+b}\right)^b} \leq \sqrt{\frac{2}{\pi}}\cdot\frac{1}{\sqrt{a+b}}. \tag{9}$$

Furthermore, using (4), (8), and (**P0**) we have

$$\sum_{u\in\mathcal{S}} P_e(a(u),b(u)) = \sum_{a,b} |\mathcal{S}(a,b)| P_e(a,b)$$

$$= \int_0^1 \frac{1}{\pi\sqrt{(1-\theta)\theta}} \left[\sum_{a,b} |\mathcal{S}(a,b)|(1-\theta)^a\theta^b \right] d\theta = 1 \qquad (10)$$

i.e., $P_e = \{ P_e(a(u), b(u)),\ u \in \mathcal{S} \}$ *is the probability distribution on the source segment set.*

The upper bound on the cumulative redundancy proved in [5] can be presented as

$$\exists f^* : \ \overline{\varrho}_\theta(f^*, \mathcal{S}) \leq 1 + \sum_{a,b} |\mathcal{S}(a,b)|(1-\theta)^a\theta^b \log \frac{(1-\theta)^a\theta^b}{P_e(a,b)} \qquad (11)$$

$$\leq 2 + \sum_{a,b} |\mathcal{S}(a,b)|(1-\theta)^a\theta^b \log\sqrt{a+b}. \qquad (12)$$

The term 1 at the right-hand side of (11) comes from the assumption that we use the Huffman code for the sequences included in the source segment set and P_e is the probability distribution on this set, and the sum on a, b in (11) can be obtained using the approach developed in the proof of Theorem 2 in [5] since the coincidence of the lengths of the code sequences is not used in the analysis. The inequality (12) follows from (9).

We try to derive a converse statement to the method of proving the inequality (12) rather than to the final result and conclude with the bound

$$\overline{\varrho}(f, \mathcal{S}) \geq \int_0^1 \frac{1}{\pi\sqrt{1-\theta)\theta}} \left[\sum_{a,b} |\mathcal{S}(a,b)|(1-\theta)^a\theta^b \log \frac{(1-\theta)^a\theta^b}{P_e(a,b)} \right] d\theta$$

for any source segment set $\mathcal{S}$ and encoding f satisfying (2) and (5). This inequality is proved in the next section.

LOWER BOUND ON THE MAXIMAL CUMULATIVE REDUNDANCY OF UNIVERSAL CODING

Theorem : *If (2) and (5) holds, then the maximal cumulative redundancy of universal coding defined by (6), (7) satisfies the inequality*

$$\overline{\varrho}(f, \mathcal{S}) \geq H(P_e(\mathcal{S})) - H_e(\mathcal{S}) \qquad (13)$$

where

$$H(P_e(\mathcal{S})) = \sum_{a,b} |\mathcal{S}(a,b)| P_e(a,b) \log \frac{1}{P_e(a,b)} \qquad (14)$$

and

$$H_e(\mathcal{S}) = \sum_{a,b} |\mathcal{S}(a,b)| \int_0^1 \frac{(1-\theta)^a \theta^b}{\pi\sqrt{(1-\theta)\theta}} \left[a \log \frac{1}{1-\theta} + b \log \frac{1}{\theta} \right] d\theta. \qquad (15)$$

Proof : Let us upper-bound the maximal cumulative redundancy by the expected value of $\overline{\varrho}_\theta(f, \mathcal{S})$ over the ensemble specified by the probability density function $\pi^{-1}(1-\theta)^{-1/2}\theta^{-1/2}, \theta \in (0,1)$, i.e.,

$$\overline{\varrho}(f, \mathcal{S}) \geq \int_0^1 \frac{1}{\pi\sqrt{(1-\theta)\theta}} \overline{\varrho}_\theta(f, \mathcal{S})\, d\theta. \qquad (16)$$

Let us denote

$$P(u) = \frac{2^{-|f(u)|}}{\sum_{u' \in \mathcal{S}} 2^{-|f(u')|}}, \quad \text{for all } u \in \mathcal{S}.$$

Then

$$\overline{\varrho}_\theta(f, \mathcal{S}) = \sum_{u \in \mathcal{S}} P_\theta(u) \log \frac{P_\theta(u)}{P(u)} - \log \sum_{u \in \mathcal{S}} 2^{-|f(u)|} \geq \sum_{u \in \mathcal{S}} P_\theta(u) \log \frac{P_\theta(u)}{P(u)}$$

where the inequality follows from (5). Therefore we can continue (16) as

$$\begin{aligned}
\overline{\varrho}(f, \mathcal{S}) &\geq \int_0^1 \frac{1}{\pi\sqrt{(1-\theta)\theta}} \left[\sum_{u \in \mathcal{S}} P_\theta(u) \log \frac{P_\theta(u)}{P(u)} \right] d\theta \\
&= \sum_{u \in \mathcal{S}} \int_0^1 \frac{P_\theta(u)}{\pi\sqrt{(1-\theta)\theta}} \left[\log \frac{P_\theta(u)}{P(u)} + \log \frac{P_e(a(u), b(u))}{P_e(a(u), b(u))} \right] d\theta \\
&= \sum_{u \in \mathcal{S}} \int_0^1 \frac{P_\theta(u)}{\pi\sqrt{(1-\theta)\theta}} \log \frac{P_\theta(u)}{P_e(a(u), b(u))}\, d\theta \\
&\quad + \sum_{u \in \mathcal{S}} \int_0^1 \frac{P_\theta(u)}{\pi\sqrt{(1-\theta)\theta}} \log \frac{P_e(a(u), b(u))}{P(u)}\, d\theta \\
&= \sum_{a,b} |\mathcal{S}(a,b)| \int_0^1 \frac{(1-\theta)^a \theta^b}{\pi\sqrt{(1-\theta)\theta}} \log \frac{(1-\theta)^a \theta^b}{P_e(a,b)}\, d\theta \\
&\quad + \sum_{u \in \mathcal{S}} P_e(a(u), b(u)) \log \frac{P_e(a(u), b(u))}{P(u)} \\
&\geq \sum_{a,b} |\mathcal{S}(a,b)| \int_0^1 \frac{(1-\theta)^a \theta^b}{\pi\sqrt{(1-\theta)\theta}} \log \frac{(1-\theta)^a \theta^b}{P_e(a,b)}\, d\theta
\end{aligned}$$

where the inequality follows from (10) and the non–negativity of the divergence, and obtain (13) using notations (14), (15). Q.E.D.

Note that the inequality (13) can be also presented in such a way that the maximal cumulative redundancy of universal coding is not less than the mutual information between the continuous random variable θ distributed over the interval $(0,1)$ in accordance with the probability density function $\pi^{-1}(1-\theta)^{-1/2}\theta^{-1/2}, \theta \in (0,1)$, and the discrete random variable $u \in \mathcal{S}$. The transition probabilities are defined as $(1-\theta)^{a(u)}\theta^{b(u)}$ for all $\theta \in (0,1)$ and $u \in \mathcal{S}$.

NUMERICAL RESULTS FOR FIXED-TO-VARIABLE CODING

Suppose that the source segment set is specified as the set $\mathcal{S}^{(n)}$ consisting of all binary sequences of length n. Then

$$|\mathcal{S}^{(n)}(a, n-a)| = \binom{n}{a}, \quad \text{for all} \quad a = 0, \ldots, n$$

and $\mathcal{S}^{(n)}(a,b) = \emptyset$ for all a, b such that $a + b \neq n$. In this case, the upper bound (12) can be expressed as

$$\exists f^* : \ \overline{\varrho}_\theta(f^*, \mathcal{S}^{(n)}) \leq 2 + \log\sqrt{n}. \tag{17}$$

Since

$$\sum_{a,b} |\mathcal{S}^{(n)}(a,b)|(1-\theta)^a\theta^b \left[a \log \frac{1}{1-\theta} + b \log \frac{1}{\theta} \right] = nh(\theta)$$

where $h(\theta) = -(1-\theta)\log(1-\theta) - \theta\log\theta$ is the binary entropy function, we also have

$$H_e(\mathcal{S}^{(n)}) = nh_e$$

where

$$h_e = \int_0^1 \frac{h(\theta)}{\pi\sqrt{(1-\theta)\theta}}\, d\theta.$$

Thus, the bound (13) can be presented as

$$\begin{aligned} \overline{\varrho}(f, \mathcal{S}^{(n)}) &\geq \sum_{a=0}^{n} \int_0^1 \left[\omega_\theta^{(n)}(a) \cdot D_\theta^{(n)}(a) \right] d\theta \\ &= -\sum_{a=0}^{n} \binom{n}{a} P_e(a, n-a) \log P_e(a, n-a) - nh_e \end{aligned} \tag{18}$$

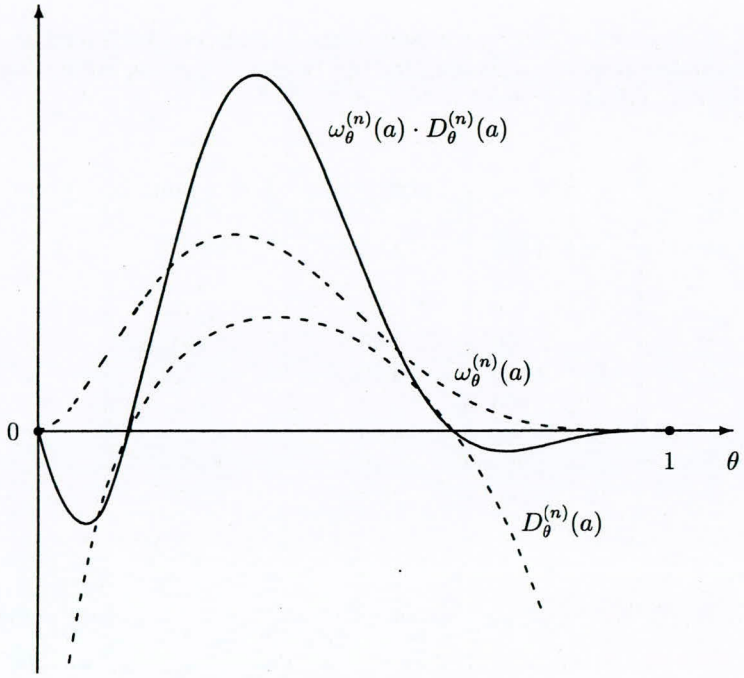


Figure 1: The functions $\omega_\theta^{(n)}(a), D_\theta^{(n)}(a)$, and their product contributing to the lower bound (18) for $a = 5$ and $n = 8$.

where

$$\omega_\theta^{(n)}(a) = \binom{n}{a} \cdot \frac{(1-\theta)^a \theta^{n-a}}{\pi\sqrt{(1-\theta)\theta}}, \quad D_\theta^{(n)}(a) = \log \frac{(1-\theta)^a \theta^{n-a}}{P_e(a, n-a)}.$$

The functions $\omega_\theta^{(n)}(a), D_\theta^{(n)}(a)$, and their product contributing to the lower bound (18) for $a = 5$ and $n = 8$ are shown in Figure 1 and some numerical results obtained from the bounds (17), (18) are given in Table 1.

Table 1: The values of the upper and lower bounds on the maximal cumulative redundancy of universal coding for the Bernoulli sources generating binary sequences of length $n = 2^k$ for $k = 1, \ldots, 9$.

n	$\log \sqrt{n}$	Upper bound	Lower bound
2	0.500	2.500	0.695
4	1.000	3.000	1.014
8	1.500	3.500	1.385
16	2.000	4.000	1.791
32	2.500	4.500	2.218
64	3.000	5.000	2.651
128	3.500	5.500	3.072
256	4.000	6.000	3.452
512	4.500	6.500	3.737

REFERENCES

[1] R.E. Krichevsky and V.K. Trofimov, "The Performance of Universal Encoding", IEEE Trans. Inform. Theory, IT–27(2), pp. 199–207, 1981.

[2] J. Rissanen, "Universal Coding, Information, Prediction, and Estimation", IEEE Trans. Inform. Theory, IT–30(4), pp. 629–636, 1984.

[3] Yu.M. Shtarkov, "Universal Sequential Coding of Single Messages", Probl. Inform. Transm., 23(3), pp. 3–17, 1987.

[4] Tj.J. Tjalkens and F.M.J. Willems, "A Universal Variable-to-Fixed Length Source Code Based on Lawrence's Algorithm", IEEE Trans. Inform. Theory, IT–38(2), pp. 247–253, 1992.

[5] F.M.J. Willems, Yu.M. Shtarkov, and Tj.J. Tjalkens, "The Context-Tree Weighting Method : Basic Properties", IEEE Trans. Inform. Theory, IT–41(3), pp. 653–664, 1995.

COMPLEXITY REDUCING TECHNIQUES FOR THE CTW ALGORITHM

Paul A.J. Volf, Frans M.J. Willems and Tjalling J. Tjalkens

Technische Universiteit Eindhoven,
Den Dolech 2, 5600 MB Eindhoven, The Netherlands
{paul,frans,tjalling}@ei.ele.tue.nl

In [3] Willems, Shtarkov and Tjalkens presented the context-tree weighting algorithm (CTW algorithm). Since then it has been the topic for many research projects. In [5] a first step towards a practical implementation has been made. This paper discusses several other implementation details concerning the CTW algorithm. The computational work has been improved, the memory requirements have been significantly reduced, and finally, a construction of a forward decomposition [2] is presented that reduces the complexity further, while it improves the performance at the same time.

INTRODUCTION

The context-tree weighting algorithm is a universal source coding algorithm. It computes a conditional weighted probability, $P_w^{\lambda}(x_{n+1}|x_1^n)$, for every symbol x_{n+1} of the source sequence. An arithmetic encoder can then be used to encode the source sequence with these probabilities. The CTW algorithm has become a well-known universal source coding algorithm in the information theory field over the past few years. But in practice it has not been fully accepted yet as a good alternative for more popular algorithms like PPM, while it can match PPM's performance. One possible reason can be that most previous implementations of the CTW algorithm were rather complex: these experimental programs could use more than 100 MB of memory and compress only a few kilobyte per second on fast workstations. In [5] a first step was taken to reduce the complexity of the CTW algorithm considerably. Here we will explain some more modifications that will lead to a new and efficient implementation. First we discuss a technique that reduces the number of mathematical operations further. Next, a hash table implementation will be investigated that reduces the memory usage of the CTW algorithm. Finally, a construction of a forward decomposition will be explained that reduces the computational and memory complexity even further.

COMPUTATIONS

The research described in [5] was primarily concerned with the reduction of the number of computations in each node. It also introduced a β-factor, which replaced both the estimated probability and the weighted probability that normally would be stored in a node. Furthermore, it introduced a fixed-point log-implementation for the floating point numbers. We will use all of these techniques, except for one modification. In a node s not at maximum depth of the context tree, the conditional weighted probability for x_{n+1} is now computed with the following steps:

1. Compute $\beta^{s\prime}(x_1^n) = \frac{\beta^s(x_1^n)}{a_s+b_s+1}$, and the terms:

$$[\beta^s(x_1^n)P_e^s(0|x_1^n)] = \beta^{s\prime}(x_1^n)(a_s + 1/2),$$
$$[\beta^s(x_1^n)P_e^s(1|x_1^n)] = \beta^{s\prime}(x_1^n)(b_s + 1/2).$$

2. Compute the intermediate results with the previous results:

$$P_i^s(x_1^n, 0) = [\beta^s(x_1^n) \cdot P_e^s(0|x_1^n)] + P_w^{t's}(0|x_1^n),$$
$$P_i^s(x_1^n, 1) = [\beta^s(x_1^n) \cdot P_e^s(1|x_1^n)] + P_w^{t's}(1|x_1^n).$$

3. Now, compute the conditional weighted probabilities (and normalize at the same time):

$$P_w^s(0|x_1^n) = \frac{P_i^s(x_1^n, 0)}{P_i^s(x_1^n, 0) + P_i^s(x_1^n, 1)},$$
$$P_w^s(1|x_1^n) = \frac{P_i^s(x_1^n, 1)}{P_i^s(x_1^n, 0) + P_i^s(x_1^n, 1)}.$$

4. Update β^s and store the new value:

$$\beta^s(x_1^{n+1}) - \frac{[\beta^s(x_1^n)P_e^s(x_{n+1}|x_1^n)]}{P_w^s(x_{n+1}|x_1^n)}.$$

5. Increment a_s if $x_{n+1} = 0$, or b_s otherwise.

Here t' is the child node of s which is also on the context path and which provides the conditional weighted probability from the next level in the context tree. Note that the decoder can perform steps 4 and 5 only after symbol x_{n+1} has been decoded. These steps use one multiplication, one division and one addition less than the ones described in [5].

MEMORY

The records in a context tree consist of two parts: a CTW part that contains all CTW related information, and a structural part, used to find the record that corresponds to a certain context.

The *CTW information* in the records differs for records representing nodes and records representing leaves. A record that represents a node s needs to store the counts a_s and b_s and the factor β^s. As described in [5], not more than 2 bytes are necessary to store β^s. Furthermore, by applying a count-halving technique, both counts in a node will be halved and rounded up if one of them reaches 256, the counts require 1 byte of storage space each. Thus the CTW information in nodes takes only 4 bytes. In leaves the CTW algorithm needs to store the counts and, if the leaf is not at the maximum depth, some information to extend the context path. Every memory efficient implementation of the CTW algorithm will construct the context paths only until they become unique. When a context s becomes unique (all contexts so far that had s as suffix, are identical), the CTW algorithm can create a leaf for that context, because $P_e^s(.) = P_w^s(.)$. But if a new context occurs that has s as a suffix, and that differs from the unique path, then, in order to achieve a good performance, the old unique path through the leaf has to be extended until it is unique compared to the new context. One solution would be to store the file in a file buffer, and store in each leaf an index to the position of that context in the file. But since we do not want to waste valuable memory resources on a file buffer, we decided to store a fixed number of new context symbols in each leaf. Experiments show that with 3 context symbols stored in each leaf, the performance of the CTW algorithm will not degrade more than 0.22 % on the entire Calgary Corpus. This leaves only one byte for the counts, but fortunately, from simulations we know that almost all leaves have counts smaller than 16. The leaves in which one of the counts is greater than 15 account in general for less than 1.3 % of all counts. Therefore, if we apply the count-halving technique if one of the counts in the leaves reaches 16, then the counts can be stored in 4 bits, and the CTW information of the leaves also fits in 4 bytes. But note that the CTW algorithm needs three additional bits: one bit to be able to distinguish between nodes and leaves, and two bits to specify the number of valid context symbols in each leaf.

Similar to [4], we implemented the context trees in a hash table. The only *structural information* in a record is the key with which it can be identified. Hash

tables can be efficiently searched, and records can be easily added and updated. A disadvantage of hash tables is that their efficiency decreases as soon as the hash table is filled for more than 80 %. The key in every record can be smaller than 2 bytes, if we use the sequential search behaviour of the CTW algorithm. As the CTW algorithm traverses through a context tree it searches in every step for a context that is one symbol longer than the previous context. First we make a table of the indexes of the root nodes of the context trees. Once we know which context tree to use next, we immediately know the index of its root node. Furthermore, we make a table with offsets: every possible source symbol gets a different offset. Now, suppose we already found the record corresponding to context s in a certain context tree. Its child node corresponding to a context $t's$ is created at an index which is at a multiple of the offset of t' from record s. This child node has as key the next context symbol t' (which specifies the offset) and the number of steps that were necessary to reach this record from the record corresponding to its parent node s. This key uniquely specifies the parent node of this record, and the CTW algorithm can find the child node by looking at indexes which are multiples of the offset from the parent node. We will limit the number of steps to 31, thus each record needs 1 byte to store the most recently used context symbol, and 5 bits to store the number of steps.

In total the CTW information (4 bytes plus 3 bits) and the structural information (1 byte plus 5 bits) fit in only 6 bytes.

THE FORWARD DECOMPOSITION

The forward decomposition (see e.g. [2]) determines how the non-binary source symbols are transformed in series of bits. The binary decomposition is a binary tree with one leaf for every possible source symbol. A symbol is encoded by encoding the path from the root node to the leaf corresponding to that symbol. In each node a binary decision has to be made. Each node has its own context tree, counting for every context the number of times each of its two children was chosen. The CTW algorithm uses this context tree to encode the decision. Figure 1 shows an example of a forward decomposition for a 5-ary source.

So far we often use the "ASCII-decomposition". This is a binary tree in which all 256 leaves have depth 8 and the path from the root node to a leaf corresponds to the binary ASCII representation of that symbol. Then a source symbol is encoded by encoding each of the 8 bits of its representation. But any binary tree with sufficient leaves can serve as a binary decomposition. If a symbol

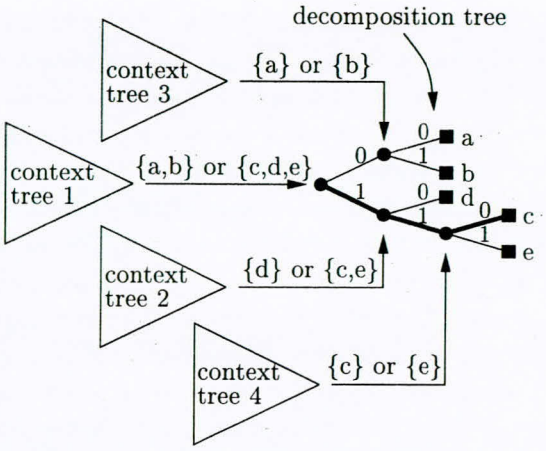


Figure 1: An example of a forward decomposition for a 5-ary source.

occurs often in a certain file, that symbol can be placed at a lower depth in the forward decomposition. As a result fewer nodes have to be visited to encode this symbol, and consequently fewer context trees have to be updated. Since these context trees will be created dynamically, this reduces not only the number of computations, but also the number of created records. Finally, by choosing a good ordering of the source symbols in the leaves of the forward decomposition, the model used by the CTW algorithm to encode a decision could be simplified, which can improve the performance.

The most straightforward solution to construct an efficient forward decomposition is to use the Huffman tree. The Huffman code minimizes the average code word length of a fixed-to-variable length source code, thus if the corresponding Huffman tree is used as a forward decomposition it will minimize the number of visited nodes and consequently the number of updated context trees. Unfortunately, if the Huffman tree constructed from the frequencies of the source symbols is used as forward decomposition, the performance of the CTW algorithm degrades. Furthermore, the Huffman tree has to be encoded, which increases the code word length further. But one does not have to use the Huffman tree constructed from the symbol frequencies to minimize the number of visited nodes. Any tree in which the leaves are at the same depth as in the Huffman tree will minimize number of visited nodes. Thus, instead we will use the Huffman code to

compute the Huffman depths of the symbols only, and then construct a forward decomposition with these depths. This has two advantages. First, we have more freedom in combining the symbols in such a way that the performance improves. Secondly, the forward decomposition can be encoded by encoding the Huffman depth of each symbol. If the depth of the tree is limited to 31 levels, then only 5 bits per source symbol are necessary, or 160 bytes in total.

We implemented various constructions of the forward decomposition, given the Huffman depths of each symbol. One very complex construction also used the file itself. It actually resulted in the best performance, and it resulted in the greatest reduction in the number of created records, but it took sometimes hours to complete its construction process. Of the other, more time efficient, constructions, we now use the construction dubbed "the Huffman-III decomposition", since it results in the best compression performance and at the same time in the greatest reduction of the memory complexity. The Huffman-III construction is very similar to the Huffman construction. It starts with the source symbols with the highest Huffman depth. At any time, when there are two or more symbols left it performs the following operation. From the symbols or super-symbols at the highest depth (there are at least two of these), combine the two symbols with the highest ASCII-value. This creates a new super-symbol at a depth one below the current depth and the "ASCII-value" of this new super-symbol is the lowest ASCII-value of all symbols collected in this super-symbol. By constructing the forward decomposition in this way we try to keep symbols with a similar ASCII-value close to each other in the forward decomposition. In the ASCII-code the lower case letters, the capital letters, and the numbers, are all collected in groups. By trying to keep them together in the forward decomposition we try to get more meaningful groups in the forward decomposition, which may lead to simpler models, and thus to a better performance. Experiments show that this indeed gives a much better performance, and a much lower memory complexity, than scattering the symbols more randomly through the forward decomposition.

RESULTS

The Calgary Corpus is a well-known set of test files. We used this corpus to compare the performance of the Huffman-III decomposition with the more standard ASCII-decomposition. The compression program uses a fixed-point log-implementation and the order of computations as described above. It uses a hash table of 32 MB, which can contain about 5.6 million nodes and leaves. The

Table 1: A comparison between the two decompositions

file	ASCII-dec.			Huffman-III decomposition				
	size	records	time	size	records		time	
	[bytes]	[#]	[sec]	[bytes]	[#]	[%]	[sec]	[%]
bib	24949	643699	12.6	24876	426172	-33.8	9.2	-27.0
book1	207414	5482429	103.8	206194	3354824	-38.8	55.0	-47.0
book2	141784	3566446	69.5	141215	2226230	-37.6	44.0	-36.7
geo	58424	1162423	9.4	56953	795228	-31.6	7.4	-21.3
news	108886	2652234	42.2	107615	1737750	-34.5	24.4	-42.2
obj1	9994	160801	3.7	9837	133538	-17.0	3.1	-16.2
obj2	73648	1394904	21.9	72205	1122018	-19.6	17.2	-21.5
paper1	15030	423125	6.8	14945	265565	-37.2	4.9	-27.9
paper2	22674	676885	11.2	22505	397401	-41.3	6.4	-42.9
pic	53095	778220	43.8	48047	436184	-44.0	11.1	-74.7
progc	11451	300528	5.5	11412	197054	-34.4	4.3	-21.8
progl	14304	344427	8.5	14207	210174	-39.9	5.6	-34.1
progp	10201	235356	6.4	10085	154538	-34.3	4.5	-29.7
trans	16343	340903	10.1	16279	239746	-29.7	7.3	-27.7

context depth is 8 symbols. In the table we compare the performance of the two decompositions. For each file and each decomposition, the resulting length of the compressed file, the total number of used records, and the total used time are given. The time has been measured on a HP-735 workstation over 5 runs to obtain a more accurate result. The speed of this workstation is comparable to a PC with a 200 MHz. Pentium-processor. The compressed file length for the Huffman-III decomposition includes the 160 bytes necessary to encode the decomposition. The time given for the Huffman-III experiments includes the time needed to compute the Huffman depths and to construct the forward decomposition.

It is clear that the Huffman-III decomposition is much better than the ASCII-decomposition. On almost all files (except the binary files), it reduces the number of created records by 30 to 40 %. The total compression time is also reduced by 20 to 40 % on most files. On the smaller files, the speed did not improve much because it has to initiate all 32 MB of memory, and all kinds of tables, which costs an amount of time independent from the size of the file. On the longer files the gain is typically 30 % or more. Two exceptions are the picture file and book1. In the picture file the symbol 0 occurs very often and is placed at depth 1. This results in a tremendous increase in speed. The file book1 uses 5.48 million symbols with the ASCII-decomposition. Then the hash-table is filled for 98 %,

and this could significantly decrease the search speed. With the Huffman-III decomposition it only uses 3.35 million records and the hash table is filled for only 60 %, which will not have a measurable effect on the search-speed. Finally, besides all the reductions in the complexity, the Huffman-III decomposition also has a slightly better performance on every file. It compresses the entire corpus to 756375 bytes, which is about 1.5 % better than the CTW with the ASCII-decomposition which needs 768197 bytes and 1 % better than the state-selection algorithm [1] (a good PPM-algorithm), which needs about 764000 bytes (an exact number is not given).

CONCLUDING REMARKS

This paper discussed various techniques that can be used to design a fast and efficient data compression program based on the CTW algorithm. With the order of computations described here, and the hash-table implementation, it works well with 32 MB on all files of the Calgary corpus. Its speed is typically between 7 and 9 kB per second on our HP-735 workstation. Furthermore, we developed a construction of a forward decomposition, the Huffman-III decomposition, that not only improves the results by 1.5 %, but also reduces the memory complexity by 30 to 40 % and increases the speed by 20 to 40 %. With these modifications we have shown that the CTW algorithm is indeed a useful technique for modern data compression programs.

REFERENCES

[1] Suzanne Bunton. A percolating state selector for suffix-tree context models. In *Proceedings of the DCC*, pages 32–41, Snowbird, Utah, March 25-27 1997.

[2] Paul A.J. Volf. Context-tree weighting for text-sources. In *IEEE Int. Symp. on Inform. Theory*, page 64, Ulm, Germany, June 29 - July 4 1997.

[3] Frans M.J. Willems, Yuri M. Shtarkov, and Tjalling J. Tjalkens. The context-tree weighting method: Basic properties. *IEEE Trans. Inform. Theory*, 41(3):653–664, 1995.

[4] Frans M.J. Willems and Tjalling J. Tjalkens. Complexity reduction of the context-tree weighting algorithm: A study for KPN research. Technical report, EIDMA, Eindhoven, 1997.

[5] Frans M.J. Willems and Tjalling J. Tjalkens. Complexity reduction of the context-tree weighting method. In *Symp. on Inform. Theory in the Benelux*, volume 18, pages 123–130, Veldhoven, The Netherlands, May 15-16 1997.

STOCHASTIC LANGUAGE MODELLING USING CONTEXT TREE WEIGHTING

P. Vanroose

Katholieke Universiteit Leuven, div. E.S.A.T.–PSI
K.Mercierlaan 94, B–3001 Heverlee, Belgium
Peter.Vanroose@esat.kuleuven.ac.be

The Context-Tree Weighting (CTW) algorithm is used to construct a trigram language model for continuous speech recognition. This results in a language model with a lower perplexity as compared to the traditional approach.

INTRODUCTION

Stochastic speech recognition and *data compression* have traditionally been well separated research areas, although both use very similar underlying concepts viz. sequential data streams, symbol probabilities, entropy, ... The similarities become even more apparent if we more specifically focus on *language modelling* and *universal source coding*. In both areas one tries to capture the stochastic characteristics or behaviour of a data source, in order to use this knowledge for further stochastic processing, i.c. speech recognition and data compression, respectively.

STOCHASTIC LANGUAGE MODELLING

A continuous speech recognition system aims at finding the text (i.e., sequence of words) $W := w_1 w_2 w_3 \cdots$ for which the conditional probability $P(W|A)$ is maximal, where A is the acoustic signal.

This problem is split in two parts, using Bayes' rule: Maximising $P(W|A)$ means maximising $P(W) \cdot P(A|W)$, as $P(A)$ is fixed, so it suffices to determine independently the probability $P(A|W)$ of an acoustic signal given the word sequence spoken (*acoustic modelling*), and the probability $P(W)$ of a certain word sequence (*language modelling*). See [7] for a good overview.

The probability $P(W) = P(w_1 w_2 w_3 \cdots)$ can in turn be decomposed into the sequentially obtainable product $P(w_1) \cdot P(w_2|w_1) \cdot P(w_3|w_1 w_2) \cdots$ A natural

assumption here is that text has *finite memory*, i.e., that $P(w_n|w_1w_2\cdots w_{n-1}) = P(w_n|w_{n-D}\cdots w_{n-1})$ for some small integer D.

Hence a speech recognition system relies on a language modelling module which it can ask for the probability of (in principle) any word sequence, or alternatively of any word, given its left context (or history). Stated in source coding terms, a reliable language model for continuous speech recognition should accurately estimate the conditional probability of a word in spoken text, given the foregoing D words.

Currently, the standard approach to this problem consists of the following steps: (1) using a very large text database, estimate all trigram probabilities $P(w_3|w_1w_2)$ (i.e., assuming $D = 2$) for all word triples occurring in the database; (2) prune the resulting probability tree, to make the results less dependent on infrequent word triple probabilities.

In this contribution the Context-Tree Weighting (CTW) algorithm will be applied instead to construct a reliable language model. CTW is a universal source coding (compression) algorithm, and hence is able to estimate source statistics from a given data stream. It has proved to be superior to most other universal source coding techniques, especially for relatively short data streams.

UNIVERSAL SOURCE CODING

Universal source coding aims at compressing the output of a source with unknown symbol generating statistics. Part of the method consists of (implicitly) determining the actual source statistics, cf. [1].

In language modelling, the problem is exactly to determine the source statistics. This suggests that in principle the methods used in one domain can be used in the other domain.

The context tree weighting (CTW) algorithm was introduced in [5, 6]. It is a sequential universal data compression procedure asymptotically achieving the entropy of any (finite-memory tree) source, and (most importantly) also performing very well for realistic (finite) source sequence lengths, and for any source sequence, i.e., not only on the average. On the other hand, the complexity of the CTW algorithm is relatively low as compared to other methods, and it allows to sequentially encode and decode, i.e., in a single pass through the data sequence.

The main idea of CTW is to 'weigh' (average) all applicable models in a certain 'context' (history), not to choose one (best) model. This averaging gives good results even for small data sequences. The source models to be averaged are

so-called *bounded memory tree sources* which can be seen as sources that generate an output sequence which can be partitioned into memoryless subsequences of variable (but bounded) length. This is a generalisation of the concept of FSMX sources, in turn a generalisation of finite-order Markov sources.

Applied to language modelling, where 'source symbols' are words, one typically assumes sentences to be independently generated, as one is mainly interested in n-gram probabilities inside sentences. So tree sources seem appropriate here.

THE KT ESTIMATE FOR MEMORYLESS SOURCES

In [3] universal source coding was analysed, and it was shown that estimating the source statistics by weighting all memoryless sources with the Dirichlet density with parameter $(\frac{1}{2}, \frac{1}{2}, \cdots, \frac{1}{2})$ is optimal for block-to-variable universal source coding, and it guarantees uniform convergence of the parameter redundancy. This so-called Krichevsky-Trofimov (KT) estimator $f_{\mathrm{KT}}(\mathbf{u})$ for the probability of a memoryless source to generate the sequence $\mathbf{u}$ is thus defined as

$$f_{\mathrm{KT}}(\mathbf{u}) := P_{\mathrm{e}}(a_1, \ldots, a_\alpha) := \frac{\Gamma(\alpha/2)}{\pi^{\alpha/2}} \int_\Omega \prod_{j=1}^{\alpha} \theta_j^{a_j - 0.5}\, d\theta_1 \cdots d\theta_\alpha$$

where Γ is the Gamma function, Ω is the set of all probability vectors $(\theta_1, \ldots, \theta_\alpha)$ over the alphabet $\{1, 2, \ldots, \alpha\}$ and a_j is the number of occurrences of the symbol j in the sequence $\mathbf{u}$, i.e., $\mathbf{u}$ is of type $(a_1, \ldots, a_\alpha)$.

Define, for positive integers n, $G(n, x) := \Gamma(x+n)/\Gamma(x) = x \cdot (x+1) \cdots (x+n-1)$. Note that $G(n, x)$ has n factors, and that $G(n+1, x) = (n+x) \cdot G(n, x)$. The KT estimator can be expressed in recursively computable form as (see also [4, 5]):

$$f_{\mathrm{KT}}(\mathbf{u}) = P_{\mathrm{e}}(a_1, \ldots, a_\alpha) = \frac{\prod_{j=1}^{\alpha} G(a_j, \frac{1}{2})}{G(\sum_{j=1}^{\alpha} a_j, \frac{\alpha}{2})}.$$

Otherwise stated, the estimated conditional probability of a memoryless source with unknown parameters generating the symbol $x \in \{1, \ldots, \alpha\}$ after having generated a_1 1's, $\ldots$, a_α symbols α, is

$$P_{\mathrm{est}}(X = x | a_1 \ldots a_\alpha) = \frac{a_x + \frac{1}{2}}{\sum_{j=1}^{\alpha} a_j + \frac{\alpha}{2}}, \tag{1}$$

which means that it suffices to keep track of the numbers $a_j + \frac{1}{2}$ (and their sum) in order to easily calculate estimated next symbol conditional probabilities.

CONTEXT-TREE WEIGHTING

This memoryless KT estimate is not directly useful for sources with memory. But it can still be used when the source is a bounded memory tree source, with upper bound D on the memory. This is where the concept of context weighting comes in.

If we knew the 'model class' of the tree source for which we want to estimate the parameters, and we have seen the sequence $u_1, \ldots, u_n$ as source output, we know for each symbol u_i its context s (i.e., state of the source at that point, which is (part of) $u_{i-D}, \ldots, u_{i-1}$) and we can keep track of the number $a_j^{(s)}$ of symbols $u_i = j$ in context s. If we are in context s at a certain moment, the KT estimator for the next source symbol is obtained from (1) with the numbers a_j substituted by $a_j^{(s)}$.

But since we do not know the model class of the source, and hence we do not know the context s we are in, instead of estimating s CTW suggests to weigh all possible models in obtaining the estimated conditional next symbol probability. This is done by first assigning counts $a_1^{(s)}, \ldots, a_\alpha^{(s)}$ and weighted probabilities Q_s to each node s of the full α-ary tree of depth D, where each node s is represented by the unique α-ary string leading from this node to the root (which itself is represented by the empty string):

$$Q_s := \begin{cases} P_e(a_1^{(s)}, \ldots, a_\alpha^{(s)}) & \text{if } s \text{ is a leaf} \\ \frac{\alpha-1}{\alpha} P_e(a_1^{(s)}, \ldots, a_\alpha^{(s)}) + \frac{1}{\alpha} \prod_{j=1}^{\alpha} Q_{js} & \text{otherwise} \end{cases}$$

where js is the state obtained from s by prepending symbol j to the string s, i.e., the product in the second term is over all child nodes of s. Clearly, $a_i^{(s)} = \sum_{j=1}^{\alpha} a_j^{(js)}$ if s is not a leaf.

The CTW estimated next symbol conditional probability is then given by the quotient of the value Q_ε in the root of the tree with its previous value, before having seen the last symbol.

Note that the context tree of depth D as described above is easily updated at each new input symbol, initially starting with a tree where all counts are zero and all probabilities 1. Actually, tree nodes need only be created when one of their counts becomes nonzero.

BUILDING A TRIGRAM LANGUAGE MODEL

A trigram language model specifies conditional probabilities $P(w_n|w_{n-2}w_{n-1})$, hence $D = 2$. After having built the context tree of depth 2 and calculated Q_ε, the estimated $P_{\text{est}}(w|w''w')$ is the quotient of Q'_ε and Q_ε, where Q'_ε is calculated just like Q_ε but after having incremented $a_w^{(\varepsilon)}$, $a_w^{(w')}$ and $a_w^{(w''w')}$ by one.

For node $s = w''w'$ this implies $Q_s = P_e(s)$ is multiplied by a factor $f'' := (a_w^{(w''w')} + \frac{1}{2})/(\sum_j a_j^{(w''w')} + \frac{\alpha}{2})$.

In node w', P_e is multiplied with $f' := (a_w^{(w')} + \frac{1}{2})/(\sum_j a_j^{(w')} + \frac{\alpha}{2})$, hence

$$Q'_{w'} = Q_{w'} \cdot f'' + \frac{\alpha - 1}{\alpha} P_e(w') \cdot (f' - f'').$$

Finally, with $f := (a_w^{(\varepsilon)} + \frac{1}{2})/(\sum_j a_j^{(\varepsilon)} + \frac{\alpha}{2})$, we can also express Q'_ε in terms of pre-calculated values, and $P_{\text{est}}(w|w''w') =$

$$f'' + \frac{\alpha - 1}{\alpha}(f - f'')\frac{P_e(\varepsilon)}{Q_\varepsilon} + \frac{\alpha - 1}{\alpha}(f' - f'')\frac{P_e(w')}{Q_{w'}} \cdot \left(1 - \frac{\alpha - 1}{\alpha} \cdot \frac{P_e(\varepsilon)}{Q_\varepsilon}\right).$$

Note that, for a given language model, only f, f' and f'' have to be recalculated, and that only the $\alpha + 1$ quotients $P_e(w')/Q_{w'}$ and $P_e(\varepsilon)/Q_\varepsilon$ are needed, not the values of P_e or Q_s themselves.

The first order approximation of $P_{\text{est}}(w|w''w')$ is f'', which is the KT estimate for a memoryless source. When w is a previously unseen word, $f'' = \frac{1}{2}/(\sum_j a_j^{(w''w')} + \frac{\alpha}{2})$ where $a_j^{(w''w')}$ is the number of occurrences of the context $w''w'$ in the training text. Thus, also other occurrences of context $w''w'$ influence the value of the conditional probability. In classical language modelling this is obtained in an ad-hoc way by so-called backing-off, while here it follows in a natural way, without having to specify any tuning parameters.

EXPERIMENTAL RESULTS

Preliminary tests on a large text database show satisfactory results. Perplexity values decrease by about 5% as compared to classical methods to generate a trigram language model. The method is also easily extensible to more general schemes than trigrams, either by simply pruning the context tree where data becomes sparse, or even by using more general definitions of context, like using information about the grammatical function of a word, or breaking the context at sentence boundaries...

Due to the typically large value of α, in the order of magnitude of 10^5, it is generally not possible to build the full context tree of depth 2. Instead, nodes with a very low visit count have to be re-used. This is acceptable, as infrequent words —which thus disappear— are not relevant for the language model. Typically, 60% of the words in a large text database only appear once!

REFERENCES

[1] C.E. Shannon, "Prediction and entropy of printed English", *Bell System Technical Journal* 30, pp. 50–64, 1951.

[2] E.N. Gilbert, "Coding based on inaccurate source probabilities", *IEEE Transactions on Information Theory* IT-17(3), pp. 304–314, 1971.

[3] R.E. Krichevsky, V.K. Trofimov, "The performance of universal encoding", *IEEE Transactions on Information Theory* IT-27(2), pp. 199–207, 1981.

[4] T.J. Tjalkens, Y.M. Shtarkov, F.M.J. Willems, "Context tree weighting: multi-alphabet sources", *Proceedings of the Fourteenth Symposium on Information Theory in the Benelux*, Veldhoven, pp. 128–135, 1993.

[5] F.M.J. Willems, Y.M. Shtarkov, T.J. Tjalkens, "The context-tree weighting method: basic properties", *IEEE Transactions on Information Theory* IT-41(3), pp. 653–664, 1995.

[6] F.M.J. Willems, Y.M. Shtarkov, T.J. Tjalkens, "Context weighting for general finite-context sources", *IEEE Transactions on Information Theory* IT-42(5), pp. 1514–1520, 1996.

[7] M. Federico, R. De Mori, "Language modelling", chapter 7 from *Spoken dialogues with computers*, Renato De Mori, ed., in series "Signal Processing and its Applications", pp. 199–230; Academic Press, 1998.

A FAST FULL SEARCH EQUIVALENT FOR MEAN-SHAPE-GAIN VECTOR QUANTIZERS

J. Cardinal

Brussels Free University, Computer Science Department
Campus Plaine, Bld. du Triomphe, CP212, B1050 Brussels, Belgium
jcardin@ulb.ac.be

Vector quantization is an efficient lossy digital signal compression method. It encodes a k-dimensional vector by a reference in a dictionary – or codebook *– of representative vectors called* codewords. Mean-shape-gain vector quantizers *(MSGVQ) encode separately the mean and the length – or* gain *– of the vector, using two scalar quantizers. The mean-removed, normalized vector is called the* shape, *and is encoded by an index in a shape codebook. Finding the best shape codeword requires a* maximum correlation search *which we show to be equivalent to a nearest-neighbor search on a unit hypersphere. We address this problem, and try to find fast full-search equivalents. An efficient elimination rule based on a a spherical triangular inequality is derived, and combined with the classical partial distance search. Reference vectors for the application of the inequality are the first principal components of the codewords distribution. We also present some experimental results, and show that our technique is faster than some other existing elimination rules.*

INTRODUCTION

A standard vector quantizer (VQ) may be defined as a mapping $M : \mathbb{R}^k \rightarrow \mathcal{C}$, where $\mathcal{C}$ is a finite set of vectors in $\mathbb{R}^k$ called the *codebook*. Each representative vector in $\mathcal{C}$ is called a *codeword*. Vector quantization is a good way of vector data compression used for example in digital image coding, where the vectors to be encoded are the pixel values extracted from image blocks. The codebook is constructed from a set a representative vectors called the *training set* using clustering algorithms like the *generalized Lloyd algorithm* (GLA, [5]). The quantization phase consists in, given an input vector Q, finding the vector $\hat{Q} \in \mathcal{C}$ minimizing the distorsion $d(Q, \hat{Q})$, usually defined as the euclidean distance. This nearest-neighbor search is the bottleneck of most VQ.

We now define the *mean-shape-gain vector quantizer* (MSGVQ, [3], p.451), which is a natural extension of the *shape-gain vector quantizer* (SGVQ, [3], p.441). Let $Q \in \mathbb{R}^k$. We define the *mean* $m = 1/k.\sum_{i=1}^{k} Q_i$, the *gain* $g = \| Q - m.\mathbf{1} \|$, where $\mathbf{1} = (1, \ldots, 1)$ and the *shape* $s = (Q - m.\mathbf{1})/g$. The shape vector is both zero-sum and unitary. Mean-shape-gain vector quantization consists in encoding separately those three parameters using three codebooks. The reconstructed vector has the form :

$$\hat{Q} = \hat{g}.\hat{s} + \hat{m}.\mathbf{1} \ ,$$

where $\hat{g}$, $\hat{m}$ and $\hat{s}$ are taken from three different codebooks, respectively $\mathcal{C}_g$, $\mathcal{C}_m$ (scalar codebooks) and $\mathcal{C}_s$ (vector codebook). The goal of the encoder is to minimize the distorsion $d(Q, \hat{Q}) = \| Q - \hat{Q} \|$. We can show :

$$d(Q, \hat{Q})^2 = g^2 - 2.\hat{g}.Q.\hat{s} + \hat{g}^2 + k(m - \hat{m})^2 \ .$$

Thus the minimization may be carried out in three steps :

- find $\hat{m}$ as close as possible to m,
- find $\hat{s}$ so that $Q.\hat{s}$ is maximal,
- find $\hat{g}$ as close as possible to $Q.\hat{s}$.

We see that the quantization of g is dependent of the choice of $\hat{s}$, which is not the case for m. The algorithm finding the best shape codeword $\hat{s}$ by computing every dot product $Q.\hat{s}$ is called the *maximum correlation search* (see [3], p.442). In this paper, we address the problem of finding a faster way to find the best shape codeword.

In SGVQ, the mean-remove step is skipped, and the shapes are only the normalized input vectors. We will concentrate on MSGVQ, but most of the following is also applicable to SGVQ.

REDUCING THE MAXIMUM CORRELATION SEARCH TO A NEAREST-NEIGHBOR PROBLEM

The search for the best shape codeword consists in maximizing $Q.\hat{s}$ for $\hat{s} \in \mathcal{C}_s$. The length of the shape vectors is always equal to 1, so this maximization is equivalent to the minimization of the angle between Q and $\hat{s}$. We will use the notation $\angle_{xy} \in [0, \pi]$ to denote the angle between two vectors x and y.

If we denote $Q = (Q^1, \ldots, Q^k)$ and $\hat{s} = (\hat{s}^1, \ldots, \hat{s}^k)$, we can show :

$$Q.\hat{s} = \sum_{i=1}^{k} Q^i.\hat{s}^i = \frac{1}{2}(\| Q \|^2 + \| \hat{s} \|^2 - d(Q, \hat{s})^2) \ ,$$

where d is the euclidean (L^2) distance. Since $\| Q \|$ is constant during the search, and $\| \hat{s} \| = 1 \ \forall \hat{s} \in \mathcal{C}_s$, we see that maximizing $Q.\hat{s}$ is equivalent to minimizing $d(Q, \hat{s})$, i.e. finding the nearest neighbor of Q in $\mathcal{C}_s$.

We can also remark that $Q.\hat{s} = (g.s + m.\mathbf{1}).\hat{s} = g.s.\hat{s}$, so finding the best $\hat{s}$ is also equivalent to finding the nearest neighbor of s in $\mathcal{C}_s$.

From this, we can use the previously known VQ nearest-neighbor search methods, which we review in the next section.

KNOWN SEARCH TECHNIQUES FOR STANDARD VQ

Many search methods rely on the triangular inequality to eliminate candidates during the search. In [8], a method is proposed involving the precomputation of each inter-codeword distance. This method has a quadratic space complexity, and works as follows : suppose we have found a candidate B and are about to test if C is closer to Q than B. We can immediately eliminate C if $d(B, C) > 2.d(Q, B)$. Since we know $d(Q, B)$ and we have precomputed $d(B, C)$, this test involves only negligible time overhead, largely compensated by the acceleration obtained. This method is often referred to as the *spherical constraint*, since it reduces the search space to a sphere centered on the best neighbor found so far. In [4], a much more space-economical method is proposed. It uses a precomputed table of the distances of each codeword from the origin, taken as a reference point for the application of the triangular inequality : if we denote B the nearest point found so far, any codeword C for which either $d(O, C) > d(O, Q) + d(Q, B)$ or $d(O, C) < d(O, Q) - d(Q, B)$ is discarded. Again the test does not consume much CPU time and allows quick elimination of candidates. This method is called the *annular constraint*, because it constrains the search in a volume defined by the difference of two hyperspheres centered on O.

In [6] and [7], the mean and the variance of the codewords are used to eliminate candidates. This method, called *equal-average nearest neighbor search* (ENNS), is not useful here, since those features have a constant value in MSGVQ.

A very popular search method is the *partial distorsion elimination* (PDE, [1]). It simply cuts off the calculation of the distance to a new candidate as soon

as the partial sum calculated so far is greater than the squared distance to the current best neighbor.

Principal Component Analysis (PCA) is also used in some techniques, for dimensionality reduction purpose. See e.g. [2] for an example.

AN ANGULAR CONSTRAINT FOR SPHERICAL NEAREST NEIGHBOR SEARCH

Our new search algorithm is based on the minimization of the angle between Q (or s) and the shape codewords $\hat{s} \in \mathcal{C}_s$. It uses the following lemma :

Lemma 1 (Angular constraint) *Let $R, Q, B, C \in \mathbb{R}^k$. We have*

$$\angle_{QB} \leq |\angle_{RC} - \angle_{RQ}| \Rightarrow \angle_{QB} \leq \angle_{QC} \ . \tag{1}$$

Proof: Let us consider the subspace spanned by Q,R and C. This subspace is at most three-dimensional, thus the following assertions hold :

$$\angle_{RC} \leq \angle_{QC} + \angle_{RQ} \text{ and } \angle_{RQ} \leq \angle_{QC} + \angle_{RC} \ .$$

From these, we can deduce :

$$\angle_{QC} \geq |\angle_{RC} - \angle_{RQ}| \ ,$$

and the end of the proof is straightforward. ∎

The search algorithm may use the lemma in the following way :

1. choose a reference direction R, and for each codeword $\hat{s}$ in $\mathcal{C}_s$ precompute the angular distance $\angle_{R\hat{s}}$,
2. for each query Q, compute the angle $\angle_{RQ}$ and use Equ. 1 to save codewords comparisons during the search : if we denote B the best codeword found so far, we can eliminate every codeword C satisfying $\angle_{QB} < |\angle_{RC} - \angle_{RQ}|$.

This constraint is a spherical generalization of the annular constraint method. We will call it the *angular constraint.*

We can show that the search region defined by the angular constraint has the same form than in the ENNS method (see [7]) : the search is constrained in a volume bounded by two parallel hyperplanes. An 2-dimensional illustration is given in Fig. 1.

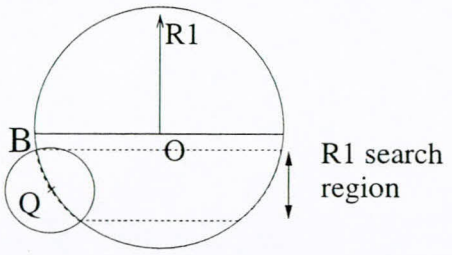


Figure 1: Search region defined by an angular constraint using the reference vector R_1

We propose to use two combined angular constraints, with two reference vectors R_1 and R_2. In our implementation, we use the first two principal components (PC) of the shapes codebook $\mathcal{C}_s$ as reference directions. The first PC is the direction which maximizes the the variance of the shapes projections. These projections are also the cosines of the angles between the shapes and the PC, thus maximizing the projections variance is equivalent to maximizing the angles variance (the cosine function is monotone on $[0, \pi]$). We expect the minimum number of shapes tested to correspond to the maximum angles variance. These vectors are orthogonal to each other, thus they minimize the search regions overlap area. Note that the computation of the principal components is done offline.

A further improvement on the technique is obtained when the codewords are sorted in increasing order of their projections (or angles) on the first reference vector R_1. In $\lceil \log_2 |\mathcal{C}_s| \rceil$ comparisons, using a binary search, it is possible to find the codeword C for which $|\angle_{R_1 C} - \angle_{R_1 Q}|$ is minimal. This codeword may be used as a first estimation of the nearest neighbor. Then the codewords are compared using the ordering along the R_1 axis, as described in Algorithm 1. The two **while** loops compare the codewords C for which $\angle_{R_1 C} > \angle_{R_1 Q}$ and $\angle_{R_1 C} < \angle_{R_1 Q}$, respectively, and stop as soon as the difference $|\angle_{R_1 C} - \angle_{R_1 Q}|$ becomes too large. For each codeword tested in the loop (some may be completely ignored), the condition related to R_2 is also verified, and only the candidates satisfying both tests are directly compared.

COMBINATION WITH PDE

The angular constraint may be combined with the PDE method, because we have shown that a distance criterion is still applicable in the framework of SGVQ

Algorithm 1 Algorithm using two angular constraints

Q is the query vector
we define $\mathcal{C}_S = \{\hat{s}_1, \ldots, \hat{s}_n\}$
$l_{i=1,\ldots,n}$ is a list of shape codewords indices, sorted in increasing order of their projections on R_1
$\alpha_1 \leftarrow \angle_{R_1 Q}$
$\alpha_2 \leftarrow \angle_{R_2 Q}$
$i^* \leftarrow \arg\min_{i=1,\ldots,n} |\angle_{R_1 \hat{s}_{l_i}} - \angle_{R_1 Q}|$
$\alpha_{\text{best}} \leftarrow +\infty$
$j \leftarrow i^*$
$C \leftarrow \hat{s}_{l_j}$
while $j \leq n$ **and** $\alpha_{\text{best}} > \angle_{R_1 C} - \alpha_1$ **do**
  if $\alpha_{\text{best}} > |\angle_{R_2 C} - \alpha_2|$ **then**
    $\alpha_{\text{new}} \leftarrow \angle_{QC}$
    if $\alpha_{\text{best}} > \alpha_{\text{new}}$ **then**
      $\alpha_{\text{best}} \leftarrow \alpha_{\text{new}}$
      $B \leftarrow C$
    end if
  end if
  $j \leftarrow j + 1$
  $C \leftarrow \hat{s}_{l_j}$
end while
$j \leftarrow i^*$
$C \leftarrow \hat{s}_{l_j}$
Here the previous **while** statement is repeated, with the condition $j > 0$ **and** $\alpha_{\text{best}} > \alpha_1 - \angle_{R_1 C}$, and with decreasing j
return B

method	comparisons	time (s.)
full search	1024	33.50
spherical constraint	396.73	11.83
angular constraints	292.78	8.50

Table 1: Average performances of three search methods, with a codebook of size 1024, and blocks of size 4×4

and MSGVQ. The correspondance between the euclidean distances used in PDE and the angles used in our angular constraint is given by : $\cos \angle_{Q\hat{s}} = (\| Q \|^2 + 1 - d(Q, \hat{s})^2)/2$. We could imagine to combine with [8] as well.

EXPERIMENTAL RESULTS

The tests were made in the framework of image compression using MSGVQ. We compared the performances of Algorithm 1 combined with PDE, with the method of Orchard, described in [8], combined with PDE as well. The results for the simple full search method have also been reported.

These algorithms were tested using a shape codebook containing 1024 16-dimensional codewords, trained using the GLA. Performances are measured on 6 greyscale images (Barbara, Boat, Goldhill, Peppers, Mandrill and Lena) of size 512×512. For each algorithm, we measured the average number of codewords compared per query (i.e. for which a distance calculation was necessary), and the average time taken by the search on a simple Pentium-based PC. Results are shown in Table 1. The acceleration ratio for the spherical constraint is 2.83, and 3.94 for the angular constraint. Not only our method is faster than the spherical constraint, but it has a linear space complexity, while the spherical constraint needs a quadratic size table. The ratio between the number of compared vectors is close to the one of the execution times, so the number of tested codewords is a good approximation of the actual performances of the methods. These eliminations methods do not work as well as for standard VQ, where acceleration factors greater than 10 are common. This may be explained by the fact that the principal components contain less information when applied on shape vectors than it does for non-transformed image blocks.

ACKNOWLEDGEMENTS

This work is supported by a PhD. scholarship from the National Scientific Research Fund (FNRS).

REFERENCES

[1] C. D. Bei and R. M. Gray, "An improvement of the minimum distorsion encoding algorithm for vector quantization", *IEEE Trans. Comm.*, **COM-33**:1132-1133, October 1985.

[2] C.-C., Chang, D.-C. Lin, T.-S. Chen, "An improved VQ codebook search algorithm using principal component analysis", *Journal of Visual Communications and Image Representation*, Vol.**8**, No. 1, pp. 27-37, March 1997.

[3] A. Gersho, R. M. Gray, "Vector quantization and signal compression", Kluwer Academic Publishers, Boston, 1992.

[4] C. M. Huang, Q. Bi, G. S. Stiles, and R. W. Harris. "Fast full search equivalent encoding algorithms for image compression using vector quantization". *IEEE Trans. Image Proc.*, **1(3)**:413-416, July 1992.

[5] Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantizer desing", *IEEE Trans. Comm.*, **COM-28**:84-95, January 1980.

[6] C. H. Lee, L. H. Chen, "Fast closest codeword search algorithm for vector quantization", in : *IEE Proc. Vis. Image Signal Process.*, Vol. **141**, No. 3, June 1994.

[7] C. H. Lee, L. H. Chen, "High-speed closest codeword search algorithms for vector quantization", *Signal Processing* **43**:323-331, 1995.

[8] M. T. Orchard, "A fast nearest-neighbor search algorithm", in *ICASSP 91: 1991 International Conference on Acoustics, Speech and Signal Processing. Toronto, Ont., Canada. 14-17 April 1991.*, volume **4**, p. 2297-3000, New-York, NY, USA, 1991. IEEE.

EFFICIENT CODING OF NON-STATIC TEXTURE IN 3D SCENES

Stef Desmet, Bert DeKnuydt, Luc Van Gool and Luc Van Eycken

Katholieke Universiteit Leuven, ESAT–PSI
K.Mercierlaan 94, B–3001 Heverlee, Belgium
{Stef.Desmet,Bert.DeKnuydt}@esat.kuleuven.ac.be
http://www.esat.kuleuven.ac.be/psi/visics

As cheap and powerful 3D render engines become commonplace, demand for nearly realistic 3D scenes is increasing. Besides more detailed geometric and texture information, this presupposes the ability to map dynamic textures. This is obviously needed to model movies, computer and TV screens, but also for example the landscape as seen from inside a moving vehicle, or shadow and lighting effects that are not modeled separately.

Downloading the complete scene to the user, before letting him interact with the scene, becomes very unpractical and inefficient with huge scenes. If the texture is not a canned sequence, but a stream, it is altogether impossible. Often a back channel is available, which allows on demand downloading so the user can start interacting with the scene immediately. This can save considerable amounts of bandwidth.

Specifically for dynamic texture, if we know the viewpoint of the user (or several users), we can code the texture taking into account the viewing conditions, i.e. coding and transmitting each part of the texture with the required resolution only.

Applications that would benefit from view-dependent coding of dynamic textures, include (but are not limited to) multiplayer 3D games, walkthroughs of dynamic constructions or scenes and 3D simulations of dynamic systems. In this paper, a scheme based on an adapted OLA (Optimal Level Allocation) video codec is shown. Important data rate reductions can be achieved with it.

Keywords: *Texture coding, dynamic texture.*

INTRODUCTION

The coding process of texture to be mapped on 3D-scenes can benefit from knowledge of rendering parameters, such as spatial configuration, lighting, haze,

motion blur and resolution of the output device. If a back channel is present, the most obvious parameter to take account with is the viewpoint: clearly, if a huge texture map will be rendered on an object far away, or at a very sharp angle, there is no need to transmit it to full detail: many texture pixels will be rendered to one image pixel. See [5] and [4] for the static case. Especially for dynamic texture, huge bitrate reductions can be obtained by coding the texture locally to only the resolution and quality needed.

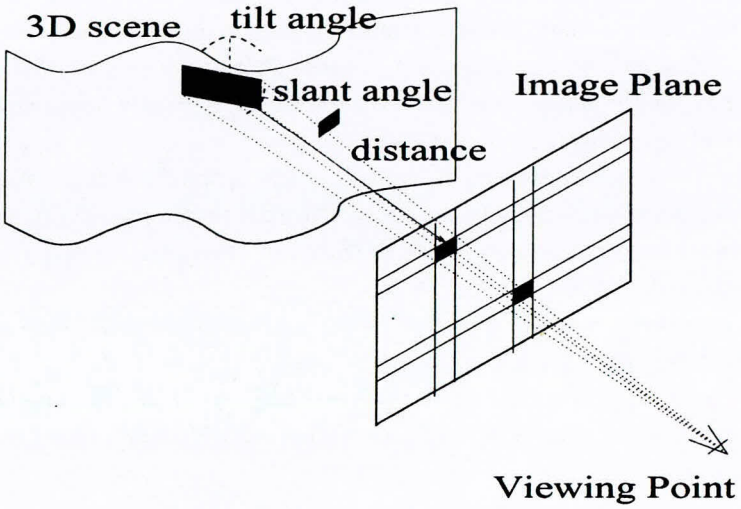
As is shown in figure 1, exploiting this information means that for each part of the texture, we have to calculate the distance, slant and tilt angle with which it will be mapped. The calculation is based in the 3D scene information available at the transmitter and the viewpoint returned to the transmitter via the back channel. If there are multiple viewers, one could either send each one a different texture stream optimal for his own viewpoint, or one could generate a single texture stream that satisfies all viewers, at the expense of a higher bitrate. This, together with any other relevant information, allows us to calculate the lowest horizontal and vertical resolution needed in the texture map, such that the rendered scene will be of high enough quality. Finally, we can code the texture with a video coding engine that is able to code each part of the map just good enough for the required resolution.

OLA CODING ENGINE

In general, any video coder could be used for the coding of dynamic texture. However, DCT based coders, like MPEG1, MPEG2 and even MPEG4, do not have enough flexibility to easily change the resolution in specific image areas and directions. More specifically, one can only drop higher order DCT coefficients to reduce the resolutions, which is a rather crude mechanism, often resulting in severe ringing.

Ideally, we would like a coder, which allows to specify an arbitrary quality criterion for each image area. In this case, the quality criterion would include a directional variable resolution filter. The OLA coder is capable of dealing with any quality criterion.

The standard OLA video coder starts by analyzing the input images and assigning attributes of visual importance (like "texture" or "visually important edges") to each area. In this application, 16×16 blocks were used, but the algorithm is not limited to this. The selected attributes correspond to a classification, which in its turn selects an appropriate visually relevant quality criterion.



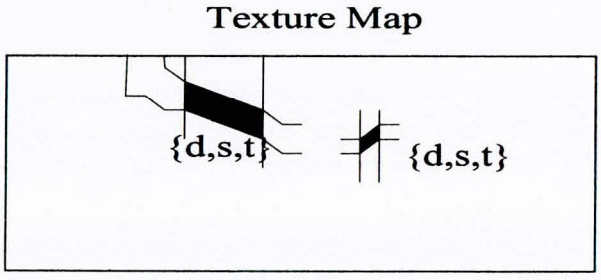


Figure 1: Scene overview

Then, for each image block, the best prediction is chosen out of two motion compensated predictions from the prior images and an intra prediction. The motion vector selection algorithm has half pixel resolution and can handle illumination changes. The prediction selection and the motion vectors are predicted and the error is sent to an adaptive arithmetic coder (AAC). See [2].

The appropriate quality criterion is now evaluated and the result compared with the required quality. If quality is not sufficient, the average value of the prediction error is quantized and transmitted. The criterion is now re-evaluated and an OLA coding step is applied. This is done iteratively until the criterion is fulfilled (or the maximum number of iterations, normally 5, is reached). The number of iterations is predicted and AAC coded.

The OLA coding step itself consists of the optimal selection of two levels, which are quantized and transmitted. Each pixel in the prediction error is now updated with one of these two levels, or with zero. The selection map is adaptively scanned, predicted and AAC coded.

Optionally, a motion vector deblocking and an image deblocking scheme are applied, inside the coding loop. See [1].

Although the encoding scheme is rather complex, mainly because of the analysis steps, the decoding complexity is very low, which would make it suitable for downloading.

EXTENDED QUALITY CRITERION

Because of the information returned via the back channel, we know the viewpoint of the observer. We can easily calculate the distance, slant and tilt angle with respect to the image plane, of each part of the texture to be mapped. We express them in a visibility number, 0 meaning not visible (object at infinite distance, object occluded, or mapping plane perpendicular to the image plane) and 1 meaning fully visible (mapping plane coincides with the image plane).

Then, we can than determine a mapping filter that corresponds to the projection of the texture back to the image plain. The filter is a directional subsampling filter. As we are only interested in the quality after mapping, we can include it in the quality criterion for the OLA coding step. Because of linearity of the system, we can apply the filter to obtain the coding errors after mapping from those in the texture domain. The errors after mapping are added up to determine the final quality of that image part.

In the codec, the filter is implemented as a variable size, separable Gaussian filter. Theoretically, the size of this filter can become infinite. In practice we have to limit the size to about the coding block size. Mirroring is applied at the block borders to reduce bias.

From the filtered errors, the PSNR could be calculated, but a slightly more complex measure is taken. The errors are sent though a lookup table consisting of three parts. First there is a dead zone around zero, to reduce the impact of (camera) noise, followed by a quadratic part, in accordance with the PSNR criterion, and finally a linear part, to limit the effect of large single pixel errors. The accumulated sum of these values returns a quality measure.

Configuration

In the setup of the coder, a number of possibilities remain open. For instance, what to do with texture mapped on a object that becomes occluded. We decided to continue sending only motion vectors, in order to minimize the data we have to send, when the object would reappear.

SIMULATION

Room Walkthrough

We constructed a virtual room, where a movie is playing in a flat television screen on the back wall. Meanwhile, a person enters the room from a door at the backside, walks around the table and finally takes a seat in the couch, close to the screen. The action takes 60 seconds.

If we code the movie as plain video material, we need on average 3.17 Mbit/sec for MPEG2 like quality. If we code it as dynamic texture, only 0.79 Mbit/sec is needed. Rendered, both approaches result in the same quality.

In figure 2, you see a frame from the rendered scene, in figure 3 the texture for the virtual television screen, once coded as video (top), once as dynamic texture (bottom). Finally, in figure 4 you see the momentary bitrates for the respective ways of coding.

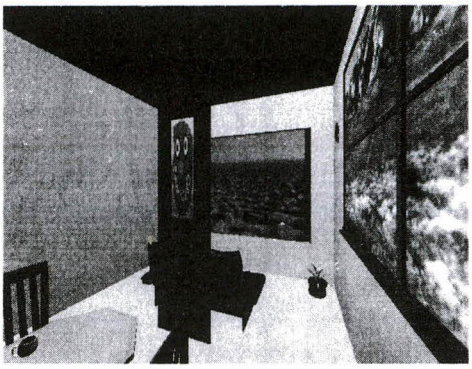


Figure 2: Rendered scene

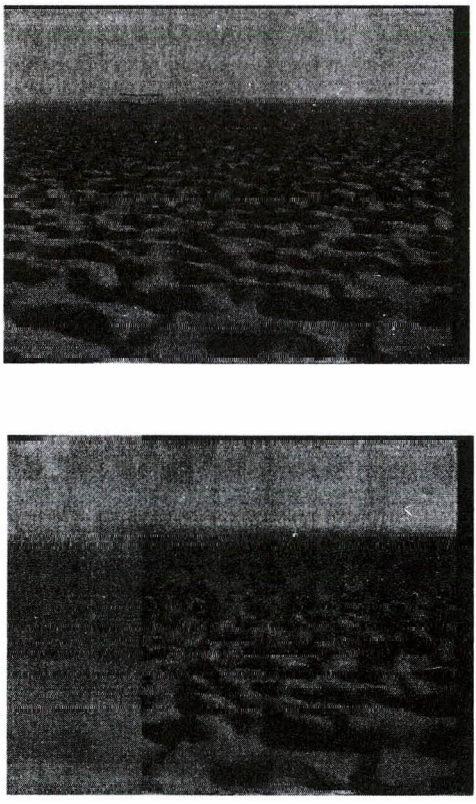


Figure 3: Screen coded as video (top) and as dynamic texture (bottom)

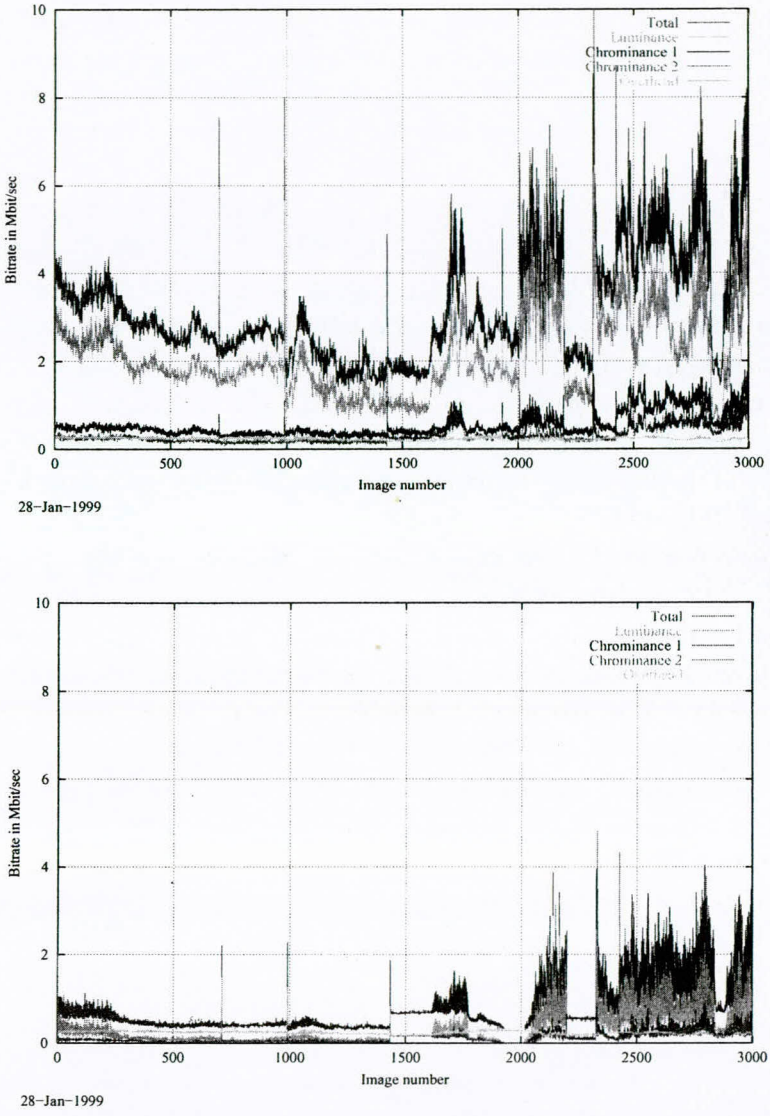


Figure 4: Bitrates for coding as video (top) and as dynamic texture (bottom)

CONCLUSION

In complex 3D scenes, the exploitation of scene information, and especially the viewing point, can greatly reduce the bitrate needed to transmit dynamic (and static) textures. This implies a coder able to deal with other quality criteria than plain PSNR. Existing DCT based (video) coders are not well suited for this application, mainly because the coding does not take place in space domain. More flexible, or ideally downloadable, (de)coding schemes are urgently needed.

REFERENCES

[1] Bert DeKnuydt, Luc Van Eycken, and André Oosterlinck. A simple deblocking scheme suitable for use inside the codec loop. In *IEEE Proceedings, Melecon '94, Antalya, Turkey*, volume 1, pages 9–12, April 1994.

[2] Stef Desmet, Bert DeKnuydt, Nan Li, Luc Van Eycken, and André Oosterlinck. A simple algorithm to extract realistic motion fields out of video sequences. In *Proceedings, EOS-SPIE International Symposium on Fiber Optic Networks and Video Communications, Berlin, Germany*, volume 1977, pages 248–254, April 1993.

[3] Christos Faloutsos and Shari Roseman. Fractals for secondary key retrieval. *CS-TR-2242*, May 1989.

[4] Stefan Horbelt, Fred Jourdan, and Touradj Ebrahimi. View dependent texture coding for transmission of virtual environment. In *Proceedings, Conference on Image Processing and its Applications, Dublin, Ireland*, volume 1, pages 433–437, June 1997.

[5] Fred Jordan and Touradj Ebrahimi. View-dependent texture for transmission of virtual environment. MPEG4-meeting, doc. ISO/IECJTC/SC29/WG11/M1501, Maceió, Brasil, November 1996.

High-Quality Embedded Compression for Digital TV

Mihaela v.d. Schaar-Mitrea [1] and Peter H.N. de With [2]

[1] Philips Research Labs., 345 Scarborough Rd. Briarcliff Manor, USA.
[2] University Mannheim, Fac. Computer Engineering,
B 6-26, D-68131 Mannheim, Germany.

Abstract – *We present a novel near-lossless technique to reduce the memory costs of an MPEG-2 decoder with a factor of almost 6, by recompressing the I- and P-reference pictures prior to memory storage for motion compensation. For this purpose, a special feedforward compression system is embedded, which enables a near-lossless quality by reusing the MPEG quantization parameters.*

1 Introduction

In the coming years, digital television broadcasting will gradually replace analog television. It is expected that the increasing number of broadcasting services and the high video and audio quality offered will lead to a growing demand for DTV receivers. However, the consumer acceptance of this new technology will rely considerably on low implementation costs of the DTV receivers. This paper presents a new, near-lossless technique to reduce the memory costs of a key component in all DTV receivers, i.e. the MPEG-2 decoder [1], by recompressing the I- and P-reference pictures prior to memory storage for motion compensation. The implementation of such an embedded compression technique is complicated, since it is inserted in the recursive motion-compensated MPEG coding process. In [2][3][4], it was already shown that the embedded compression of the MPEG-decoder reference pictures is possible. This paper presents a new scheme, which partially reuses the MPEG decoding system itself, to obtain a near-lossless performance. Besides cost reduction, the embedded compression within DTV receivers is an attractive aid to reduce the communication bandwidth with the shared (background) memory. This communication channel is often congested, because many digital signal processing functions and a control processor demand access to the memory simultaneously.

The paper is divided as follows. Section 2 addresses the most relevant constraints for embedding a compression system with near-lossless quality in the motion-compensated prediction loop of the MPEG decoder. Section 3 is devoted to the embedded memory-compression system being used and discusses the conditions for near-lossless quality operation. Section 4 presents the simulation results of a state-of-the-art MPEG-2 decoder with embedded compression and gives the conclusions.

2 System requirements

In a preliminary investigation [2], it was found that embedded compression of reference data in MPEG is possible with a very simple transform coding system, yielding a compression factor of 4. However, the proposed algorithm preserves the final picture quality only at low and moderate transmission bit-rates. In this paper, we propose a new system which realizes a near-lossless compression of the reference frames with a factor of 6, independent on the transmission bit-rate or picture format. This is a relevant feature, since a DTV receiver should be able to operate flawlessly for various bit rates and image formats. In this paper, we assume that the display resolution is larger or equal to the input image resolution. This is the most challenging system, since any interference between MPEG-2 decoding and embedded (memory) compression becomes clearly visible and it is not masked by a lower resolution display, as assumed in alternative systems [3][4]. In the sequel, it is assumed that the input and output images have standard-definition resolution (i.e. 720 pixels $\times$ 576 (/ 480) lines at 25 (/ 30) frames/s). However, the algorithm presented is suited for all 18 different ATSC formats, including the 1920 $\times$ 1080 format at 30 frames/s. For a successful application of embedded memory compression in MPEG (from now on referred to as "embedded compression"), a number of system requirements have to be satisfied, some of them being rather unique for this type of system.

- Fixed compression factor for each individual picture.
- Suitability for various bit rates and image formats, and for HDTV-resolution display.
- Near-lossless image compression, such that no artifacts from the embedded compression become visible.
- Robustness for repetitive coding/decoding in the recursive MPEG coding loop (i.e. no prediction drift).
- Easy pixel data access, to enable real-time operation of the motion-compensated decoding.

Especially the last three system requirements reveal that special algorithms are required for embedded compression.

3 Embedded compression algorithm

3.1 Feedforward coding scheme

To fulfill the previously mentioned system requirements (e.g. data access, fixed compression up to 6, etc.), while preserving a high image quality independent of the MPEG-2 transmission bit-rate, a novel feedforward-buffered coding scheme, as shown in Fig. 1, has been adopted. The main steps of the scheme are as follows.

Multiplication-free DCT. The DCT and IDCT employed in the embedded compression scheme are based on a new efficient computation scheme [2], which substitutes all the multiplications with add and shift operations. This was achieved by individual coefficient normalization and using special integer representations. The proposed multiplication-free DCT-IDCT was found superior with respect to cost and computational transparency, when compared with other fast DCT algorithms.

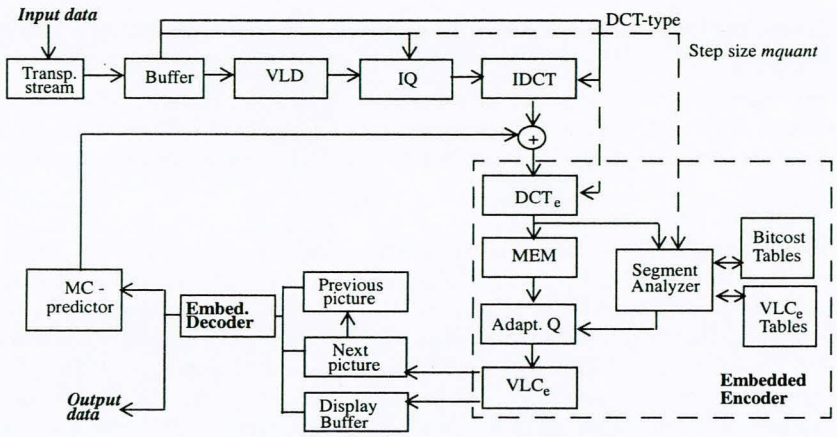


Fig. 1. MPEG-2 decoder with embedded compression using feedforward transform coding.

The embedded codec also employs blocks of 8 × 8 pixels to match it with MPEG-2, thereby avoiding any reformatting of the data by the embedded coder.

Robust Quantization. For the frequency-dependent quantization of the transform coefficients, two options exist: either adopt a dedicated low-complexity quantizer based on simple shift operations (i.e. quantization with powers of 2) or employ the MPEG-2 quantizer. Since this choice is key to our embedded compression algorithm, we will discuss this later in more detail (see Section 3.2). However, in either case, the quantization algorithms should be modified as described in [2] to prevent that the decoder would drift away from the motion-compensation reconstruction loop.

Entropy Coding. A low-cost Huffman encoder has been designed, featuring adaptive zonal selection of non-zero coefficients: the position of the relevant coefficients is specified by the largest column and row values for which non-zero coefficients still occur. Despite its slightly lower compression capability, this entropy coder is considerably less complex than the MPEG-2 significance map compression, because it requires less processing steps and memory.

Feedforward bit-rate control. A feedforward system was used to ensure a fixed compression factor per picture (i.e. frame or field). The reference pictures were divided in groups of one or several MPEG-2 macroblocks, called *segments*, which are coded together as an entity. The video data within the segments are analyzed prior to compression, in order to select the quantization strategy that gives the best image quality and still yields the desired compression factor. Experiments have been conducted for fixed and variable compression factors per segment. In the latter case, the bits left unused in a certain segment can be employed for an improved quantization of the subsequent highly detailed segments.

3.2 Near-lossless embedded compression dependent on the transmission bit-rates

The conventional video coding algorithms (i.e. MPEG-2, H.263, etc.) and extensions such as MPEG-4, can be divided in two classes employing different quantization techniques, depending on their image quality and the corresponding transmission bit-rate.

Low to moderate transmission bit-rates. The first class includes video coding schemes giving low and moderate decoded image quality (SNR $<$ 33 dB), like MPEG-2 codecs at low bit-rates (e.g. $<$ 6 Mbit/s for SD-resolution images), H.263, MPEG-4, etc. To obtain these target bit-rates, a relatively coarse quantization is applied for most macroblocks and only a small number of non-zero AC coefficients are transmitted. As a consequence, the employment of a relatively coarse quantizer in the embedded coder does not lead to a visible interference with the surrounding video coding system. For this class, the 2^n quantizer mentioned in the previous subsection forms an appropriate choice, which allows for a very efficient hardware implementation and prevents the unnecessary VLC coding of large AC amplitudes. Since the image quality obtained by the embedded coder considerably outperforms that of the surrounding system, the two coding mechanisms do not need to be "synchronized" (i.e. can work independently of each other). It is interesting to mention that this low-complexity embedded coder is extremely suitable for reducing the cost of the conventional video encoders, since the transmission bit-rate is *a-priori* known and can be controlled by the encoder itself. For low and moderate bit-rates, a more accurate quantizer would result in an unnecessary complexity increase.

High transmission bit-rates. The second class includes the DTV receivers that should be able to decode all possible bit-rates and image qualities. For relatively large bit-rates (e.g. $>$ 9 Mbit/s for SD resolution images), the decoded image quality is very high (SNR $>$ 36 dB) and the transform coefficients are quantized only marginally. Therefore, the embedded compression quantizer should avoid the zeroing of AC coefficients, by employing a finer weighting function for the embedded compression quantizer. To minimize the potential interference between the embedded compression and the applied coding standard, the best alternative is to "track" the applied MPEG dequantization step (i.e. *mquant*) and reuse the MPEG quantization process. To realize a near-lossless compression, the MPEG-2 coding parameters are also employed by the embedded compression. The quantization parameter *mquant* used by the MPEG coder for the I-pictures can be reused by the embedded coder. Whenever an image has been coded on a frame basis, in order to obtain the same quantization process as in MPEG-2, the same DCT-structure of the luminance blocks (e.g. frame- or field-based) should also be employed. Consequently, the MPEG-2 IDCT and the embedded-compression DCT can be discarded from the computational flow of the I-picture decoding, thereby reducing the overall computational complexity. In most cases, the compression factor achieved by MPEG-2 for each individual segment is already larger than 6 and the embedded compression of the I-image is therefore lossless. When the desired amount of bits per segment is exceeded, the embedded feedforward codec increases the quantization strategy per

segment to ensure that the desired compression factor is achieved.

The embedded compression of the P-pictures is different, since the decoded MPEG-2 blocks represent only temporal prediction errors, while the entire reconstructed picture should be stored within the memory for further decoding. After a group of MPEG-2 macroblocks has been decoded (i.e. prediction-error decompression and motion compensation), the data is again (embedded) compressed prior to its storage within the reference frame memory. The feedforward codec presented in Section 3.1 analyzes the data within a segment by employing different quantization strategies, varying from coarse to fine. For the MPEG-2 quantizer inside the embedded coder, these different quantization strategies are represented by various values of the *mquant* parameter. For each quantization strategy, the total bit-cost after the entropy coding of the significant AC coefficients is compared with the desired compression factor. If the desired compression factor is not reached, the quantization step is increased and the process is repeated.

For the feedforward coding of the P-pictures, two initialization strategies can be adopted:

- initialize the feedforward embedded coder with the coding parameters employed by MPEG-2 for the compression of the previous I-picture;
- initialize the feedforward embedded coder with the finest quantization strategy and use no coding parameters of previous images.

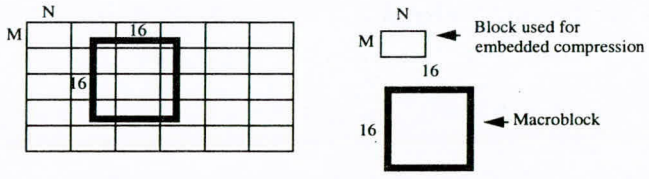


Fig. 2. Macroblock extraction from a compressed frame.

3.3 Memory-bandwidth reduction computation

In MPEG-2, the motion-compensation requires that macroblock data should be retrieved from memory on "arbitrary" positions, which mostly do not coincide with the block grid employed for embedded compression. Therefore, additional (embedded) decoding of border pixels is required, which consumes additional memory bandwidth, as illustrated in Fig. 2. Since the image data is stored in the memory in compressed form, an image header is separately stored, containing both the start addresses of the coded segments and the (offset) positions of each compressed block within the segment, called P_B (Pointer Bits). For analysis, it is proposed to evaluate a memory-bandwidth reduction (MBR) factor, assuming that the embedded compression system with compression CF employs $N \times M$ blocks, so that

$$MBR = \frac{16}{\left\lfloor \frac{16+N-1}{N} \right\rfloor} \cdot \frac{16}{\left\lfloor \frac{16+M-1}{M} \right\rfloor} \cdot \frac{1}{\frac{M \cdot N}{CF} + \frac{P_B}{8}} \cdot \tag{1}$$

For the system discussed with $N = M = 8$, CF=6 and P_B=9, the MBR equals 2.5. The MBR should be considered carefully because for a CF=2, the memory bandwidth would have increased, due to the embedded compression. Similar calculations can be derived for various block sizes, compression factors and address information systems. This analysis becomes extremely important when an embedded compression system needs to be dimensioned such that it achieves the best trade-off between image quality (i.e. large block sizes) and memory bandwidth reduction (i.e. small block sizes).

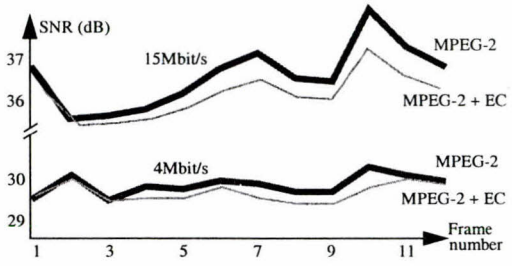


Fig. 3. Relative comparison between the embedded compression (EC) image quality and the image quality of the MPEG-2 codecs at various bit-rates.

4 Results and conclusions

Experiments have been carried out employing this novel embedded compression technique inside MPEG-2 decoders. The results obtained show that the described feedforward embedded coder with the MPEG-2 quantizer achieves a memory reduction of 6 with almost no loss in image quality, independent of the transmission bit-rate (see Fig. 3). Simulations with the embedded compression have been performed for a range of image sequences at various bit rates (4-15 Mbit/s). Mostly, the quality decrease due to embedded compression is very small, e.g. 0.2 dB and 1 dB for the worst case, whereas the memory cost of the DTV receiver is considerably reduced. A major advantage of the proposed algorithm is its robustness for repetitive coding/decoding in the recursive MPEG decoding loop, so that negligible prediction drift occurs. Another important feature of the technique proposed is that it is equally well applicable to other image resolutions, because it can "follow" the MPEG decoder, as it is partially based on the same techniques and coding parameters.

The hardware complexity of the proposed system can be downscaled, if required. The complexity of the embedded coder is dominated by the feedforward buffer and the MPEG-2 quantizer. Fortunately, both blocks can be reduced in complexity, e.g. by using only one MPEG macroblock as a buffer for the feedforward coding scheme and/or employing the 2^n quantizer which can be implemented with bit-shifts only. A trade-off between cost and performance is therefore possible and is illustrated by Fig. 4a and b.

Finally, it was found that the novel algorithm can be also successfully applied for reducing the memory requirements of other digital signal processing functions (e.g. sample-rate conversion, de-interlacing, etc.) or video encoding systems (e.g. MPEG-2 encoders, MPEG-4 and H.263 codecs). The proposed embedded coder outperforms the algorithms currently known in the literature for reducing the memory size of DTV receivers, because it can operate with near-lossless quality for a large range of bit-rates and image resolutions. Therefore, the proposed algorithm can be successfully employed for reducing the memory of a scalable (programmable) DSP system, of which the operating range and input image size is not known in advance.

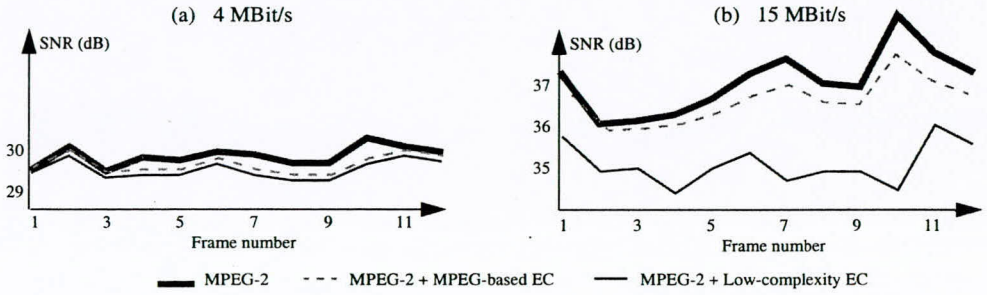


Fig. 4. Image quality of the MPEG-2 decoded SD-resolution MOBI sequence (very critical) at 4 Mbit/s (a) and 15 Mbit/s (b) without and with embedded compression (EC) with fixed compression factor 6.

References

[1] *Coding of Moving Pictures and Associated Audio*, International Standard, ISO/IEC 13818, November 1994.

[2] M. v.d. Schaar-Mitrea and P.H.N. de With, "Novel embedded compression algorithm for memory reduction in MPEG codecs", VCIP '99, San Jose CA, SPIE Proc. Vol. 3653 part 2, pp. 864–873, Jan. 1999.

[3] A. Vetro *et al.*, "A minimum drift 3-Layer scalable DTV decoder", ICCE '98 Digest Techn. Papers, Los Angeles CA, pp. 54–55, June 1998.

[4] J.G.N. Henderson, "All-Format Decoders and Set-Top Boxes", Proc. ICIP '98, Vol. 1, Chicago IL, p. 4, Oct. 1998.

[5] P.H.N. de With. P.H. Frencken and M. v.d. Schaar-Mitrea, "An MPEG decoder with embedded compression for memory reduction", IEEE Trans. Consum. Electron., Vol. 44, No. 3, pp. 545–555, August 1998.

TIMING ATTACK: WHAT CAN BE ACHIEVED BY A POWERFUL ADVERSARY?

Gaël Hachez, François Koeune, Jean-Jacques Quisquater

UCL Crypto Group,
Place du Levant 3, B-1348 Louvain-la-Neuve, Belgium.
{hachez,fkoeune,jjq}@dice.ucl.ac.be
URL: http://www.dice.ucl.ac.be/crypto

INTRODUCTION

Implementations of cryptographic algorithms often perform computations in non-constant time, due to performance optimizations. If such operations involve secret parameters, these timing variations can leak some information and, provided enough knowledge of the implementation is at hand, a careful statistical analysis could even lead to the total recovery of these secret parameters. This idea, due to Kocher [Koc96], was developed in [DKL+98], were a timing attack against an actual smart card implementation of the RSA[1] was conducted.

The paper's conclusion was that, however impressive, the obtained results could be improved even further in several aspects, especially regarding the error-correction policy.

The paper first presents the basic principle of the timing attack, then briefly discusses several error-correction policies and describes the results we obtain implementing them on a parallel architecture of 4 processors PA8000 @ 180Mhz with 4 Gbytes RAM.

TIMING ATTACK

We begin by describing the main features of the timing attack. A more complete description can be found in [DKL+98].

We will attack an RSA implementation, performed in an earlier version of the cryptographic library we developed for the CASCADE [Cas] smart card.

[1] In fact, a pre-version of the cryptographic library the UCL Crypto Group developed for CASCADE[Cas]: in view of the devastating results of the timing attack, the final version was modified to make it immune against it.

The scenario of the attack is the following: Eve disposes of a sample of messages M and, for each of them, the time needed to compute the signature of the message with the key k. Her goal is to recover k.

To perform the timing attack, Eve needs a bit information about the implementation. We suppose she knows that the computation of $m^k \bmod n$ is performed using the well-known left to right square and multiply (fig. 1). She also knows that both the multiplication and the square are done using the Montgomery algorithm.

```
x = m
FOR i = n − 2 DOWNTO 0
    x = x^2
    IF (k_j == 1) THEN
        x = x · m
ENDFOR
RETURN x
```

Figure 1: Square and multiply

We will not describe the Montgomery algorithm in detail here. The only point of interest is that the time for a Montgomery multiplication is constant, independently of the factors, except that, if the intermediary result of the multiplication is greater than the modulus, then an additional subtraction (called a reduction) has to be performed at the end of the multiplication. This means, and that is the basis of our attack, that for some factors the multiplication time will be longer than for others.

A first target: attacking the multiply

The most obvious way to take advantage of this knowledge is to aim our attack at the *multiply* step of the square and multiply.

We start by attacking k_2, the second bit[2] (MSB first) of the secret key. Performing the Montgomery algorithm step-by-step, we see that, if that bit is 1, then the value $m \cdot m^2$ will have to be computed during the square and multiply.

Now, for some messages m (those for which the intermediary result of the multiplication will be greater than the modulus), an additional reduction will have to be performed during this multiplication, while, for other messages, that

[2]Of course we can suppose that the first bit of the key is always 1.

reduction step will not be necessary. So, we are able to divide our set of samples in two subsets: one for which the computation of $m \cdot m^2$ will induce a reduction and another for which it will not. If the value of k_2 is really 1, then we can expect the computation times for the messages from the first set to be slightly higher than the corresponding times for the second set.

On the other hand, if the actual value of k_2 is 0, then the operation $m \cdot m^2$ will not be performed. In this case, our "separation criterion" will be meaningless: there is indeed no reason for which a m inducing a reduction for the operation $m \cdot m^2$, would also induce a reduction for $m^2 \cdot m^2$, or for any other operation. Therefore, the separation in two subsets should look random, and we should not observe any significant difference in the computation times.

Once this value is known, we can simulate the computation up to the multiplication due to bit k_3, attack it in the same way as described above, and so on for the next bits.

The previous approach allowed us to recover 128-bit keys by observing samples of 50 000 timings. However, this approach was unsatisfying on several respects (see [DKL+98]) and another one, described below, turned out to be much more efficient.

Second target: attacking the square

There is a more subtle way to take advantage of our knowledge of the Montgomery algorithm: instead of the multiplication phase, we could turn ourselves to the *square* phase.

The idea is quite similar to that of section : suppose we know the first $i-1$ bits of the key and attack the ith. We begin by executing the first $i-1$ steps of the square and multiply algorithm, stopping just before the possible - but unknown - multiplication by m due to bit k_i ; we denote by m_{temp} the temporary value we obtain.

First, we suppose k_i is set. If this is the case, the two next operations to be performed are

1. multiply m_{temp} by m,
2. square the result,

and both of these operations will be done using the Montgomery algorithm. We simply execute the multiplication and then, for the square, determine whether

an additional reduction will be necessary or not. Doing this for every message, we divide our samples set in two subsets M_1 (additional reduction) and M_2 (no reduction).

Next, we suppose $k_i = 0$. In this case, no multiplication will take place, and the next operation will simply be m_{temp}^2. Once again, we divide the samples set in two subsets M_3 and M_4, depending on whether this square requires a reduction or not.

Clearly, only one of these separations makes sense, depending on the actual value of k_i. All we have to do now is to compare the separations: if the timing difference between M_1 and M_2 is more important than that between M_3 and M_4, then conclude $k_i = 1$, otherwise, conclude $k_i = 0$.

This approach has several advantages on the previous one, but its most interesting feature for the current purpose is its error-detection property. This will be developed in next section.

Using this attack, we were able to recover 128-bit keys with 20 000 timings. Some keys were disclosed with only 12 000 timings.

ERROR-DETECTION

One remarkable property of our attack is that it has an error-detection property. This is easy to understand on an intuitive point of view: remember that the attack basically consists in simulating the computations until some point, then building two decision criteria, with only one of them making sense, depending on the searched value, and finally deciding the bit value by observing which criterion actually makes sense. Also note that each step of the attack relies on the previous ones (we need the previous bits values to simulate the computation).

Now, suppose we made an erroneous decision for the value of bit k_i. In the following step, we will not correctly simulate the computations, so that the value m_{temp} we will obtain will not be the one involved in step $i + 1$. Our attempts to decide whether the Montgomery multiplications will involve an additional reduction or not will thus not make sense, and the criteria we will build will *both* be meaningless. This remains true for the following bits.

In practice, this translates to abnormally close values for the two separations: while, as long as the choices were right, the two separations were generally[3] easy

[3]There are however some tedious cases, were the two criteria are uneasy to differentiate although no error has been made. That is why it is better to wait until several contiguous low values are observed before to conclude to an error.

to distinguish, one of them being clearly more significant than the other, they appear much more similar (and both bad) after an erroneous choice has been made. This fact is well illustrated in figure 2, showing the attack of a 512-bit key on the basis of 350 000 observations. The decision criterion is simply the difference between the mean times for the two subsets, and the graph shows the absolute value of $diff_1$ (the difference between M_1 and M_2) minus $diff_2$ (difference between M_3 and M_4). Clearly, an error has occurred near bit 149.

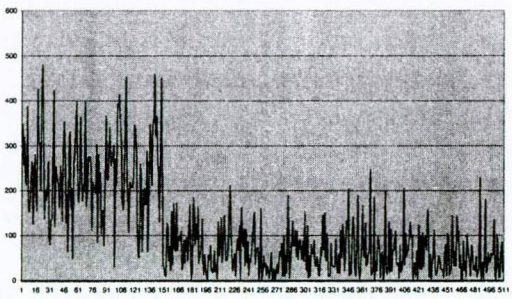
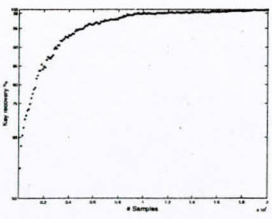


Figure 2: Detection of an error for a 512-bit key

Two things are worth noting about this figure: first, it can be seen that, even when the guesses are correct, some values of the criterion turn out to be very small. This suggests that, to detect an error, it would be better to consider the criterion on few consecutive bits instead of single ones. Secondly, it must be noted that this graph corresponds to a rather large sample; as the sample size decreases, errors will of course become much more difficult to detect.

Thanks to this error-detection property, it should be possible to build attack variants that would be robust against errors, as long as their proportion remains sufficiently low. This can be extremely important, as the sample size appears to decrease very fast when the error proportion grows (see figure below). A good correction policy should thus allow to reduce drastically the sample size. We will come back on this later.

Success rate	64 bits	128 bits	256 bits
75%	300	800	2500
90%	1000	2700	10000
95%	1750	5000	20000
99%	3750	9500	42000



PARALLEL IMPLEMENTATION

Once the samples have been collected, most of the computation time is spent in simulating the algorithm's behavior for each message and computing statistics. These computations can easily be parallelized.

We built a parallel implementation, in which the total sample is cut into parts and shared among several processors. Each processor handles its own subset (checking Montgomery multiplications, computing partial means, ...) and they only interact to share their statistics and make a common decision about the bit value. The level of parallelism is thus very high, and the speed is multiplied by the number of processors at disposal.

Our implementation on an HP/S-Class of 4 processors PA8000 @ 180Mhz, with 4 Gbytes SDRAM, allowed to reduce the time to break a 512-bit key to about 40 minutes. 128-bit keys were recovered in a few seconds.

Unfortunately, the real bottleneck for the attacker is not so much his computation power than the smart card's one. As a matter of fact, the only way to build the time sample is by forcing the smart card to sign every message. Such a processing of some hundred thousands different messages by the same smart card seems, at several respects, unrealistic. Therefore, of much more interest would be to be able to reduce the sample size we need for the attack. This is the purpose of next section.

ERROR CORRECTION POLICY

As we said before, it should be possible, by implementing an adequate error-correction policy, to reduce drastically the size of the needed sample. We therefore tried several ways to make the attack robust against an as big error proportion as possible.

Errors are always detected based on some criterion value. After many trials, we decided to use a criterion based on the decision criterion for the next bit value. We trace our decision criterion back from the last bit. If we find a bit with a small criterion value preceded by a window of bits with high criterion values, we concluded that the last bit was not correct. If we apply this criterion to figure 2, we find that bit 149 was the first bad bit. Quite surprising, more sophisticated criteria based on statistical models, did not reveal more efficient.

Basically, the error correction policy is the following:

1. start by performing the attack until the end, without trying any correction;

2. identify the last place P_1 were the criterion was high: we can reasonably suppose that the guess was correct until this point;
3. try a first correction, by changing the bit value at position $P_1 + 1$; continue the attack until its end and identify the last place P_2 were the criterion was high;
4. if P_2 is further than P_1, then the correction we tried was probably right; we can now recursively repeat the process: execute step 3, using P_2 instead of P_1;
5. otherwise, our correction attempt was incorrect: restore previous bit value, and try to change the bit value at position $P_1 + 2$, then $P_1 + 3$, ...
6. if the correct key is not found this way, we conclude the first error occurred *before* P_1; we thus find the last place, before P_1, were the criterion was high, and restart the same process.

Of course, every time a complete key is guessed, we check whether it is the right one (this is easily done in a public key cryptosystem) and stop as soon as the key is found.

However, one problem remains: we have seen that some bits are more difficult than others, in the sense that the criterion is not very clear there, and that the guess decision is taken in the wrong direction. The above algorithm will try to detect these errors and correct them. But what will happen if two "difficult bits" are consecutive? If this is the case, then the correction attempt will be directly followed by an error ; the criterion will thus remain low for subsequent bits and the algorithm will not detect it has done the good correction.

One solution is to try to correct not only single bits, but small bit windows. In the above algorithm, instead of changing a single bit value, we will explore all possibilities for three or four consecutive bits. We build a list with all possibilities and order this list from the most to the least probable using our decision criterion. For each possibility, we then try to complete the attack and proceed recursively as before. Note that, for this to be fully efficient, we need to use at least two different window sizes for the evaluation criterion window and for the correcting window. (otherwise we simply report the problem a bit further).

CONCLUSION

The previous paper [DKL+98] showed that timing attacks are serious threats against cryptosystems implementation on devices where timing measurements

can be precisely done. This is especially the case for smart cards. The longest step (in time) is to get enough timing samples. The number of needed samples are quite large for usual key sizes. Further reduction of this number will increase the practicability of the attack.

In this paper, we focused on improvements able to reduce the number of samples needed to successfully recover a secret key. We showed that the timing attack enjoy good error-correcting properties. With our error-correcting method, we are able to reduce the number of samples by a factor two.

Key size	Results (number of samples)	
	without error correction	with error correction
64	1 500–6 500	1 500–4 000
128	12 000–20 000	5 000–8 000
256	70 000–80 000	30 000–40 000
512	350 000–400 000	150 00–200 000

We are still working on devising better error-correcting schemes because we believe that more information can be extracted from these timings.

REFERENCES

[Cas] Cascade (Chip Architecture for Smart CArds and portable intelligent DEvices). Project funded by the European Community, see http://www.dice.ucl.ac.be/crypto/cascade.

[DKL+98] J.-F. Dhem, F. Koeune, P.-A. Leroux, P. Mestré, J.-J. Quisquater, and J.-L. Willems. A practical implementation of the timing attack. In *Proc. CARDIS 1998, Smart Card Research and Advanced Applications*, 1998.

[Koc96] P. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In N. Koblitz, editor, *Advances in Cryptology - CRYPTO '96, Santa Barbara, California*, volume 1109 of *LNCS*, pages 104–113. Springer, 1996.

THE DIGITAL TIMESTAMPING PROBLEM

Bart Van Rompay*, Bart Preneel†, Joos Vandewalle

Katholieke Universiteit Leuven, ESAT–SISTA/COSIC
K. Mercierlaan 94, B-3001 Heverlee, Belgium
bart.vanrompay@esat.kuleuven.ac.be

In this paper we examine the relation between time and cryptography. Time-stamping systems provide temporal authentication of electronic documents, they certify the date a document was created or last modified. Important applications are found in legal situations, electronic contracts, the protection of Intellectual Property Rights, and Internet security in general. We give an overview of the problem and discuss several state-of-the-art techniques.

INTRODUCTION

The interaction between time and cryptography is a relatively new subject. [HS91] is the reference work in this area. A study was made in the framework of the Belgian Timesec project (see [MQ97] and [PRQ+98]). Most cryptographic research has been focused on secure messaging systems. Symmetric and public key encryption are used for protecting confidentiality. Authenticity (data integrity and data origin authentication) can be obtained by applying keyed hash functions (for the symmetric key case) or digital signatures (in the public key scenario).

However, the secure maintenance of documents with a long lifetime is more complicated. Key pairs used in public key systems have a limited lifetime and may be revoked if the private key has been compromized. In a non-repudiation service we must be able to determine, at a later time, if a document was signed (using the owner's private signature key) within the validity period of the certificate. In other applications the time itself can be important, for example the date of an inventor's patent claim to establish precedence over competing claims.

Digital timestamping provides the solution to this problem, by certifying the date and time on which a particular document is submitted to the timestamping

* Sponsored by the Timesec project of the Federal Office for Scientific, Technical and Cultural Affairs (OSTC), Belgium.

† F.W.O. postdoctoral researcher, sponsored by the Fund for Scientific Research – Flanders (Belgium).

service by the owner. When dealing with electronic data it will be necessary to timestamp the data itself, without any regard to the physical medium on which it resides. The security depends on the following general properties:

- It must be infeasible to timestamp a document with a date and time different from the present one.
- It must be infeasible to change even a single bit of a timestamped document without the change being apparent.

A SIMPLE CENTRALIZED SERVICE

A basic solution for timestamping relies on the use of a trusted third party, the *Time Stamp Authority* (TSA). The TSA appends each request with the current date and time and digitally signs the result to produce a timestamp. The scheme can be enhanced by using a hash function to compress the document sent to the TSA. This allows to preserve the confidentiality of the original data, reduces the bandwidth and storage requirements, and improves the efficiency (digitally signing is a computationally expensive operation).

If a document is to be securely and uniquely represented by its hash value, the hash function used must meet certain cryptographic requirements. This means that the choice of function has to be imposed by the TSA. More specifically the hash function needs to be one-way (impossible to invert the hash to find the original data) and collision-free (impossible to find two documents with equal hash values). These properties cannot be proven for practical functions and many constructions have been broken (for example MD4, which was suggested as a practical hash function in [HS91]). See [RPV98] for a discussion on practical hash functions. RIPEMD-160 and SHA-1, recently standardized by ISO/IEC, are good candidates.

The *Stamping Protocol* used to generate a timestamp works as follows:

1. Client A sends his identity and the hash value of the document he wants to have timestamped: he sends $ID_A, h(X)$.
2. The TSA returns the timestamp certificate $C = S_K(ID_A, h(X), n, t)$, where n is the certificate serial number, t the current date and time, and S_K the digital signature by the TSA.
3. The client receives the certificate and checks that it is signed by the TSA, that it contains the hash of the document he asked a timestamp for and the correct time (within reasonable limits of precision).

Another user who questions the validity of the timestamp C for the document X will use the following *Verification Protocol*:

1. Check that the digital signature on the certificate corresponds to the TSA.
2. Check that the hash value $h(X)$ corresponds to the document X.

The fundamental problem with this scheme is that we need absolute trust in the TSA since nothing prevents it from producing fake timestamps. One possible alternative is to distribute the trust over several randomly chosen signers, making forgery difficult since all parties involved would have to be corrupt ([HS91]). Such a scheme doesn't even need the use of a centralized TSA. However the large degree of cooperation which is required from other parties seems to make it impractical in large environments. Hence we will further concentrate on schemes which enhance the centralized service by means of a mechanism that links together all the timestamps issued by the TSA, creating an unforgeable temporal chain.

A NOTE ON SECURE TIME

The timestamps issued by the TSA contain an *absolute temporal measure*, so there is a need for a common reference time, agreed upon by all users. Reliable time can be provided by atomic clocks and distributed by systems such as NTP, the *Network Time Protocol*, a time service offered on the Internet (see [NTP]), allowing the synchronization of internal clocks with an accuracy of about a few tens of milliseconds on a WAN.

THE BASIC LINKING PROTOCOL

In [HS91] a protocol is introduced that minimizes the trust requirement by making the TSA link all the issued timestamps together in a temporal chain. This is achieved by including in each timestamp information from the previous timestamp. To protect the temporal ordering we need another one-way and collision-resistant hash function H. The timestamps are constructed as follows:

1. The client, who makes the nth request to the TSA, sends $ID_n, h(X_n)$.
2. The TSA returns the certificate $C = S_K(ID_n, h(X_n), n, t_n, L_n)$, which contains linking information $L_n = (ID_{n-1}, h(X_{n-1}), t_{n-1}, H(L_{n-1}))$.
3. When the next request is received the TSA also sends ID_{n+1} to client ID_n to produce a link in the future direction of time. The complete timestamp consists of (C, ID_{n+1}).

4. Client ID_n verifies the signature, the hash of his document and the time.

Someone who wishes to verify the timestamp will:

1. Check the signature of the TSA and the hash of the document.
2. To make sure there was no collusion between client ID_n and the TSA he can contact clients ID_{n+1} and ID_{n-1}, ask for their certificates and check the linking between them. Using their timestamps he can go as far forwards or backwards along the chain as he wishes.

The use of the hash function H in linking timestamps together gives us reliable *relative temporal information.* One can check that a particular timestamp was created before or after another one. Users of the system could generate and timestamp random numbers at regular time intervals to obtain trusted linking items so they can locate other timestamps between them. Another possibility is periodic publication of timestamps in authentic media such as newspapers.

If the TSA cannot find collisions for the hash function H it cannot forward-date or back-date timestamps along the chain. A possible attack would be by producing a fake chain of timestamps which is longer than a suspicious challenger will verify. In [Jus97] an attack is suggested that generates a fake sub-chain. To prevent these attacks timestamps have to be located relative to trusted items in the chain.

The problem with this scheme is that we need the cooperation of the other users to produce their timestamps which allow to verify the chain. Alternatively the TSA could store all timestamps and keep them available for everyone but the storage requirement would be prohibitive.

Extended linking The protocol can be modified by linking each timestamp with the k previous and next ones ([HS91]). The cooperation of only a subset of these users is required, but the storage space significantly increases.

Renewing old timestamps In the event that a hash function used by a timestamping service is broken, all of the timestamps issued prior to that time would be called into question. But if timestamps are renewed with a more secure hash function before the original one is broken, the time evidence contained in the original certificate will still be valid ([BHS92]). It is possible to use two hash functions in parallel so that the system remains secure if one of them is broken.

A NOTE ON INFRASTRUCTURES

Time Stamp Authorities can be incorporated in a public key infrastructure. *Certificate Authorities* can offer timestamping as an additional feature or the TSA's can be seperate entities. A TSA which also authenticates the clients and verifies the contents of the submitted documents is called a *Notary Authority.* Entities can be structured in a hierarchical tree and cross-certification between seperate trees is possible. In the timestamping scheme described above TSA's which want to be able to verify each other's timestamps have to link their chains together periodically.

ACCUMULATING TIMESTAMPS

The previous method which links all timestamps in a linear way poses a very high demand on cooperation and may impose a long computation time before a trusted timestamp is encountered on the chain. The main idea in accumulated hashing is to divide the timestamping procedure into rounds. At the end of each round a timestamp is calculated which depends on all requests submitted during that round and on the timestamp of the previous round. The measure of time now is the length of a round, which will have to be chosen according to the usage of the system and the requirements of the application. In general the stamping procedure works as follows:

1. The client ID_i sends the hash of his document to the TSA: $y_i = h(X_i)$.
2. The TSA accumulates all requests during one round (y_i for $1 \leq i \leq m$) and computes a timestamp based on all these values and on the timestamp of the previous round.
3. The TSA sends to each client the timestamp for the round and data necessary for reconstructing this timestamp. This is the timestamp for his document.

These timestamps can be verified as follows:

1. Compute the hash of the document and check that it is part of the data that reconstructs the timestamp for the round.
2. Check the linking between the timestamp for this round and timestamps for preceeding or succeeding rounds (they can be kept available by the TSA) until a trusted value is encountered. This should be a value published in authentic widely available media such as newspapers.

Timestamps for a round can be constructed in an efficient way with one of the following methods:

Building a binary authentication tree As explained in [BM91] and [BHS92] all timestamping requests accumulated during a round can be organized into a binary tree structure. The leaves of this tree are the values y_i. They are concatenated by two and hashed to obtain a parent value: $Z_{12} = H(y_1, y_2)$, etc. At the next level these values (internal nodes of the tree) are again concatenated by two and hashed, for example $Z_{14} = H(Z_{12}, Z_{34})$. This process continues until we obtain one value, the root of the tree, which is combined and hashed with the timestamp of the previous round to form the timestamp of the current round. When necessary, nodes with random or default values are inserted into the tree so that there is an even number of nodes at each level.

The timestamp of a particular document (represented by a hash value y_j) from one of the clients consists of the information necessary to rebuild the branch of the tree to which y_j belongs. This produces the root value and the timestamp of the round. Such a branch contains $\mathcal{O}(\log m)$ values (with m the number of requests during the round), the computation time is also $\mathcal{O}(\log m)$. This is superior to a straightforward linear hashing of values y_i where we would require $\mathcal{O}(m)$ operations. The tree is unforgeable if the hash function H is collision-resistant.

All requests submitted in the same round will have the same temporal measure, but if the TSA builds the tree by applying the values y_i in a chronological way, the relative positions for the corresponding requests can be preserved. For this however we need unconditional trust in the TSA. An implementation of this method is described in [MAQ99].

Using a one-way accumulator In [BM94] a method is proposed where the size of timestamps and computation time are constant with respect to the number of requests m. For this we need a primitive called one-way accumulator. It is a kind of hash function that accumulates an arbitrary number of values in such a way that the order in which these values are processed is irrelevant. The function must also be impossible to invert. This is achieved by the modular exponentiation operation.

The timestamp for a round is constructed starting from a random or default value Z_0 as follows: $Z = Z_0^{\prod_{i=1}^{m} y_i} \bmod N$. The timestamp for the document

corresponding to the hash y_j consists of the values $Z_j = Z_0^{\prod_{i=1, i\neq j}^{m} y_i} \bmod N$ and y_j. The timestamp for the round is reconstructed by the operation $Z = Z_j^{y_j} \bmod N$. (The timestamps are also linked with each other by combining and hashing with the timestamp of the previous round.)

To produce a fake timestamp for a value y' one would have to find a value Z' so that $Z'^{y'} = Z \bmod N$. If N is chosen as the product of two distinct primes this is equivalent to solving the RSA problem (which is believed to be as difficult as integer factorization). The modulus has to be chosen large enough (for example 1024 bits), additionally it is suggested in [BM94] to construct it as a rigid integer in order to prevent the reduction of the size of the image after repeated applications of the function.

In this scheme it is no longer possible to preserve the relative positions of the requests submitted in the same round (they are simultaneous). Another drawback is that modular exponentiations are slower than hash functions.

Also note that the accumulator function has a trapdoor, namely the factorization of N. The trapdoor function has to be provided by a trusted third party (which can be off-line). In [Nyb96] another accumulator function is proposed which has no trapdoor. However the size of the resulting timestamps seems to be prohibitively large. An analysis of accumulators is made in [Mas98].

Binary linking schemes The most recent scheme for timestamping was proposed in [BLLV98]. It is a mixture of the linear linking scheme and the accumulation method using trees. Each timestamp is linked with the previous one and also with another suitably chosen timestamp. Rounds are constructed with a fixed number of requests m (rather than a fixed time slot). The main advantage of the scheme is that the relative position of two arbitrary requests can be determined (without trusting the TSA) even if they were issued in the same round. The verification time and size of the timestamps are $\mathcal{O}(\log m)$.

CONCLUSION

The interaction between time and cryptography is a recent research subject. We have stressed the importance of time in non-repudiation services and more specific applications. We then described basic mechanisms for providing temporal authentication of electronic documents and discussed the advantages and limitations of several state-of-the-art techniques. Since standardization in this area (by

organizations such as the IETF) is still in early stages further developments and improvements in these systems can be expected.

REFERENCES

[BHS92] D. Bayer, S. Haber, W.-S. Stornetta, "Improving the efficiency and reliability of digital time-stamping", *Methods in Communication, Security, and Computer Science - Sequences '91*, Springer-Verlag, 1992, pp. 329–334.

[BM91] J. Benaloh, M. de Mare, "Efficient broadcast time-stamping", *Clarkson University, Department of Mathematics and Computer Science, TR-91-1*, August 1991.

[BM94] J. Benaloh, M. de Mare, "One-way accumulators: a decentralized alternative to digital signatures", *Advances in Cryptology - Eurocrypt '93, LNCS 765*, T. Helleseth, Ed., Springer-Verlag, 1994, pp. 274–285.

[BLLV98] A. Buldas, P. Laud, H. Lipmaa, J. Villemson, "Time-stamping with binary linking schemes", *Advances in Cryptology - Crypto '98, LNCS 1462*, H. Krawczyk, Ed., Springer-Verlag, 1998, pp. 486–501.

[HS91] S. Haber, W.-S. Stornetta, "How to time-stamp a digital document", *Journal of Cryptology*, vol. 3, no. 2, 1991, pp. 99–112.

[Jus97] M. Just, "Some timestamping protocol failures ", *Carleton University, School of Computer Science, TR-97-16*, August 1997.

[MQ97] H. Massias, J.-J. Quisquater, "Time and cryptography", *Technical report, Timesec project (Federal government project, Belgium)*, March 1997. Available at `http://www.dice.ucl.ac.be/crypto/TIMESEC.html`.

[Mas98] H. Massias, "A survey of accumulators", *Université Catholique de Louvain, Crypto group, Technical report*, February 1998.

[MAQ99] H. Massias, X. Serret Avila, J.-J. Quisquater, "Design of a secure timestamping service with minimal trust requirement", *Proceedings 20th Symposium on Information Theory in the Benelux*, this volume.

[Nyb96] K. Nyberg, "Fast accumulated hashing", *Fast Software Encryption '96, LNCS 1039*, D. Gollmann, Ed., Springer-Verlag, 1996, pp. 83–88.

[PRQ+98] B. Preneel, B. Van Rompay, J.-J. Quisquater, H. Massias, X. Serret Avila, "Design of a timestamping system", *Technical report, Timesec project (Federal government project, Belgium)*, 1998. To be available at `http://www.dice.ucl.ac.be/crypto/TIMESEC.html`.

[RPV98] B. Van Rompay, B. Preneel, J. Vandewalle, "On the security of dedicated hash functions", *Proceedings 19th Symposium on Information Theory in the Benelux*, Veldhoven, NL, May 28-29, 1998, pp. 103–110.

[NTP] The Network Time Protocol, `http://www.eecis.udel.edu/~ ntp`.

DESIGN OF A SECURE TIMESTAMPING SERVICE WITH MINIMAL TRUST REQUIREMENT

H. Massias, X. Serret Avila, J.-J. Quisquater
UCL Crypto group
Place du Levant, 3 , B-1348 Louvain-la-Neuve, Belgium
massias, serret, jjq@dice.ucl.ac.be

This paper presents our design of a timestamping system for the Belgian project TIMESEC. We first introduce the timestamping method used and we justify our choice for it. Then we present the design of our implementation as well as some of the important issues we found and the solutions we gave to them.

INTRODUCTION

The creation date of digital documents and the times expressed in them are becoming increasingly important as digital documents are being introduced into the legal domain.

We define "digital timestamp" as a digital certificate intended to assure the existence of a generic digital document at a certain time.

In order to produce fully trusted timestamps, very specific designs have been introduced. We give an overview of the most relevant methods and we introduce the one we used for the implementation of the Belgian project TIMESEC (see [PRQ+98]), justifying our choice for it. Then we present the design of the timestamping system we made for this project. We separate the different processes that are: document timestamping, timestamp verification, auditing, system start-up and system shutdown.

INTRODUCTION OF THE TIMESTAMPING TECHNIQUES

There are two families of timestamping techniques: those that work with a trusted third party and those that are based on the concept of distributed trust. Techniques based on a trusted third party rely on the impartiality of the entity that is in charge of issuing the timestamps. Techniques based on the distributed trust consist on making documents dated and signed by a large set of people in order to convince the verifiers that we could not have corrupted all of them. The trusted third party techniques can also be classified into two different kinds: those where the third party is completely trusted and those where it is partially

trusted. A detailed study of timestamping techniques can be found in [MQ97]. We believe that techniques based on distributed trust are not really workable in a professional environment, that is why we concentrate on the trusted third party approach. Nevertheless, we imposed to ourselves the requirement to lower the necessary trust on the third party to the maximum extend.

The "easy" solution, which consists on concatenating the document with the current time and sign the result, has been discarded because it has two main drawbacks:

1. We must completely trust the third party, called Secure Timestamp Authority (STA), which can issue undetectable back-dated timestamps.
2. The limited lifetime of cryptographic signatures, which can be shorter than the document time-to-life.

The timestamping method that we have chosen uses a binary tree structure and has been described in [HS91] and [HS97]. This method works by rounds. For each round a binary tree is constructed with the requests filled during it. The rounds have a fixed duration, which is the result of a trade-off between the timestamps accuracy and the number of requests submitted. In Figure 1 we can see a graphical representation of a round constructed using this method.

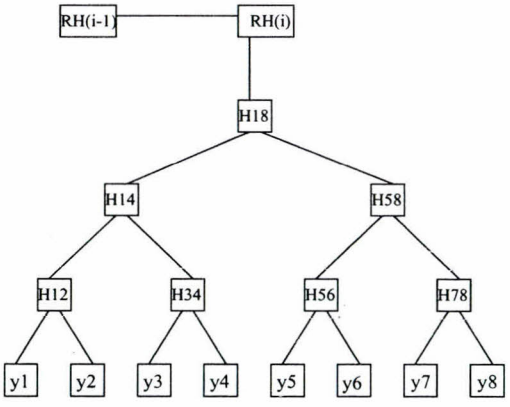


Figure 1: The binary tree structure

Each of the timestamp requests consists on a hash value of a given document. The leafs of the tree are each of those hash values. The leaf values are then

concatenated by two and hashed again to obtain the parent value (Ex: $H_{34} = H(y_3 \mid y_4)$). The process is repeated for each level until a single value is obtained. Finally, the top value of the round tree (H_{18}), called the "Round Root Value", is then concatenated with the value obtained for the preceding round (RH_{i-1}) and then hashed again to obtain the actual "Round Value" (RH_i).

The timestamp of the document contains all the values necessary to rebuilt the corresponding branch of the tree. For example, the timestamp for y_4 contains $\{(y_3, L), (H_{12}, L), (H_{58}, R), (RH_{i-1}, L)\}$. The verification process consists of re-building the tree's branch and the linking chain of "Round Values" until a trusted (from the verifier point of view) "Round Value" is recomputed. This verification method is explained in detail in [HS91] and [MQ97].

Periodically, one of the "Round Values" is published on an unmodifiable, widely witnessed media (Ex: newspaper...). These special "Round Values", which we will call "Big Round Values", are the base of the trust for all the timestamps issued. All verifiers must trust these "Big Round Values" as well as the time associated with them. This is a reasonable requirement because those values are widely witnessed. The absolute time trusted by all the potential verifiers is the time indicated by the unmodifiable media. We suppose that this time is the same than the time indicated by the STA for the "Big Round". Forcing the clients to check the timestamps as soon as they get them is another requirement. In that way the process is continuously audited and the STA will not have any margin to maneuver in an untrusted way.

A very useful method for extending the lifetime of timestamps is described in [BHS92]. It basically consists on re-timestamping the hash of the document as well as the original timestamp before the hash function is broken.

We build two trees in parallel for each round using two different hash functions (SHA-1 and RIPEMD-160). In that way, the system remains secure in the case of an unexpected break of one of the hash functions used.

DESCRIPTION AND ANALYSIS OF THE TIMESEC TIMESTAMPING IMPLEMENTATION

We will now introduce the basic design of the system we have developed, which is based on the technique introduced above.

Initially, the user designates a document to be timestamped. Two hashes of it are created using the SHA-1 and RIPEMD-160 algorithms. The request

containing the two hashes is then sent by the client to the STA . Upon request receipt, the STA creates the corresponding timestamp using the following process.

Main description of the timestamping process

The system design follows a highly decoupled multi-threaded approach. Each step is assigned to a specific component, which has its own different thread. In the Figure 2 we present a schematic outline of the process. The multi-thread approach is justified by the requirement to obtain a highly responsive and load independent implementation. By isolating the process charges into independent steps we try to decouple the load between them. Each step has also a working queue. Those queues are in charge of softening the speed differences between the different process steps.

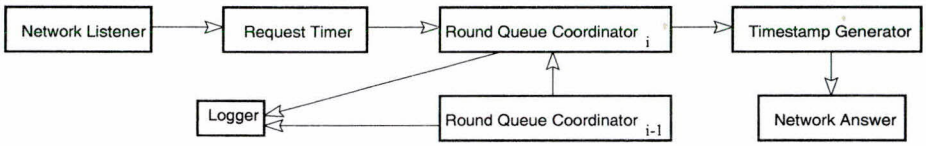


Figure 2: Interactions between the components

The "Network Listener" is in charge of continuously listen to the clients' timestamp requests. The "Request Timer" receives the constructed requests from the "Network Listener". Then, it times and forwards them to the actual "Round Queue Coordinator". Each round has its own "Round Queue Coordinator", which is in charge of compiling and processing into a tree all the requests belonging to the round. When the round tree has been computed it is forwarded to the "Timestamp Generator", which generates the corresponding timestamps. Once a timestamp is generated, the "Timestamp Generator" forwards it to the "Network Answer", which in turn forwards it to the client.

The Network Listener

The "Network Listener" responsibility is to listen the network continuously for timestamping requests. When it receives a data stream, the "Network Listener" checks it in order to determine if it is a valid request. In the case it is, it sends an affirmative contact response to the client, it creates a "Timestamp Request" object and adds it to the "Request Timer" queue. Then it goes back to listen

to the network. In the case the request message is not correct, it sends an error message to the client.

We tried to give as few tasks as possible to the "Network Listener" to let it listen the network, which is its primary task. In order to improve the overall performance, and to avoid the fact that a slow client connection could affect the other ones, several copies of the "Network Listener" can be active at the same time.

The Request Timer

There is only an instance of "The Request Timer" in the system. The "Request Timer" is in charge of ordering the requests received from the several "Network Listeners" and timing them accordingly. All delays introduced by the system before that point (namely, those introduced by the "Network Listener") are indistinguishable from network delays, and thus not taken into account. Once a request has been timed, the "Request Timer" tries to add it to the current round queue. As the rounds are closed asynchronously by the corresponding "Round Queue Coordinator" this operation is not always successful, in that case, the "Request Timer" re-times the request and retries to queue it until it finds an open round. In that process the request sequence is preserved in order to provide a consistent behavior.

Round Queue Coordinator creation: "Round Queue Coordinator" instances are created by the "Request Timer" upon processing a request corresponding to a non-existing round. The creation of the rounds that have no requests is delayed until a request is received. Once created, those empty rounds are immediately processed, introducing no significant delay into the process.

Round number determination: Round numbers form a non-interrupted increasing integer sequence. Rounds are always in synchronization with the round duration intervals. In other words, if the round duration is one minute, all rounds will start in an absolute minute boundary, independently from when the system has been started. "Big Rounds" are determined by the "Request Timer" using a similar approach to the one followed to determine the round boundaries. We do not restrict the duration of the round to a fixed value for the lifetime of the STA. To achieve this, the information about round and "Big Round" duration is introduced into the system at the start-up phase. If we wish to modify it, we must

first shutdown the system, change the values and then restart the system, which is the only safe procedure we had foreseen.

The Round Queue Coordinator

The first thing a "Round Queue Coordinator" does is to determine the offset between the actual time and the round due time. Requests will be accepted only if the round is still valid (round is open). When requested by the "Request Timer", the "Round Queue Coordinator" adds the request to the queue and logs it. This logged request will be latter used for process auditing purposes.

When the round time is over, it obtains the "Round Values" from the preceding round and it computes the round binary trees (one for each hash algorithm) to obtain the corresponding "Round Values". Then it gives the computed trees to the "Timestamp Generator" and finally adds to the log the "Round Values" and the "Round Root Values". Those logged values will be latter used for timestamp verification and process auditing purposes. If the actual round is a "Big Round" those values are forwarded to a fixed media as well.

As you may have noticed in the section "Introduction of the timestamping techniques", the binary tree is defined for a number of leafs (requests) that is a power of 2. In general, this is not the case. We could create fake requests to finish the tree, but this will add a lot of requests (if we have 2^n+1 requests, then we will need to add $2^n - 1$ fake requests). A smarter solution is to add a random value only when we need it. Then, we add at most n values (one for each level of the tree). We call these nodes "Special Node", which will be logged as well. Instead of random values we could choose to use 0 or another fixed value, this would be as secure as our choice if the hash functions were "perfect". As hash functions are only "presumably perfect", we though that we could made our design more secure with really few additional computations.

In our implementation, the STA queues the requests and computes the tree at the end of the round. At first sight, it could seem a more natural solution to build the tree as soon as the requests arrive. At the end of the round, the computation of the tree would then be ended by getting the last "Round Value" and computing the actual "Round Value". In fact, this solution is harder to implement, and has no effect on the security achieved as no one can check that the STA does not perform any reordering of the requests before it publishes the "Round Value".

The Timestamp Generator

The "Timestamp Generator" processes the round trees by pairs (one for each hash algorithm) in order to generate the timestamps for each of the requests contained in the trees. In order to maximize the system responsiveness, once a timestamp has been generated it is immediately forwarded to the "Network Answer". Finally, when all the timestamps contained in a round tree have been processed the tree is destroyed.

The Network Answer

The "Network Answer" is in charge of forwarding the processed timestamps to the clients. It has been specified in such a way that it can run several threads, in that way the rest of the timestamping process can be isolated from possible network delay problematic.

The timestamp verification process

First, the verifier designates a document and its corresponding timestamp for verification. Then, the verifier's system (his personal computer or a remote computer independent from the STA) generates the two document hashes and checks if they match with those contained in the timestamp. Afterwards, the "Round Value" is reconstructed using the data provided in the timestamp. If the computed "Round Value" is consistent with the one contained in the timestamp then the next step in the verification process is to compare this "Round Value" to the "Round Value" obtained from the STA repository. Finally, the verifier provides his system with the two "Big Round Values" that he founds in the "unmodifiable media"; the verifier's system gets all the necessary "Round Values" and "Root Round Values" from the STA and it checks the coherency of the two linking chains (one for each hash function).

The audit process

The auditor designates two "Big Rounds", which he fetches from a fixed media. The system behavior will be checked between these two "Big Round Values". For each round, the auditor's system gets all the hash values (leafs of the tree and "Special Nodes") and the "Round Value" from the STA. Then, it constructs the two trees and checks that the "Round Value" is consistent. These two steps are

repeated until all the considered rounds are checked or until an error has been found. In that way, all theoretically verifiable system behavior can be verified a posteriori.

The system start-up process

Here the most sensible issue is to be able to correctly start-up the system when an unexpected shutdown has occurred. If that is the case, the log will show an unfinished round; then the system marks all entries after the last complete round as invalid and publishes that round as a "Big Round". If the log was consistent, it accesses the last valid "Round Value" in the log and publishes it as a "Big Round". This process insures a fully verifiable behavior; we are able to detect non fully-processed requests.

The system shutdown process

The administrator signals the system to shutdown. No more timestamping requests are accepted. The system waits until the current round is finished and this "Round Value" is published as "Big Round".

REFERENCES

[BHS92] D. Bayer, S. Haber, and W.-S. Stornetta. Improving the efficiency and reliability of digital timestamping. In Springer Verlag, editor, *Sequences'91: Methods in Communication, Security, and Computer Science*, pages 329–334, 1992.

[HS91] S. Haber and W.-S. Stornetta. How to timestamp a digital document. *Journal of Cryptology*, 3(2):99–112, 1991.

[HS97] S. Haber and W.S. Stornetta. Secure names for bit-strings. In *Proceedings of the 4th ACM Conference on Computer and Communication Security*, pages 28–35. ACM Press, April 1997.

[MQ97] H. Massias and J.-J. Quisquater. Time and cryptography. Technical report, TIMESEC Project (Federal Governement Project, Belgium), 1997. Available at `http://www.dice.ucl.ac.be/crypto/TIMESEC.html`.

[PRQ+98] B. Preneel, B. Van Rompay, J.-J. Quiquater, H. Massias, and X. Serret Avila. Design of a timestamping system. Technical report, TIMESEC Project (Federal Governement Project, Belgium), 1998. To be available at `http://www.dice.ucl.ac.be/crypto/TIMESEC.html`.

CHARACTERIZATION OF THE SECRECY OF A COMMON KEY CONSTRUCTED VIA DATA TRANSMISSION OVER THE TWO–WAY "AND" CHANNEL

Vladimir B. Balakirsky

Eindhoven University of Technology, Electrical Engineering Department
(on leave from the Data Security Association "Confident", St.-Petersburg, Russia)
P.O.Box 513, 5600 MB Eindhoven, the Netherlands
v.b.balakirsky@ele.tue.nl, htpp://www.mathematik.uni-bielefeld.de/~vbal/

We address cryptology aspects of data transmission over the two–way AND channel, also called the multiplying channel, and solve the combinatorial problem of computing the conditional entropy of a common key that can be constructed by communicating persons when the sequence at the output of the channel is known to an eavesdropper.

STATEMENT OF THE PROBLEM AND FORMULATION OF THE RESULT

Problem : *Given* $n \geq 2$, *construct*

(a) a matrix of the probability distributions

$$\mathbf{P} = \begin{bmatrix} P_{X_1} & P_{X_2|X_1,Z_1} & \dots & P_{X_n|X^{n-1},Z^{n-1}} \\ P_{Y_1} & P_{Y_2|Y_1,Z_1} & \dots & P_{Y_n|Y^{n-1},Z^{n-1}} \end{bmatrix}$$

where $X_1, \dots, X_n$ *and* $Y_1, \dots, Y_n$ *are random variables taking values in* $\{0,1\}$ *and* $Z_1, \dots, Z_n$ *are uniquely determined by the pairs* $(X_1, Y_1), \dots, (X_n, Y_n)$ *in such a way that* $Z_t = X_t \wedge Y_t, \quad t = 1, \dots, n$;

(b) two deterministic functions $K, L : \{0,1\}^n \times \{0,1\}^n \to \mathcal{K}$, *where* $\mathcal{K}$ *is some finite set.*

The following requirements must be satisfied,

(R1) $P_{X^n,Y^n}(\mathbf{x},\mathbf{y}) > 0 \implies K(\mathbf{x}, \mathbf{x} \wedge \mathbf{y}) = L(\mathbf{y}, \mathbf{x} \wedge \mathbf{y})$, *for all* $\mathbf{x}, \mathbf{y} \in \{0,1\}^n$;

(R2) $H(K|Z^n) \longrightarrow \max$, *where*

$$H(K|Z^n) = -\sum_{key \in \mathcal{K}} \sum_{\mathbf{z} \in \{0,1\}^n} P_{K,Z^n}(key, \mathbf{z}) \log \frac{P_{K,Z^n}(key, \mathbf{z})}{P_{Z^n}(\mathbf{z})}$$

Table 1: The values of the function H_n/n for $n = 2^m$, where $m = 1, \ldots, 14$.

n	H_n/n	n	H_n/n
2	0.25	256	0.292274
4	0.261364	512	0.292583
8	0.273843	1024	0.292738
16	0.282995	2048	0.292816
32	0.287937	4096	0.292855
64	0.290415	8192	0.292874
128	0.291654	16384	0.292884
		∞	0.292893

and the probability distributions belong to the ensemble specified by the matrix $\mathbf{P}$.

Theorem : *Let H_n denote the maximal conditional entropy $H(K|Z^n)$ that can be achieved in the Problem. Then the function H_n can be recurrently found by*

$$H_n = (1-\xi_n)^2 H_{n-1} + 2\xi_n(1-\xi_n)(1+H_{n-2}) + \xi_n^2 H_{n-2}, \quad n = 2, 3, \ldots$$

where $H_0 = H_1 = 0$ and

$$\xi_n = \frac{1 - H_{n-1} + H_{n-2}}{2 - H_{n-1} + H_{n-2}}$$

The function H_n is bounded from below as

$$H_n \geq H_n^{(\xi)} = 2(1-\xi)\xi^{n-1} \cdot \left. \frac{\partial F_\xi^{(n)}(D, I_\xi)}{\partial D} \right|_{D=D_\xi} \tag{1}$$

where

$$\begin{aligned} F_\xi^{(n)}(D, I_\xi) &= \frac{1}{\xi} \sum_{\substack{n_1=0 \\ n_1+n \text{ odd}}}^{n-1} \binom{(n-1+n_1)/2}{n_1} (1+D)^{(n-1-n_1)/2} I_\xi^{n_1} \\ &+ \sum_{\substack{n_1=0 \\ n_1+n \text{ even}}}^{n-2} \binom{(n+n_1)/2-1}{n_1} (1+D)^{(n-n_1)/2} I_\xi^{n_1} \end{aligned}$$

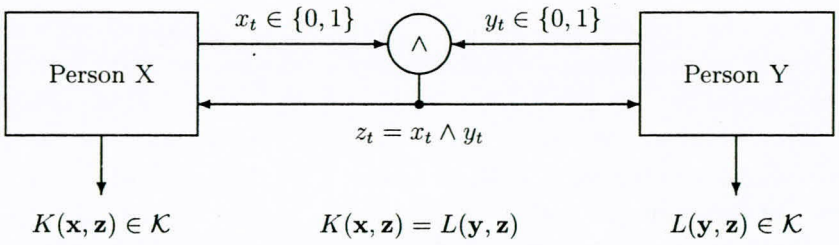


Figure 1: Constructing a common key via data transmission over the AND channel.

and

$$\xi = \sqrt{2} - 1, \quad D_\xi = \frac{2(1-\xi)}{\xi}, \quad I_\xi = \frac{(1-\xi)^2}{\xi} \tag{2}$$

Furthermore,

$$\lim_{n\to\infty} \frac{H_n}{n} = 1 - \frac{\sqrt{2}}{2} = 0.292893\ldots, \quad \lim_{n\to\infty} \xi_n = \sqrt{2} - 1 = 0.414214\ldots$$

Some values of the function H_n/n are given in Table 1.

CONSTRUCTING A COMMON KEY VIA DATA TRANSMISSION OVER THE TWO–WAY "AND" CHANNEL

The problem formulated above relates to data transmission over the two–way AND channel (see Figure 1) where the persons X and Y send bits x_t and y_t to each other and both of them receive $z_t = x_t \wedge y_t$ at time instants $t = 1, \ldots, n$. The bits x_t and y_t transmitted by the persons may depend on $x_1, \ldots, x_{t-1}, z_1, \ldots, z_{t-1}$ and $y_1, \ldots, y_{t-1}, z_1, \ldots, z_{t-1}$, respectively. Communication over such a channel when the persons want to exchange messages was introduced by Shannon [1] : in this case, x_t and y_t also depend on the messages of the persons X and Y. The problem of finding the region containing all pairs of the maximal possible numbers of messages and the corresponding transmission algorithms was considered in [2],[3],[4],[5], and other papers, but this problem is still unsolved. We address a different problem concerning the two–way channels that belongs to the direction known as constructing the common randomness (see [6],[7],[8], and other papers) when the goal of the persons is the creating a common random process for some further cryptology purposes. In our setup, the person X computes $K(\mathbf{x},\mathbf{z})$ and the person Y computes $L(\mathbf{y},\mathbf{z})$, where the functions K and L are fixed in advance.

If these values coincide for all pairs $(\mathbf{x}, \mathbf{y})$ that can be exchanged over a channel, then they serve as pointers to a table of keys which was also fixed in advance. An eavesdropper has only $\mathbf{z}$ and does not know $\mathbf{x}, \mathbf{y}$. His ambiguity about the pointer is measured by the conditional entropy $H(K|Z^n)$. For the notational convenience, we will associate the values of the functions K and L computed by the persons with the corresponding keys.

The actions of the persons can be represented as some realization of the algorithm consisting of *encoding* (the protocol for data transmission) and *decoding* (computing $K(\mathbf{x}, \mathbf{z})$ and $L(\mathbf{y}, \mathbf{z})$). We investigate the specific communication algorithm that will be referred to as *the H–type algorithm* since it is similar to the Hagelbarger's algorithm for the transmission of messages over the two–way AND channel ([1], p.375).

DESCRIPTION OF THE H–TYPE ALGORITHM

We assume that the persons independently assign one of two values to some auxiliary variable S_t at time instant t in such a way that these values coincide; thus, S_t can be interpreted as the current state of the communication system and we denote the possible states by α and β. The line above in the description of the algorithm denotes the binary inversion, i.e., $\overline{0} = 1$ and $\overline{1} = 0$.

Encoding : the persons X and Y set $S_1 = \alpha$ and follow the procedure below for $t = 1, \ldots, n$, where $\xi_n, \ldots, \xi_1 \in (0, 1)$ are given.

(α) If $t < n$ and $S_t = \alpha$, then the persons X and Y choose x_t and y_t as independent random variables taking values 0 and 1 with the probabilities ξ_{n-t+1} and $1 - \xi_{n-t+1}$, respectively; both persons set $S_{t+1} = \alpha$ if $z_t = 1$ and $S_{t+1} = \beta$ if $z_t = 0$. If $t = n$ and $S_t = \alpha$, then the person X sets $x_t = 1$ and the person Y sets $y_t = 1$.

(β) If $S_t = \beta$, then the person X sets $x_t = \overline{x}_{t-1}$ and the person Y sets $y_t = \overline{y}_{t-1}$; both persons set $S_{t+1} = \alpha$.

Decoding : the persons X and Y set $t = 1$ and follow the procedure below while $t \leq n$.

(1) If $z_t = 1$, then the person X sets $\hat{y}_t = 1$ and the person Y sets $\hat{x}_t = 1$; both persons increase t by 1.

(00) If $(z_t, z_{t+1}) = (0, 0)$, then the person X sets $(\hat{y}_t, \hat{y}_{t+1}) = (\overline{x}_t, 0)$ and the person Y sets $(\hat{x}_t, \hat{x}_{t+1}) = (\overline{y}_t, 0)$; both persons increase t by 2.

(01) If $(z_t, z_{t+1}) = (0,1)$, then the person X sets $(\hat{y}_t, \hat{y}_{t+1}) = (0,1)$ and the person Y sets $(\hat{x}_t, \hat{x}_{t+1}) = (0,1)$; both persons increase t by 2.

The keys are defined as

$$K(\mathbf{x},\mathbf{z}) = [\ (x_1,\hat{y}_1),\ldots,(x_n,\hat{y}_n)\ ],\quad L(\mathbf{y},\mathbf{z}) = [\ (\hat{x}_1,y_1),\ldots,(\hat{x}_n,y_n)\ ].$$

The proof of the theorem includes the statement : *the maximal conditional entropy of a common key H_n is attained using the H–type algorithm.* The algorithm with $\xi_1 = \ldots = \xi_n = \xi$, where ξ is fixed, will be referred to as *the $H^{(\xi)}$–type algorithm* and we present its nonasymptotic analysis leading to (1).

NONASYMPTOTIC ANALYSIS OF THE $H^{(\xi)}$–TYPE ALGORITHM

We will associate data transmission over the two–way AND channel specified by the $H^{(\xi)}$–type algorithm with the parsing of binary sequences when the sequence is represented as a concatenation of codewords of the prefix code $\{00, 01, 1\}$. The code tree is given in Figure 2. A particular binary sequence uniquely determines the path which follows the encoder by returning to the root as soon as he reaches some terminal node. Let us label the nodes of the tree by α and β in such a way that the root and all terminal nodes have the label α. We also introduce two formal variables, D and I, assign the generating function to every edge of the tree as it is shown in Figure 2, and define the generating function of each path of the encoder as the product of the generating functions of all edges of that path. For example, the codewords 00, 01, 1 have the generating functions $D, 1, I$, and the generating function of the sequence 001011011, which is parsed as $(00, 1, 01, 1, 01, 1)$, is equal to DI^3.

For all $j = 0,\ldots,n-1$, let $T_\alpha^{(j)}(D,I)$ and $T_\beta^{(j)}(D,I)$ denote the sums of generating functions of all binary sequences of length j when the encoder reaches the nodes labeled by α and β at the j–th time instant, respectively. Let us also denote

$$\mathbf{T}^{(j)}(D,I) = \left(T_\alpha^{(j)}(D,I), T_\beta^{(j)}(D,I)\right),\quad j = 0,\ldots,n-1.$$

Then, as it is easy to see,

$$\begin{aligned} \mathbf{T}^{(0)}(D,I) &= (1,0) \\ \mathbf{T}^{(j)}(D,I) &= \mathbf{T}^{(j-1)}(D,I)\cdot \begin{bmatrix} I & 1 \\ 1+D & 0 \end{bmatrix},\quad j = 1,\ldots,n-1. \end{aligned} \tag{3}$$

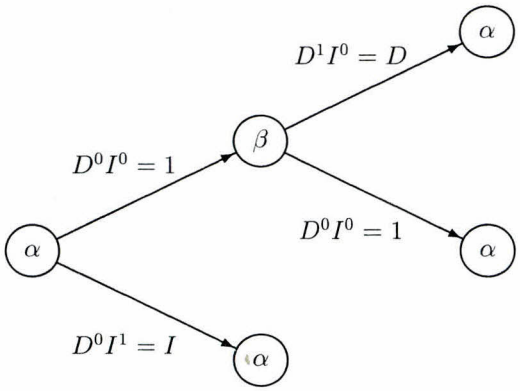


Figure 2: Tree of the prefix code $\{00, 01, 1\}$ with the nodes labeled by α, β and the generating functions assigned to the edges.

Lemma : *Let*

$$T^{(n)}(D, I) = \frac{1}{\xi} \cdot T_{\alpha}^{(n-1)}(D, I) + (1 + D) \cdot T_{\beta}^{(n-1)}(D, I). \tag{4}$$

Then

$$H_n = 2(1 - \xi)\xi^{n-1} \cdot \left. \frac{\partial T^{(n)}(D, I)}{\partial D} \right|_{\substack{D=D_\xi \\ I=I_\xi}} \tag{5}$$

where D_ξ and I_ξ are defined in (2).

Proof : Let $\mathcal{Z}_{\alpha}^{n}(d, i)$ and $\mathcal{Z}_{\beta}^{n}(d, i)$ denote the set of binary sequences of length n which lead the encoder of the prefix code to the nodes labeled by α and β at time instant n and have the property that the parsing procedure, being applied to any of these sequences, generates d codewords 00, i codewords 1, and $n_{01}(\mathbf{z})$ codewords 01, where

$$n_{01}(\mathbf{z}) = \begin{cases} (n - i)/2 - d, & \text{if } \mathbf{z} \in \mathcal{Z}_{\alpha}^{n}(d, i) \\ (n - 1 - i)/2 - d, & \text{if } \mathbf{z} \in \mathcal{Z}_{\beta}^{n}(d, i). \end{cases} \tag{6}$$

On the other hand, as it follows from the description of the encoding of the $H^{(\xi)}$–type algorithm, any sequence belonging to the set $\mathcal{Z}_{\alpha}^{n}(d, i) \cup \mathcal{Z}_{\beta}^{n}(d, i)$ appears at the output of the AND channel if and only if there are $d + i + n_{01}(\mathbf{z})$ time instants when the persons decide that α is the current state of the communication system and they generate d pairs $(x_t, y_t) \in \{(0, 1), (1, 0)\}$, i pairs $(x_t, y_t) = (1, 1)$, and

$n_{01}(\mathbf{z})$ pairs $(x_t, y_t) = (0,0)$ at these time instants. Therefore

$$P_{Z^n}(\mathbf{z}) = (2\xi(1-\xi))^d \cdot \left((1-\xi)^2\right)^i \cdot \left(\xi^2\right)^{n_{01}(\mathbf{z})}, \quad \text{for all } \mathbf{z} \in \mathcal{Z}_\alpha^n(d,i) \bigcup \mathcal{Z}_\beta^n(d,i). \tag{7}$$

Furthermore, since (1) any of these sequences leads to one of 2^d keys constructed by the persons X and Y; (2) the probability distribution on the set of keys is uniform,

$$H(K|\mathbf{z}) = d, \quad \text{for all } \mathbf{z} \in \mathcal{Z}_\alpha^n(d,i) \bigcup \mathcal{Z}_\beta^n(d,i). \tag{8}$$

Equations (6)–(8) and notations (2) imply

$$H_n = \xi^n \sum_{d,i} \left(|\mathcal{Z}_\alpha^n(d,i)| + \frac{1}{\xi} \cdot \left|\mathcal{Z}_\beta^n(d,i)\right| \right) D_\xi^d I_\xi^i \cdot d. \tag{9}$$

The cardinalities of the sets $\mathcal{Z}_\alpha^n(d,i)$ and $\mathcal{Z}_\beta^n(d,i)$ can be expressed using the coefficients of the generating functions $T_\alpha^{(n-1)}(D,I)$ and $T_\beta^{(n-1)}(D,I)$: if these functions are represented as finite sums

$$T_\alpha^{(n-1)}(D,I) = \sum_{d,i} t_\alpha^{(n-1)}(d,i) D^d I^i, \quad T_\beta^{(n-1)}(D,I) = \sum_{d,i} t_\beta^{(n-1)}(d,i) D^d I^i$$

then

$$|\mathcal{Z}_\alpha^n(d,i)| = t_\beta^{(n-1)}(d,i) + t_\beta^{(n-1)}(d-1,i), \quad |\mathcal{Z}_\beta^n(d,i)| = t_\alpha^{(n-1)}(d,i) \tag{10}$$

where the first equation follows from the fact that the last codeword after the parsing of any sequence $\mathbf{z} \in \mathcal{Z}_\alpha^n(d,i)$ is either 00 or 01, and the second equation follows from the fact that the last bit of any sequence $\mathbf{z} \in \mathcal{Z}_\beta^n(d,i)$ is 0 and that the last codeword constructed by the parsing procedure ends at time instant $n-1$. By (9) and (10),

$$H_n = \xi^n \sum_d \sum_i \left(t_\beta^{(n-1)}(d,i) + t_\beta^{(n-1)}(d-1,i) + \frac{1}{\xi} \cdot t_\alpha^{(n-1)}(d,i) \right) D_\xi^d I_\xi^i \cdot d. \tag{11}$$

Since (4) defines the generating function

$$T^{(n)}(D,I) = \sum_{d,i} t^{(n)}(d,i) D^d I^i$$

where $t^{(n)}(d,i) = t_\alpha^{(n-1)}(d,i)/\xi + t_\beta^{(n-1)}(d,i) + t_\beta^{(n-1)}(d-1,i)$, equation (11) can

be presented as (5). Q.E.D.

Using (3) and induction on $j = 0, \ldots, n-1$ we prove that

$$T_{\alpha}^{(n-1)}(D,I) = \sum_{\substack{n_1=0 \\ n_1+n \text{ odd}}}^{n-1} \binom{(n-1+n_1)/2}{n_1} (1+D)^{(n-1-n_1)/2} I^{n_1}$$

$$T_{\beta}^{(n-1)}(D,I) = \sum_{\substack{n_1=0 \\ n_1+n \text{ even}}}^{n-2} \binom{(n+n_1)/2-1}{n_1} (1+D)^{(n-n_1)/2-1} I^{n_1}$$

where the second sum is understood as 0 when $n = 1$, and using (4), (5) obtain (1).

REFERENCES

[1] C.E. Shannon, "Two-Way Communication Channels", in *Claude Elwood Shannon : Collected Papers.* N.J.A. Sloane and A.D. Wyner, Eds. IEEE Press, pp. 351–384, 1993.
The paper was published in the Proc. 4–th Berkley Symp. Math. Stat. and Prob., pp.611–644, 1961.

[2] G. Dueck, "The Capacity Region of the Two–Way Channel Can Exceed the Inner Bound", Inform. Control, 40(3), pp.258–266, 1979.

[3] J.P.M. Schalkwijk, "The Binary Multiplying Channel – A Coding Scheme That Operates Beyond the Shannon Inner Bound", IEEE Trans. Inform. Theory, IT–28(1), pp.107–110, 1982.

[4] J.P.M. Schalkwijk, "On an Extension of an Achievable Rate Region for the Binary Multiplying Channel", IEEE Trans. Inform. Theory, IT–29(3), pp.445–448, 1983.

[5] Z. Zhang, T. Berger, and J.P.M. Schalkwijk, "New Outer Bounds to Capacity Regions of Two–Way Channels", IEEE Trans. Inform. Theory, IT–32(3), pp.383–386, 1986.

[6] R. Ahlswede and B. Verboven, "On Identification via Multiway Channels with Feedback", IEEE Trans. Inform. Theory, IT–37(5), pp.1519–1526, 1991.

[7] U. Maurer, "Secret Key Agreement by Public Discussion from Common Information", IEEE Trans. Inform. Theory, IT–39(3), pp.733–742, 1993.

[8] R. Ahlswede and I. Csiszár, "Common Randomness in Information Theory and Cryptography – Part I : Secret Sharing", IEEE Trans. Inform. Theory, IT–39(5), pp.1121–1131, 1993.

An attack against the Alabbadi–Wicker scheme

Sheng-bo Xu Jeroen Doumen
Department of Mathematics and Computing Science
Eindhoven University of Technology, P.O. Box 513
5600 MB Eindhoven, the Netherlands
E–mail: {sbxu, doumen } @win.tue.nl

Abstract

In this paper we discuss the security of the Alabbadi–Wicker scheme. An example is presented to demonstrate a universal forgery attack against the Alabbadi–Wicker scheme.

Keywords: Digital Signatures, Error–Correcting Codes.

1 Introduction

In 1994, Alabbadi and Wicker presented a digital signature scheme based on error–correcting codes [4]. Here we will refer to it as the Alabbadi–Wicker scheme. Alabbadi and Wicker analyzed the security of the Alabbadi–Wicker scheme and claimed that *their scheme derives its security from the complexity of three problems: the decoding of general linear error–correcting block codes [5], the factoring of large matrices and the derivation of a matrix from its right inverse.* In this paper, we will present a universal forgery attack against this scheme. The outline of this paper is as follows: we will start by giving a simple description of the Alabbadi–Wicker scheme. Next we will introduce our universal forgery attack for the Alabbadi–Wicker scheme. Finally, we will present some general comments about the security of digital signature schemes based on error–correcting codes.

2 The Alabbadi–Wicker scheme

The signer chooses a binary Goppa code C which can be described by an irreducible polynomial $G(z)$ of degree t. Next he selects a generation matrix G and parity check matrix H for this code. Furthermore he chooses two binary matrices W and V satisfying $W + V = G$ and such that the rank of W is k. Thus there exists a matrix W^{-R}, called

the right–inverse, such that $WW^{-R} = I_k$. Lastly, the signer chooses a nonsingular binary $n \times n$ matrix P. The public key of the signer consists of $G(z)$, W^{-R}, $H' = P^{-1}H^T$, $W' = P^{-1}W^{-R}$, $W'' = W^{-R}GW^{-R}$, t and t' (with $t' < t$). His private key consists of V, W, G, $W^{-R}G$, and P. In addition a nonlinear function $f(x, y) : GF(2^k) \times GF(2^n) \rightarrow GF(2^k)$ is chosen by the system operator.

To sign a k–bit message the signer randomly chooses a n–bit vector e of weight t' and a k–bit vector r of arbitrary (but nonzero) weight. The signature pair (x, s) is computed as follows.

$$x = \{rG + f(m, e)V\}P \quad (1)$$
$$s = \{e + f(m, e)W + xW^{-R}G\}P \quad (2)$$

The signer transmits s, x, and m to the receiver. The receiver may get a noise corrupted triple s', x' and m'. The receiver can validate the signature by performing the following steps:

- The syndrome of $x' + s'$ is calculated. Note that because of the relation $V + W = G$, and this syndrome satisfies $GH^T = 0$,

$$\begin{aligned}(x' + s')H' &= [\{rG + f(m, e)G + e + xW^{-R}G\}P]'P^{-1}H^T \\ &= e'H^T.\end{aligned}$$

- The Berlekamp–Massey algorithm is applied to the above syndrome to calculate the error vector e'. If $w(e') \neq t'$ the receiver requests a retransmission of x, s and m. Otherwise the receiver continues the validation process under the assumption that $e' = e$, $x' = x$ and $s' = s$.

- $f(m, e)$ is recovered from x, e, and s by computing

$$\begin{aligned}sW' + xW'' + eW^{-R} &= sP^{-1}W^{-R} + xW^{-R}GW^{-R} + eW^{-R} \\ &= f(m, e).\end{aligned}$$

- Finally $f(m, e)$ is compared with $f(m', e)$. If $f(m, e) = f(m', e)$ the signature pair is accepted.

3 A universal forgery attack against the Alabbadi–Wicker scheme

In the following, we will demonstrate a universal forgery attack against the Alabbadi–Wicker scheme by means of a typical example. The method itself will be presented in [9]. We will use the parameters of the Alabbadi–Wicker scheme [4].

$$G = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}, H = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}, W = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

$$V = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}, P = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, P^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

$$W^{-R} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}, H' = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, W' = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}, W'' = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Suppose we have succeeded in recovering the parity check matrix H. This is possible if we can obtain n signatures with independent error vectors (see [9] for how to achieve this).

First we can calculate $\tilde{P}^{-1}$ from the following equation

$$[H'|W'] = \tilde{P}^{-1}[H^T|W^{-R}] \tag{3}$$

and get

$$\tilde{P}^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Note that $\tilde{P}^{-1}$ is a full–rank matrix. We can invert $\tilde{P}^{-1}$ to get

$$\tilde{P} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Note that in general $\tilde{P} \neq P$, but the two matrices will play similar roles. Then we can calculate $\tilde{G}, \tilde{W}$ and Y from

$$\tilde{G}H^T = 0 \tag{4}$$

$$\tilde{W}W^{-R} = I_k \tag{5}$$

$$W'' = YW^{-R} \tag{6}$$

and arrive at

$$\tilde{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}, \tilde{W} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}, Y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

The matrix $\tilde{V}$ follows from the equation $\tilde{G} = \tilde{W} + \tilde{V}$

$$\tilde{V} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

Now we will forge a signature using the above matrices $\tilde{P}, \tilde{G}, \tilde{W}, \tilde{V}$ and Y. Here we still use the same example in [4] where $m = (001)$, $e = (000001)$, and $f(m, e) = (011)$ and forge a signature for this message.

We create a false signature by calculating the pair (x, s) from

$$x = \{f(m, e)\tilde{V} + xY\}\tilde{P} \tag{7}$$
$$s = \{e + f(m, e)\tilde{W} + xY\}\tilde{P} \tag{8}$$

to arrive at the forged signature pair $(x = 101111, s = 111100)$ of the message m. Note that we have to solve x from the first equation. If there are more solutions for x we simply choose one and proceed with calculating s. We now check whether the pair (x, s) passes the verification of the Alabbadi–Wicker scheme. We will suppose no errors occur in the channel.

First we check if $(x + s)H' = eH^T$:

$$(x+s)H' = (010011) \cdot \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = (001)$$

and

$$eH^T = (000001) \cdot \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = (001).$$

It is obvious that the receiver can obtain $e = (000001)$ from the signature pair (x, s). Thus the forged signature will pass the first three steps of the verification in the

Alabbadi–Wicker scheme. Now we show that the forged signature also passes the last step:

$$sW' + xW'' + eW^{-R} = (111) + (001) + (101) = (011) = f(m, e).$$

So the forged signature pair (x, s) for the message m will successfully pass all verification steps of the Alabbadi–Wicker scheme.

4 Conclusion

In this paper we presented an attack against the Alabbadi–Wicker scheme which is based on error–correcting codes. The described universal forgery mainly originates from the use of analogous matrices such as $\tilde{P}, \tilde{G}, \tilde{W}$ and Y which is different from the other attacks [1, 2, 3, 6, 7, 8]. Up to now all attempts to design a secure digital signature scheme based on error–correcting codes failed. Why are those schemes not secure? It is a long story. We shall give some reasons in [9].

References

[1] M. Alabbadi and S.B. Wicker, *Security of the Xinmei digital signature scheme*, Electronic Letters 28(9), 1992, pp.890–891.

[2] M. Alabbadi and S.B. Wicker, *Cryptoanalysis of the Harn and Wang modification of the Xinmei digital signature scheme*, Electronic Letters 28(18), 1992, pp.1756–1758.

[3] M. Alabbadi and S.B. Wicker, *Susceptibility of digital signature scheme based on error–correcting codes to universal forgery*, Proc. of 1994 IEEE International Symposium on Information Theory, Trondheim, Norway, 1994, pp.494.

[4] M. Alabbadi and S.B. Wicker, *A digital signature scheme based on linear error–correcting block codes*, Advance in Cryptology, ASIACRYPT'94, pp.238–248.

[5] E.R. Berlekamp, R.J. McEliece, and H.C.A. van Tilborg, *On the inherent intractability of certain coding problems*, IEEE Transactions on Information Theory, Vol.24, No.3, 1978, pp.384–386.

[6] L. Harn and D.C. Wang, *Cryptoanalysis and modification of digital signature scheme based on error–correcting codes*, Electronic Letters 28(2), 1992, pp.157–159.

[7] J. van Tilburg, *Cryptanalysis of the Xinmei digital signature scheme*, Electronic Letters 28(20), 1992, ppp.1935–1936.

[8] J. van Tilburg, *Cryptanalysis of the Alabbadi–Wicker digital signature scheme*, Proc. of Fourteenth Symposium on Information Theory in the Benelux, Veldhoven, Netherlands, May 1994, pp.114–119.

[9] Sheng–bo Xu, Jeroen Doumen and Henk van Tilborg, *On the security of digital signatures based on error–correcting codes*, to appear.

DIFFUSION ANALYSIS OF FEISTEL NETWORKS

J. Nakahara Jr, J. Vandewalle, B. Preneel

Katholieke Universiteit Leuven, div. E.S.A.T.–SISTA/COSIC
K.Mercierlaan 94, B–3001 Heverlee, Belgium
{jorge.nakahara,joos.vandewalle,bart.preneel}@esat.kuleuven.ac.be

Since 1977, when the DES algorithm was unveiled and research towards its cryptanalysis started, many different block cipher designs emerged, closely based on the Feistel structure of the DES, like LOKI, FEAL, KHUFU and KHAFRE.

This article addresses some guidelines for the design of Feistel Networks aimed at fast diffusion of the plaintext bits throughout the cipher, by using an appropriate combination of the swapping of sub-blocks in a round and an F-function mix of sub-blocks.

Key-words: Feistel Network, diffusion, swapping scheme, complete F-function

INTRODUCTION

A Feistel Network (FN) is a method for mixing the sub-blocks of an input block in a cipher through repeated application of keyed, non-linear F-functions in order to generate a permutation of the input block [4, 8, 9]. In a standard FN the plaintext block is divided evenly into two sub-blocks. The right sub-block is fed into a key-dependent F-function. The left sub-block is combined, via bitwise-exclusive-or (XOR), with the output of the F-function. Afterwards, the two sub-blocks are swapped (fig. 1(a)). A direct extension of the standard FN may split the input block into $n > 2$ sub-blocks [2]. It will be assumed that all sub-blocks have the same width so that each sub-block can be swapped with any other sub-block.

A general swapping scheme (SS) is a permutation of the $n \geq 2$ sub-blocks in a round. The identity swapping scheme, denoted Id, is the identity permutation. The inverse swapping scheme X^{-1} of an SS X is a permutation of the sub-blocks such that $X.X^{-1} = X^{-1}.X = Id$, where . denotes composition.

Swapping schemes, together with the F-functions, play an important role in the diffusion process in block ciphers [6]. The diffusion process of a function refers to the effect of spreading the influence of an input bit into many output bits.

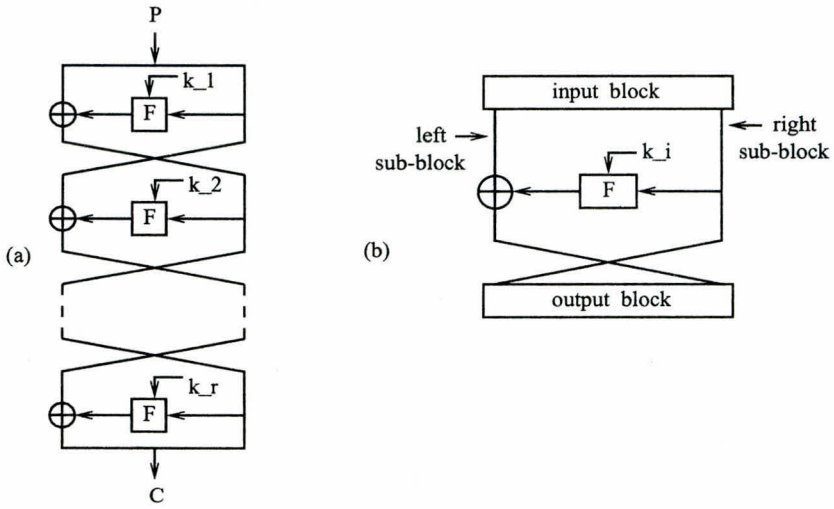


Figure 1: (a) Standard Feistel Network of DES cipher [7], (b) example of a round

A function (cipher) is called **complete** if each output bit depends on all of its input bits [3]. In the following analysis all F-functions will be assumed complete.

An F-function Mix denote a set of operations which combines sub-blocks in a round with (one or more) F-functions.

In a traditional FN the F-function is the most (computationally) expensive operation in a round and also plays a key role in the diffusion process due to its completeness property. Therefore, the effort to reach complete diffusion will be measured in terms of the amount of F-function calculations required.

An F-function in a block cipher usually has input and output of the same width as one sub-block. More generally, an (i, o) F-function is a function which accepts an input i sub-blocks wide and generates an output o sub-blocks wide. The DES cipher uses a $(1, 1)$ F-function [7].

Many SSs devised for block ciphers are symmetric i.e. they are equal to their inverses. In contrast, asymmetric schemes use different permutations than their inverse. Symmetric SSs have the advantage that the resulting FN is self-inversible i.e. it can be used both for encryption and decryption (just the order of the sub-keys might be inverted), thus simplifying design and saving space and computational cost for both hardware and software implementations. Accordingly,

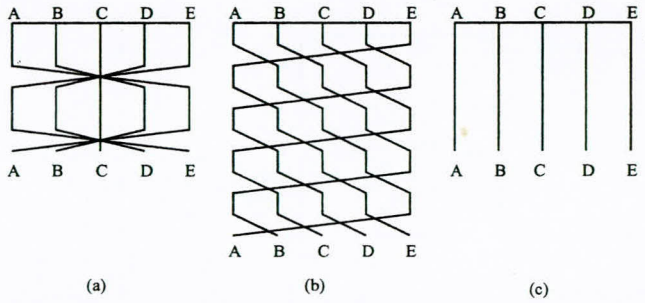


Figure 2: SSs: (a) symmetric, 3 clusters $\{A, E\}, \{B, D\}, \{C\}$, period 2; (b) asymmetric, 1 cluster $\{A, B, C, D, E\}$, period 5; (c) symmetric, 5 clusters $\{A\}, \{B\}, \{C\}, \{D\}, \{E\}$, period 1

the associated FN is also called symmetric.

It will be assumed that the same SS is used throughout the entire cipher.

Since SSs are permutations of sub-blocks and are applied repeatedly between rounds, it follows that after a certain number of rounds, all plaintext sub-blocks might return to their original relative position in a block. The minimum number of rounds needed for all input sub-blocks in an FN to return to their initial position is called the period of the SS. (fig. 2).

SSs involving n sub-blocks which have period different from n contain clusters of sub-blocks. A cluster is a subset of sub-blocks in an SS in which all sub-blocks in the cluster exchange place only with sub-blocks in the cluster (fig. 2(a) and (c)). For a block cipher which splits the input block into n sub-blocks, there are $n!$ permutations to choose from, i.e. $n!$ possible SSs. These SSs can be primarily divided into one-cluster (fig. 2(b)) and many-clusters (fig.2(a) and (c)), depending on the number of clusters. One-cluster SSs with n sub- blocks have period n. Many-clusters SSs have period $lcm(c_1, c_2, \ldots, c_j)$, where c_j is the period of the j-th cluster and $lcm(.)$ is the least common multiple function. Each cluster acts like a separate, independent permutation in an FN.

It is worth noting that if an SS is symmetric then it is also many-clusters. If an SS is one-cluster then it is asymmetric.

DIRECTED GRAPH REPRESENTATION

A directed graph of an FN is a pictorial representation in which sub-blocks (labeled by round number) denote vertices, and F-functions (labeled by sub-keys)

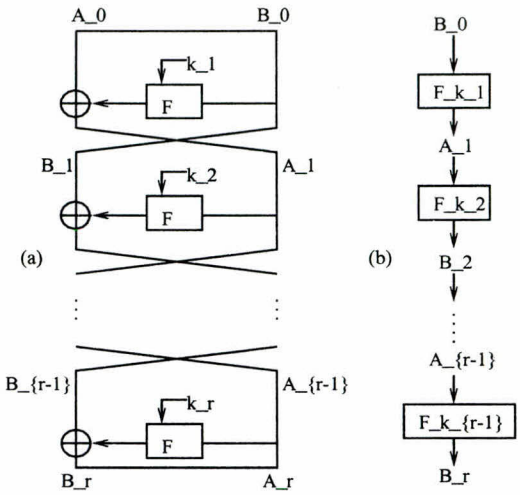


Figure 3: (a) FN and (b) its directed graph representation

denote edges (fig. 3). The graph is usually depicted top down. The initial (top) vertices comprise (some of) the input sub-blocks to the F-functions in the first round, and succeeding vertices are connected to the outputs of the F-functions in the following rounds. In order to focus attention on the diffusion process, not all the sub-blocks are necessarily depicted, but only the ones that participate directly in the input or output chain of F-functions. Only acyclic graphs are of interest.

ENCRYPTION AND DECRYPTION FNS

Depending on the FN, the diffusion process may not be equally efficient for the encryption and decryption operations. For example, in fig.4(a) the dependency of (A_3, B_3, C_3) on (A_0, B_0, C_0) can be summarized by the equations:

- $A_3 = A_0 \oplus F_{k_1}(B_0 \oplus F_{k_2}(C_0)) \oplus F_{k_6}(B_3)$
- $B_3 = B_0 \oplus F_{k_2}(C_0) \oplus F_{k_3}(C_0 \oplus F_{k_4}(A_0 \oplus F_{k_1}(B_0 \oplus F_{k_2}(C_0))))$
- $C_3 = C_0 \oplus F_{k_4}(A_0 \oplus F_{k_1}(B_0 \oplus F_{k_2}(C_0))) \oplus F_{k_5}(A_3)$

From these equations or the graph in fig.4(a), (at least) five F-function evaluations are needed to achieve complete diffusion, because A_2 does not yet depend

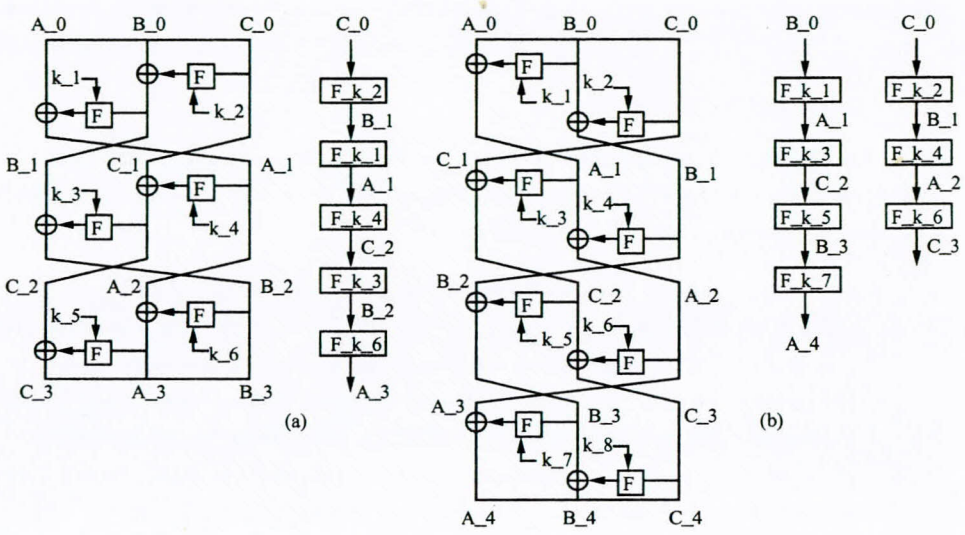


Figure 4: Diffusion performance in an FN: (a) encryption, and (b) decryption

on all input sub-blocks, but in fig.4(b), it takes (at least) seven F-functions, although one FN is the inverse of the other. The reason for this discrepancy is the chained dependency of $F_{k_{2i-1}}$ input on $F_{k_{2i}}$ output ($1 \leq i \leq 3$), in fig.4(a), that speeds up diffusion. This dependency can be noticed from the graph in fig.4(a) which is connected, i.e. there is only one graph component, while in fig.4(b) it is disconnected, resulting in a delayed diffusion.

FORMAT OF F-FUNCTION MIX

Traditional FNs use only one F-function per round. Assuming the cost of evaluating one complete F-function is non-negligible, it is a key decision to determine how many F-functions to use in a mix (i.e. per round), for efficient diffusion. Certain F-function mixes use more than one F-function instance, applied to different sub-blocks in order to execute them in parallel. These multiple-F-function mixes not only requires the necessary resources to take advantage of parallelism, but they must also achieve complete diffusion in less rounds than (simpler) F-function mixes in order to justify the larger computational cost per round. The idea of applying multiple F-functions to independent sub-blocks in order to compute them in parallel also contradicts the idea of chaining F-function. As a simplifying as-

sumption, the present analysis will restrict the mix to contain one F-function only.

CHAIN SEQUENCE

An appropriate integration between the F-function mix and the SS does effectively improve the diffusion process. Separately they can not reach complete diffusion properly. This effect is achieved by:

- an optimal balance between the input and output widths for an F-function, to improve diffusion in an FN with n sub-blocks, is a $(\lceil \frac{n}{2} \rceil, \lfloor \frac{n}{2} \rfloor)$ F-function, i.e. half of the sub-blocks are input and the other half are affected by the F-function output. For odd n another optimal balance use a $(\lfloor \frac{n}{2} \rfloor, \lceil \frac{n}{2} \rceil)$ F-function. If the available (i, o) F-function input is not large enough $(i < \lceil \frac{n}{2} \rceil)$, then $\lceil \frac{n}{2} \rceil$ of the sub-blocks can be combined via XOR into i sub-blocks. If the F-function output is not large enough $(o < \lfloor \frac{n}{2} \rfloor)$ it can still be combined with $\lfloor \frac{n}{2} \rfloor$ of the sub-blocks as long as different operators are used (the use of different, non-mutually-inverse operators is necessary, otherwise the XOR combination of some intermediate sub-blocks, which contain the same F-function value could result in an invariant value which propagates through many rounds, and is independent of the F-function output).
- the sub-blocks which are combined with the F-function output in one round might be input to the F-function in the next (immediate) round. This rule, combined with the previous one, guarantees a (balanced) chain of dependence of half of all sub-blocks affecting exactly the other half, in each successive round. The SSs also have to fulfill some properties. One-cluster SSs are preferred. Many-clusters SSs, especially clusters with one sub-block only, might be avoided because, since the F-function mix is usually the same for all of the FN, the mix will not properly diffuse sub-block dependency with one-sub-block clusters. If these conditions are met, complete diffusion can be expected in 3 rounds for even n and 4 rounds for odd n. These results follow from : if an (i, o) F-function is used per round then it takes at least $\frac{n}{i}$ rounds till all n sub-blocks are input to the F-function. Similarly, it takes at least $\frac{n}{o}$ rounds till all sub-blocks are combined to the F-function outputs carrying the chained dependency of all n sub-blocks. The minimum of $\frac{n}{i} + \frac{n}{o} = \frac{n}{i} + \frac{n}{n-i}$ occurs when $i = o = \frac{n}{2}$. If $i < \lfloor \frac{n}{2} \rfloor$ or $i > \lceil \frac{n}{2} \rceil$ then the diffusion process will be delayed, because, either too few sub-blocks

are been diffused through the F-function input, or too few sub-blocks are becoming complete by mixing with the F-function output.

AES CANDIDATES

It is to be observed that among the AES candidates [1] based on an FN with more than two sub-blocks, TWOFISH and RC6 [1] possess structures that allow the fastest diffusion if their F-functions were complete: TWOFISH after three F-function evaluations (rounds) and RC6 after four F-functions (two rounds). CAST-256 [1] would achieve complete diffusion after seven F-function instances or two rounds, if its f_1, f_2, f_3, f_4 functions were complete. MARS [1] would need five E-function evaluations (rounds) if its E-function was complete.

CONCLUSIONS

A summary of the design ideas that come up from the discussion on diffusion improvements for an FN are :

- equivalent diffusion performance might be enforced for both an FN and its inverse so that they can be used interchangeably.
- proper integration between the SS and the F-function mix is required. It is suggested enforcing a balanced chain sequences of F-functions, in which half of the sub-blocks are input to the F-function, and the other half of sub-blocks are combined to its output in each round. Sub-blocks which are mixed with the F-function output in one round might necessarily be input to the F-function in the next round.
- one-cluster SSs are preferred. Avoid SSs with fixed sub-blocks for they require more F-function evaluations per round, or variable F-function mixes to achieve complete diffusion.

One last notice. Although the given ideas predict that complete diffusion can be achieved in three or four rounds (with a complete F-function), there are many other criteria a cipher might satisfy in order to be considered minimaly secure, some of which oppose the use of so few rounds [5].

ACKNOWLEDGEMENT

Many thanks are warmly addressed to Vincent Rijmen for precious suggestions, many corrections and positive criticisms.

REFERENCES

[1] AES - Advanced Encryption Standard - http://csrc.nist.gov/encryption/aes/aes-home.htm

[2] A. Pfitzmann, R. Abmann - "Efficient Software Implementations of (Generalized) DES" - Proc. SECURICOM'90, Paris, 1990, p. 139-158

[3] A. J. Menezes, P. C. van Oorschot, S. A. Vanstone - "Handbook of Applied Cryptography" - CRC Press - 1997

[4] B. Schneier, J. Kelsey - "Unbalanced Feistel Networks and Block-Cipher Designs" - Fast Software Encryption, 3rd International Workshop Proceedings - Feb. 1996 - p. 121-144

[5] C. S. Jutla - "Generalized Birthday Attacks on Unbalanced Feistel Networks" - LNCS - vol. 1462 - Crypto'98 - p. 186-199

[6] C. E. Shannon - "Communication Theory of Secrecy Systems" - Bell Systems Technical Journal, vol. 27 - 1948 - p. 379-423, 623-656

[7] FIPS PUB 46 - "Data Encryption Standard" - Federal Information Processing Standards Publication 46, US Department of Commerce/National Bureau of Standards, National Technical Information Service, 1977 (revised as FIPS 46-1 : 1988, FIPS 46-2 : 1993)

[8] H. Feistel, W. A. Notz, J. L. Smith - "Some cryptographic techniques for machine-to-machine data communications" - Proceedings of the IEEE, 63 - 1975 - p. 1545-1554

[9] J. Kam, G. Davida - "Structured design of substitution- permutation encryption networks" - IEEE Transactions on Computers, 28 - 1979 - p. 747-753

[10] L. Brown, J. Pieprzyk, J. Seberry - "LOKI - a cryptographic primitive for authentication and secrecy applications" - AUSCRYPT 90 - LNCS 453 - p. 229-236

[11] R. C. Merkle - "Fast Software Encryption Functions" - CRYPTO 90 - LNCS 537 - p. 476-501

[12] S. Miyaguchi - "The FEAL Cipher Family" - CRYPTO 90 - LNCS 537 - p. 627-638

ANONYMITY CONTROLLED ELECTRONIC PAYMENT SYSTEMS

Joris Claessens*, Bart Preneel†, Joos Vandewalle

Katholieke Universiteit Leuven, ESAT–SISTA/COSIC
Kardinaal Mercierlaan 94, B–3001 Heverlee, Belgium
joris.claessens@esat.kuleuven.ac.be

Today's most used electronic payment systems offer no anonymity at all to their users. Electronic payment systems exist that provide unconditional anonymity. In this article electronic payment systems are described in which anonymity can be controlled. The requirements of such systems are discussed, and two types of tracing mechanisms are defined. An overview of the proposed anonymity controlled systems is given. We analyze to what extent these systems offer the defined tracing mechanisms. The importance of anonymous communication is stressed.

INTRODUCTION

The importance of electronic payment systems is increasing rapidly as electronic commerce is becoming part of our ordinary daily life. In order to make electronic payment systems acceptable to the general public, some degree of anonymity has to be offered. Just as with physical cash, the privacy of the users can be protected, so that the ability to link a buyer with a purchase is reduced. On the other hand, the anonymity of electronic money can be misused by criminals for money laundering, blackmailing and illegal purchases. In order to make anonymous electronic cash systems acceptable to governments and banks, mechanisms for revoking anonymity under certain conditions have been introduced. Such anonymity revocation must be possible only for a trusted third party, named trustee or judge.

In this article we discuss a number of electronic payment systems, ranging from today's frequently used systems which offer no anonymity, to systems which offer unconditional anonymity. In between these two are the systems which offer

* Funded by a research grant of the Flemish Institute for the Promotion of Industrial Scientific and Technological Research (IWT).

† F.W.O. postdoctoral researcher, sponsored by the Fund for Scientific Research – Flanders

some kind of controlled anonymity. We will discuss the different tracing models which can be present in an anonymity controlled electronic payment system, as well as the currently proposed systems that provide anonymity control according to these tracing models. Finally, we will address the need of anonymous communication in all these systems. Moreover, methods to obtain anonymous communication can be integrated with electronic payment systems that are far less complex, while still providing anonymity control properties.

NO PRIVACY VERSUS UNCONDITIONAL ANONYMITY

Today's electronic commerce is almost totally based on electronic credit card transactions on the World Wide Web (WWW). The security of this system relies on the security provided by the protocols used to secure the WWW [9]. The standard SET [16] provides a lot of advanced security features, but is not yet commonly used on the Internet. These systems do not protect the user's privacy, as the bank knows every purchase the user makes, via the user's credit card number. However, anonymity is probably not wanted by banks, as the value of credit card payments can be rather high.

The Belgian electronic wallet Proton [19] is an electronic payment system designed for small purchases. It is based on symmetric key cryptography and uses smart card technology. The system can already be used on the Internet, and is gaining international interest. The user's privacy is not protected. This should however be a necessary condition in order to fully emulate a wallet with physical coins.

Ecash [22] is an electronic cash system developed by (the late) DigiCash. The system uses *blind signatures* [8]. A blind signature scheme allows the user to get a message signed by a signer, without disclosing the contents of the message to the signer. In the ecash system coins are basically random strings signed by the bank. As the bank signs a blinded version of the random string, the bank cannot link the identity of the user (known during withdrawal of the coin) to the finally obtained coin (and thus to the payment in which that coin will be involved). To prevent double-spending of coins, the bank just keeps a list with all spent coins, and verifies whether a coin is already on that list. Therefore payment and deposit is done at the same time, which classifies the payment system as an on-line system.

More advanced electronic cash systems that offer unconditional anonymity, use *restrictive blind signatures* (e.g., the CAFE payment system is based on [2]).

The bank still signs a blinded string, but the identity of the user is encoded in this string. The encoding is done in this way, that the identity of the user is revealed when the coin is double-spent. The payment can be performed off-line (i.e., deposit can be done afterwards). More information about off-line electronic cash systems can be found in [20].

CONTROLLED ANONYMITY

Systems exist that provide unconditional anonymity and a mechanism to protect against double-spending. However, other crimes are possible [26], in which coins are not double-spent, such as money laundering and blackmailing. In order for an electronic payment system to be acceptable for both users, banks and government, some kind of anonymity control has to be provided. In these kind of systems, it is possible to trace the owner of a coin or to trace the coins originating from a specific withdrawal. This will only be possible in cooperation with a trusted third party (or more than one) which has additional information that can be used to link coins and users. The trusted third party (also called 'trustee') will only reveal information if e.g. the bank obtained an approval by a judge.

According to [10], generally the following *requirements* are necessary for anonymity control of electronic money:

1. Electronic money should be anonymous and unlinkable to legitimate users.
2. Anonymity should be revocable, but only by a trustee or judge and when necessary.
3. The trustee should not have any power other than tracing.
4. The bank should not be able to impersonate users (even in collaboration with the trustee or the shop).
5. The anonymity is revocable only for the transaction for which a trustee's order is given.
6. A trustee must be involved only when revocation is required, remaining off-line otherwise.
7. Anonymity revocation should not motivate crimes more serious than those it protects against.

Two types of anonymity control models are also discussed in [10], more specifically two types of *tracing mechanisms*: An *owner tracing* protocol exposes the identity of the owner of a specific coin. In this kind of protocol the bank gives to the trustee the information it received during the deposit protocol. The trustee

then returns information which can be used to identify the owners. Owner tracing prevents money laundering and making illegal purchases, as it allows the authorities to identify the involved users. This is also called 'after-the-purchase' tracing. A *coin tracing* protocol traces the coin that originated from a specific withdrawal. In this kind of protocol the bank gives to the trustee the information about a specific withdrawal. The trustee returns information that will appear when the coin is spent. Coin tracing allows the authorities to find the destination of suspicious withdrawals. It can thus be used to identify the seller of illegal goods, and prevents the blackmailing problem. This is also called 'before-the-purchase' tracing. Note that an anonymity controlled system must include both means of tracing in order to satisfy selectivity (requirement 5).

ANONYMITY CONTROLLED PAYMENT SCHEMES

This section describes the different types of schemes that implement anonymity control.

An efficient fair payment system. With a fair payment system, anonymous payments are still possible, but the anonymity can be removed with the help of a trusted party, which need not be involved in the transaction itself. The efficient fair payment system by Camenisch, Piveteau and Stadler [6] is an online system (payment and deposit at the same time) in which users have two types of accounts, personal and anonymous. An anonymous payment is a transaction from an anonymous account to a shop's account. The clue of the system consists of an efficient method for transferring money from a personal account to an anonymous account without revealing the correspondence between them (except to the judge).

The system is realized using electronic coins that can be paid only into a single account (therefore double-spending of coins can be prevented by a single counter, instead of a large database with spent coins). Perfect unlinkability of personal and anonymous accounts is realized by using a blind signature scheme. To achieve fairness, the judge knows the correspondence between the two accounts, and coins withdrawn from a personal account can only be deposit into the corresponding anonymous account.

Although the customer's identity is not revealed, the bank can still link different transactions when the same anonymous account is used for different payments. However, the customer can use different accounts for transactions that should not

be linked. This system provides owner tracing as the judge can at any time find the origin of a transfer given the anonymous account number, and coin tracing as the judge can find the destination of a transfer, given the personal account number.

Trustee-based tracing extensions. Brickell, Gemmell and Kravitz [3] add anonymity control as an extension to existing electronic cash systems – in particular [2] and [12] – but still provably protect user anonymity. It is interesting that two trustees are involved in these schemes. Only if the two agree and work together, owner/coin tracing can be performed.

These two proposals require the trustees to be on-line during withdrawal, which is not as efficient as stated in requirement 6 above.

Fair blind signatures. Stadler, Piveteau and Camenisch [5] proposed *fair blind signatures*, a new type of blind signature schemes, with the additional property that, with the help of a trusted entity, it is possible to link a message-signature pair and the corresponding protocol view of the signer.

Two different types can be defined. Type I fair blind signatures allow a judge to find the message-signature pair, given the signer's view of the protocol (which is in fact coin tracing). Type II fair blind signatures allow a judge to identify the sender, given a specific message-signature pair (i.e. owner tracing)

In anonymity controlled payment systems based on this scheme, the trustees are again on-line. Note that [25] points out some weaknesses and vulnerabilities in one of the fair blind signature schemes proposed in [5].

Advanced schemes. In the literature, more advanced schemes have appeared. In [13] an on-line electronic cash was proposed using *blind weak signatures*. A stronger tracing model, which survives the 'bank robbery' attack, was proposed in [15]. Owner and coin tracing with off-line trustees were proposed in [4] and [11]. A scheme with trustees distributed over the Internet can be found in [18].

ANONYMOUS COMMUNICATION

Although this is not always specifically mentioned in the papers, anonymous communication between the different parties is necessary, in order to have real anonymous cash. A solution for anonymous communication, called 'mixes', was already introduced in [7]. Mixes are applied in [24] for anonymous communication on the WWW. A very simple solution, but with not so strong properties, is the

Anonymizer [1]. Another interesting solution, with stronger properties, can be found in [21].

One can also specifically rely on anonymous communication during the design of an anonymous cash protocol. In this way, much simpler systems can be obtained. For example, in [23] anonymity is provided by just using one-way functions. In the following sections, two anonymity controlled payment systems are described, which specifically rely on anonymous communication or techniques used for this purpose.

Mix-based payments. In [14], Jakobsson and M'Raïhi present an electronic payment system in which anonymity and anonymity control are solely based on the use of a mix network. A payment order is of the form $o = m|a|s$ in which m is a hash of the contract, a account number merchant, and s serial number. The payer encrypts a set of orders $o_1 \ldots o_n$ to $\tilde{o}_1 \ldots \tilde{o}_n$ using the public key. The corresponding secret key is known only to the transaction center, and is kept in a distributed manner so that any quorum of mix servers can calculate it. The values of the transactions are described in d. The payer signs $\tilde{o}_1 \ldots \tilde{o}_n$ and d, resulting in σ. The transaction center verifies σ. It adds each valid $\tilde{o}_i$ to an internal list (one such list for each denomination) and debits the payer's account. It signs each $\tilde{o}_i$ (different public keys for different denominations) and sends $\sigma_1 \ldots \sigma_n$ to the payer. The payer sends $\tilde{o}_i, \sigma_i$ to the merchant, which verifies σ_i. At given intervals, the transaction center decrypts all payment orders: the input is a list of encrypted messages, the output is a permuted list of the corresponding decrypted messages. The items of these lists indicate what accounts are to be credited, and by how much. The two tracing operations are based on the mix en/de-cryption scheme. A payment order of a specific payer can be traced (coin tracing) by simply decrypting the encrypted payment order $\tilde{o}$ in question. The payer of a specific payment order can be traced (owner tracing) by reversing the encryption of o to $\tilde{o}$ step by step (how this must be done, depends on what type of mix network is used).

NetCash. NetCash [17] is an on-line electronic cash system developed at the University of Southern California. There are three parties involved: 'Clients', 'Merchants' (including Banks) and 'Currency Servers' (CS). Currency Servers have to obtain a certificate for minting currency from the 'Federal Insurance Corporation' (FIC): $\{Certif_id, CS_name, K_{CS}, issue_date, exp_date\}K_{FIC}^{-1}$. An electronic coin is represented as: $\{CS_name, s_addr, exp_date, serial_num, coin_val\}K_{CS}^{-1}, Certif_id$.

Both Clients and Merchants can perform an exchange protocol with a CS in order to verify coins, exchange coins for untraceability, purchase coins and cash in coins. A simple payer-payee exchange exists in which the Client receives a kind of ticket in exchange for coins, that can be used to get the service for which the Client has paid. Double-spending is either detected at the time the recipient verifies or exchanges coins with a CS, or the coins can only be spent by the recipient during an initial time window, allowing the recipient to cash them in before they can be double-spent. The anonymity provided by NetCash is weaker than the unconditional anonymity. It is possible for a CS to record which coins have been issued. It is expected that CSs will not do so. Clients can choose their own CS, and will choose one that they feel they can trust. If CSs can be seen as trustees, recording all information, but revealing information only when asked for by a judge, then the NetCash system can be classified as an anonymity controlled electronic cash system.

CONCLUSION

Today's currently used electronic payment systems do not provide privacy to the users. However, payment schemes exist that provide unconditional anonymity. These electronic payment systems are not used on a large scale, as they are not accepted by governments and banks.

In this article, the current state of the art of anonymity control in electronic payment systems has been discussed. We have seen the requirements for a good anonymity controlled payment system, as well as the two tracing models that have to be implemented in order to fulfill one of these requirements. We have discussed different proposed anonymity controlled electronic payment systems.

Finally, we have addressed the need for anonymous communication. Moreover, anonymity control in electronic payment systems can be implemented relying on the anonymous communication mechanisms.

REFERENCES

[1] Anonymizer, Inc. http://www.anonymizer.com.

[2] S. A. Brands. Untraceable Off-line Cash in Wallets with Observers. In *Crypto'93*, Springer-Verlag, LNCS 773, pp.302–318, 1994.

[3] E. Brickell, P. Gemmell, D. Kravitz. Trustee-based Tracing Extensions to Anonymous Cash and the Making of Anonymous Change. In *Proceedings of the 6th Annual Symposium on Discrete Algorithms (SODA'95)*, pp.457–466, 1995.

[4] J. Camenisch, U. Maurer, M. Stadler. Digital Payment Systems with Passive Anonymity-Revoking Trustees. In *ESORICS'96*, Springer-Verlag, LNCS 1146, 1996.
[5] J. Camenisch, J.-M. Piveteau, M. Stadler. Fair Blind Signatures. In *Eurocrypt'95*, Springer-Verlag, LNCS 921, pp.209–219, 1995.
[6] J. Camenisch, J.-M. Piveteau, M. Stadler. An Efficient Fair Payment System. In *ACM-CCS'96*, pp.88–94, 1996.
[7] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, v.24, n.2, pp.84-88, 1981.
[8] D. Chaum. Blind signatures for untraceable payments. In *Crypto'82*, Plenum Press N.Y., pp.199–203, 1983.
[9] J. Claessens, J. Uyttenhove. Beveiliging van het World Wide Web. Ms. Thesis (in Dutch), COSIC, K.U.Leuven, 1997.
[10] G. Davida, Y. Frankel, Y. Tsiounis, M. Yung. Anonymity Control in E-Cash Systems. In *Financial Cryptography '97*, Springer-Verlag, LNCS 1318, pp.1–16, 1997.
[11] Y. Frankel, Y. Tsiounis, M. Yung. Indirect Disclosure Proof: Achieving Efficient Fair Off-Line E-Cash. In *Asiacrypt'96*, Springer-Verlag, LNCS 1163, pp.286–300, 1996.
[12] M. Franklin, M. Yung. Towards Provably Secure Efficient Electronic Cash. Columbia Univ. Dept. of C.S. TR CUCS-018-92, 1992.
[13] M. Franklin, M. Yung. Blind weak signature and its applications: Putting non-cryptographic computation to work. In *Eurocrypt'94*, Springer-Verlag, LNCS, pp.71–83, 1994.
[14] M. Jakobsson, D. M'Raïhi. Mix-based Electronic Payments. *Fifth Annual Workshop on Selected Areas in Cryptography (SAC'98)*, Queen's University, Kingston, Ontario, Canada, August 1998.
[15] M. Jakobsson, M. Yung. Revokable and Versatile Electronic Money. In *ACM-CCS'96*, pp.76–87, 1996.
[16] MasterCard Int., Visa Int. SET Secure Electronic Transaction. http://www.setco.org.
[17] G. Medvinsky, B. C. Neuman. NetCash: A design for practical electronic currency on the Internet. In *ACM-CCS'93*, 1993.
[18] D. M'Raïhi, D. Pointcheval. Distributed Trustees and Revocability: a Framework for Internet Payment. In *Financial Cryptography '98*, Springer-Verlag, LNCS 1465, 1998.
[19] Proton World Int. http://www.protonworld.com.
[20] C. Radu. Analysis and design of off-line electronic payment systems. PhD. Thesis, COSIC, K.U.Leuven, 1997.
[21] M. K. Reiter, A. D. Rubin. Crowds: Anonymity for Web Transactions. DIMACS Technical Report 97-15, 1997.
[22] B. Schoenmakers. Basic Security of the ecash™ Payment System. In *Computer Security and Industrial Cryptography: State of the Art and Evolution*, ESAT-COSIC course, Leuven, Belgium, 1997. Springer-Verlag, LNCS 1528, 1998.
[23] D. R. Simon. Anonymous Communication and Anonymous Cash. In *Crypto'96*, Springer-Verlag, LNCS 1109, pp.61–73, 1996.
[24] P. F. Syverson, D. M. Goldschlag, M. G. Reed. Anonymous Connections and Onion Routing. In *1997 IEEE Symposium on Security and Privacy*, 1997.
[25] J. Traoré. Making Unfair a "Fair" Blind Signature Scheme. In *ICICS'97*, pp.386–397, 1997.
[26] S. von Solms, D. Naccache. On Blind Signatures and Perfect Crimes. *Computers & Security*, 11:581–583, 1992.

IMPROVED NOISE ROBUSTNESS FOR SPEECH RECOGNITION BY ADAPTIVE SVD-BASED FILTERING

K. Hermus, P. Wambacq and D. Van Compernolle*

Katholieke Universiteit Leuven, div. E.S.A.T.–PSI
K.Mercierlaan 94, B–3001 Heverlee, Belgium
Kris.Hermus@esat.kuleuven.ac.be

The presence of all kinds of noise in the speech signal can lead to a significant degradation in recognition performance; this is well-known as the noise robustness problem for speech recognition applications.

In this paper we propose a promising method to improve the robustness against the influence of additive noise at moderate SNR ratios.

The proposed technique – based on Singular Value Decomposition (SVD) – originates from classical signal enhancement, but it is adapted to the specific requirements imposed by the speech recognition process.

Noise reduction is obtained by suppressing low energetic, noise related singular components of the Hankel matrix which is constructed from the original signal. It is shown that the preservation of the formant structure of the speech sounds is of utmost importance.

Recognition experiments on a database with large vocabulary, continuous speech (Resource Management) reveal a drastic reduction in error rate, thereby outperforming existing techniques for noise robustness such as Nonlinear Spectral Estimation and SNR-Normalisation.

The SVD filtering can easily be included in the speech preprocessing scheme, and works in real time.

INTRODUCTION

Decades of research in speech recognition resulted in the development of recognisers able to transcribe fluently spoken language with a very high accuracy. Continuous speech with vocabularies of 64.000 words and more can be recognised with a WER (Word Error Rate) of less than 10%. Even better results (WER $< 5\%$) are obtained with speaker dependent systems, where the acoustic models are (re)trained by the user.

However, these excellent results could only be obtained in controlled environments

* Currently with Lernout & Hauspie Speech Products

– matched training and testing environments – without any disturbing source. The interference of e.g. noise, which is very likely to happen in real conditions, greatly compromises the recognition rates and often makes the systems completely useless.

Evidently, providing the recognisers with (noise) robust features was an indispensable step on the way to a successful commercialisation of speech recognition software. As this problem was only considered lately, no single, integrated solution was proposed. On the contrary, a countless collection of techniques [1] has been constructed, each with its own restrictions and applications.

Many of the techniques belong to one of the following classes:

Speech Enhancement These methods focus on improving the perceptual quality of the speech signal (increasing the SNR ratio). If possible, the noise itself is recorded or estimated, e.g. for *Adaptive Filtering* techniques. In case of *Spectral Estimation [3]* and its variants, the noise spectrum is estimated and then subtracted from the spectrum of the speech signal, thereby removing the noise.

Robust Features Recognisers can be made robust up to a certain level by working with (noise) robust parameters. The noise is not necessarily removed, as it is incorporated in the scheme. *SNR-Normalisation [3]* sends the signal through a filter bank, and artificially adds noise in every frequency bank such that the SNR of all filtered components is normalised towards a preset value. This results in a better recognition of noisy speech.

Model Adaptation If the influence of the noise on the acoustic models is known, one can adapt these models to the noise present in the input speech signal. Examples are *Stochastic Matching* and *Parallel Model Combination [1].*

Despite the intensive research and the large number of methods, fully robust recognisers still don't exist, certainly not for large vocabulary speech recognition. The large variability in noise conditions, and the need for simple implementations (single microphone recording, low computational cost, ...) make the development of one integrated solution very demanding. Consequently, almost every scheme puts restrictions on the allowed noise conditions.

In this paper, we present a new approach to robust speech recognition, based on SVD filtering. SVD has been applied in many signal processing fields: image coding (eigenfaces), signal enhancement (medical world), image filtering, ... Also for plain signal enhancement (improving the perceptual quality) SVD can be a solution where it comes to remove additive noise at reasonable SNR values.

However, a straightforward application of classical SVD signal enhancement to speech recognition problems leads to dramatic results. Indeed, speech recognition imposes

very specific constraints (such as preservation of the formants[1]) on the distortion introduced by the noise filtering. Indeed, the acoustic models are not (re)adapted to the SVD-enhanced signals.

We will show that, starting from the basic filtering scheme, a variety of solutions are at hand, leading to drastic recognition improvements for white and coloured additive noise.

The outline of the paper is as follows. First a short description of the speech preprocessing step is given. Thereafter, the basic theory of SVD filtering is presented, together with two practical implementations for white noise. The adaptation of the algorithms to coloured noise removal is also addressed. Finally, recognition results are given, followed by some conclusions.

SPEECH SIGNAL PROCESSING

The goal of the preprocessing step in a recogniser is to capture all the relevant acoustic information from the speech signal in a small set of uncorrelated acoustic parameters. In order to capture the *immediate* acoustic information, the signal is divided in (overlapping) frames: from each frame (typically 30 msec) a set of about 30 parameters is calculated.

A normal preprocessing scheme transforms the signal into the frequency domain (mostly cepstral domain), includes time derivatives of the parameters, applies some kind of both additive and convolutive noise removal, and tries to decorrelate the parameter set.

THEORY OF SVD FILTERING

Introduction

The noisy signal is treated as a vector in an T-dimensional space, with noise and speech components lying in orthogonal subspaces. From the noisy signal, a Hankel matrix is constructed, whose rank is reduced with SVD. The high energy components are supposed to contain only speech data, whereas the low energy components are supposed to contain only noise.

Constructing a Hankel matrix The noisy signal $y = y(0), y(1), \ldots$ consists of the desired signal x and the noise n; both components are considered being additive:

$$y(k) = x(k) + n(k) \tag{1}$$

[1]Every shape of our vocal tract (mouth, lips, ...) has its own set of resonant frequencies (formants), that produce a particular sound. Recognition of sounds is mainly based on the location of the different formants.

As in the normal preprocessing, we create overlapping frames of T samples: $y(0), y(1), \ldots, y(T-1)$. The number of samples T is to be optimised (see further).
From all frames, we construct a Hankel matrix Y:

$$Y = \begin{bmatrix} y(0) & y(1) & \ldots & y(M-1) \\ y(1) & y(2) & \ldots & y(M) \\ \vdots & \vdots & & \vdots \\ y(K-1) & y(K) & \ldots & y(T-1) \end{bmatrix} \tag{2}$$

with dimensions $K \times M$, with $K \geq M$ and $M + K = T + 1$.
According to the assumption of additive noise, we can write Y as:

$$Y = X + N \tag{3}$$

Altering the singular value spectrum

As we already mentioned, the largest singular components of Y are supposed to contain more speech whereas the smallest singular components include almost only noise. Hence, noise can be suppressed by altering the singular value spectrum, giving higher weights to the speech components and suppressing the noise components.

Several procedures to suppress the noise are possible, each with its own weighting scheme. Two of them are depicted here.

Least Squares estimation Speech is often modelled as a sum of p complex exponentials. In that case, X is of rank p, en we can obtain a *Least Squares/rank p* estimate of Y by setting the $M - p$ smallest eigenvalues to zero.
Calculation of the SVD of Y yields

$$Y = U\Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \tag{4}$$

Σ_1 contains the p largest singular values, Σ_2 the $M - p$ smallest ones. Due to the (broadband) noise, Y is mostly of full rank.
The best rank-p approximation of Y equals:

$$Y_{LS,p} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} = U_1 \Sigma_1 V_1^T \tag{5}$$

Minimum Variance estimation Another approach of SVD-filtering is the MV estimation, deduced for the white noise case, i.e.

$$N^T N = \sigma_{noise}^2 I \tag{6}$$

The mathematical formulation of this estimation, together with a geometrical interpretation, can be found in [2].
The weighting factor w_i for the i-th singular component with singular value σ_i equals

$$w_i = 1 - \frac{\sigma_{noise}^2}{\sigma_i^2} \tag{7}$$

where σ_{noise}^2 is estimated based on speechless frames. The MV estimation is given by

$$Y_{MV} = U\,(W\Sigma)\,V^T \;\;;\;\; W = diag\left(1 - \frac{\sigma_{noise}^2}{\sigma_1^2}, \ldots, 1 - \frac{\sigma_{noise}^2}{\sigma_M^2}\right) \tag{8}$$

with W a diagonal matrix containing the weights.

Restoring the Hankel structure Matrices $Y_{LS,p}$ and Y_{MV} are not Hankel anymore such that an easy extraction of the enhanced signal $\hat{x} = \hat{x}(0), \hat{x}(1), \ldots, \hat{x}(T-1)$ is impossible. However, the Hankel structure can be restored by constructing a new matrix $\overline{X}$ where every element from an anti-diagonal of $\{Y_{LS,p}, Y_{MV}\}$ is replaced by the average value along that anti-diagonal.

Evaluation of both methods

Equivalent FIR-filter representation The i-th singular component of the Hankel matrix Y can be extracted directly in the time domain by filtering $y(k)$ by a FIR-filter deduced from the i-th right singular vector of Y.

Hence, altering the singular spectrum of Y can be regarded as applying an M-th order filter bank on $y(k)$, and giving each filtered component its proper weight.

Least Squares By selecting some singular components, and suppressing the other components (binary approach)[2], some frequency bands may be filtered out completely. This destroys the formant structure of the speech signal, resulting in a corrupt parameter set and poor recognition performance.

[2]also called 'truncated SVD'

Minimum Variance For the MV estimation, every singular component remains present in the enhanced signal, such that no frequency dips can occur. This approach results in a much higher recognition improvement.

COLOURED NOISE REMOVAL

The noise reduction schemes described so far are only valid for speech corrupted by white noise with a flat spectrum. However, the algorithms can easily be adapted for the removal of coloured noise if an estimate of the noise is available.

Noise Reference

It is obvious that the more information we have about the noise, the better the noise removal can be performed. Since we only allow a simple one-microphone recording, knowledge about the noise should be extracted from the speech signal during noise-only periods.
Therefore, we need an *accurate* speech/noise classifier, since misclassification will lead to erroneous filtering. We assume stationary noise such that the shape of the noise remains constant during the next speech frames, until a new estimate can be made. Once the noise spectrum is known, it can be included in the SVD method by applying a 'prewhitening' step.

Prewhitening

For the general coloured noise case, $N^T N \neq I$. However, if N is available we can use the Cholesky factor from the QR-factorisation of N to prewhiten the noise.
To that end, we multiply the Hankel matrix Y with R^{-1} obtained from $N = QR$:

$$\hat{Y} = YR^{-1} = XR^{-1} + NR^{-1} \tag{9}$$

such that the noise factor NR^{-1} of $\hat{Y}$ complies with (6). Indeed,

$$\left(NR^{-1}\right)^T \left(NR^{-1}\right) = Q^T Q = I \tag{10}$$

The formulas derived for the white noise case can now be applied to $\hat{Y}$. Finally, a dewhitening step is performed by multiplying $\hat{Y}_{MV}$ or $\hat{Y}_{LS,p}$ with R.

Implementation by QSVD

Subsequent prewhitening and dewhitening can cause a loss of accuracy due to numerical instability. Working with the QSVD (Quotient SVD) of the pair (X, N) is an efficient way to avoid this problem [2].

RECOGNITION EXPERIMENTS

Experiments were performed with the ESAT-speech recogniser on the Resource Management database. This database consists of continuous speech of office quality, with a vocabulary of 1000 words. Tests were performed on the feb89 test set using a Word Pair grammar and Context-Independent acoustic models.

For a description of the ESAT-speech recogniser, the reader is referred to [4].

The SVD filtering comes before the normal preprocessing and has following specifications. We used a frame length of 30 msec (= 480 samples at 8 kHz sampling rate); dimension M of the Hankel matrix Y was set to 8.

To obtain noise corrupted data we created four databases from the clean RM-data (on which we get a reference WER of 4.88%) by adding additive white and coloured noise (low-pass filtered white noise, cut-off frequency = 4 kHz), at 10 and 15 dB SNR.

Least Squares Estimation This approach normally leads to the destruction of the formant structure. Although the perceptual quality of the speech signal improves, the extracted parameter set is almost worthless for the acoustic pattern matching step.

Experiments revealed the shortcomings of the Least Squares estimation. The relative reduction in WER never decreased by more than 15%. Therefore, we concentrated on the more powerful Minimum Variance estimation.

Minimum Variance Estimation The WER's for the MV estimation noise removal are summarised in table 1.

These results are compared with those obtained by two other well-known techniques: Nonlinear Spectral Estimation (NSE) and SNR-Normalisation (SNR-N.). For SNR-N., we normalised towards the optimal values of 12 and 18 dB for the 10 and 15 dB SNR database respectively.

From the table we learn that our new approach is far superior to existing techniques as Nonlinear Spectral Subtraction (NSE) and SNR-Normalisation. We get relative reductions of the WER of more than 65% for white noise and of more than 55% for coloured noise.

CONCLUSION

SVD filtering is a powerful tool for the enhancement of speech signals. When adapted to the requirements of speech recognition, it enables the development of noise robust systems for large vocabulary continuous speech recognition.

Noise reduction is obtained by altering the singular value spectrum of the Hankel

SNR ratio	Ref.	NSE	SNR-N.	MV
white noise				
10 dB	57.17	41.74	39.79	17.96
15 dB	25.77	21.36	21.71	8.79
coloured noise				
10 dB	56.19	34.21	39.24	23.55
15 dB	20.62	17.96	17.13	9.22

Table 1: *Word Error Rates for Minimum Variance SVD-noise removal. Ref: Reference (no noise reduction); NSE: Nonlinear Spectral Estimation; SNR-N.: SNR-Normalisation; MV: Minimum Variance.*

matrix. From the numerous possible implementations, we described the basic Least Squares solution, and the very powerful Minimum Variance estimation.

Due to the prewhitening of the noise, the technique remains valid for the general coloured noise case. If the noise doesn't change between subsequent speechless frames, non-stationary noise can be removed with the aid of an accurate noise reference, estimated from speechless frames.

Moreover, the computational load is rather low; the implementation of the MV approach in C works in real time.

ACKNOWLEDGEMENT

The support is acknowledged from the GOA Programme (K.U. Leuven Research Fund).

REFERENCES

[1] Y. Gong, "Speech Recognition in Noisy Environments: a Survey", Speech Communication, vol. 16, nr. 3, pp. 261–291, April, 1995.

[2] S.H. Jensen, P.C. Hansen, S.D. Hansen and J.A. Sørensen, "Reduction of Broad-Band Noise in Speech by Truncated QSVD", IEEE Transactions on Speech and Audio Processing, vol. 3, pp. 439–448, November, 1995.

[3] T. Claes, F. Xie and D. Van Compernolle, "Spectral Estimation and Normalisation for Robust Speech Recognition", Proc. International Conference on Spoken Language Processing, vol. IV, pp. 1997–2000, October, 1996.

[4] J. Duchateau, K. Demuynck and D. Van Compernolle, "Fast and Accurate Acoustic Modelling with Semi-Continuous HMMs", Speech Communication, vol. 24, nr. 1, pp. 5–17, April, 1998.

ON THE OBJECTIVE SPEECH QUALITY OF THE TETRA SYSTEM

C.H. Slump[1]

T. de Bont[2], A.M. Mertens[2], K. Verwey[2]

[1]University of Twente
Dept. Electrical Engineering
PO Box 217, 7500 AE Enschede
The Netherlands
C.H.Slump@el.utwente.nl

[2]National Police Agency
IT - Organization
PO Box 100, 3970 AC Driebergen
The Netherlands

We address the speech quality to be expected from the new TErrestrial Trunked RAdio (TETRA) digital mobile communication system for public order and safety. TETRA is the new digital transmission standard for private mobile radio from the European Telecommunication Standards Institute (ETSI). In TETRA an Algebraic Code-Excited Linear Prediction (ACELP) algorithm for speech compression and decompression is implemented. Of interest is the behavior of this ACELP algorithm together with the forward error protection in case the Bit-Error Rate (BER) is nonzero.

INTRODUCTION

The communication of professionals operating in the realm of public order and safety such as police, fire - brigade and ambulance services, is different from that which is possible with mobile telephones. There is group rather than one-on-one communication. Also, the user must be able to realise contact immediately, while in telephone communication the buildup of the connection takes a certain amount of time. A project has started to realise a countrywide network in The Netherlands, based upon the TETRA standard. TETRA stands for TErrestrial Trunked RAdio [1]. At this time it is the only pan - European standard which is able to meet all requirements of mobile users such as police, fire - brigade and ambulance services. In order to gain experience with the technical aspects of the TETRA standard, the IT organisation of the National Police Agency carries out a technical pilot. The pilot consists of a radio network (a switch and three base stations) and mobiles. The network is currently operational in the area around Driebergen, see Figure 1. The permanent transmitters/receivers are located in Linschoten, Maartensdijk and at the IT organisation site in Driebergen. The IT organisation is connected with Linschoten and Maartensdijk via rental lines of the telephone company. Communication is possible in the area in between by means of mobile phones. TETRA is

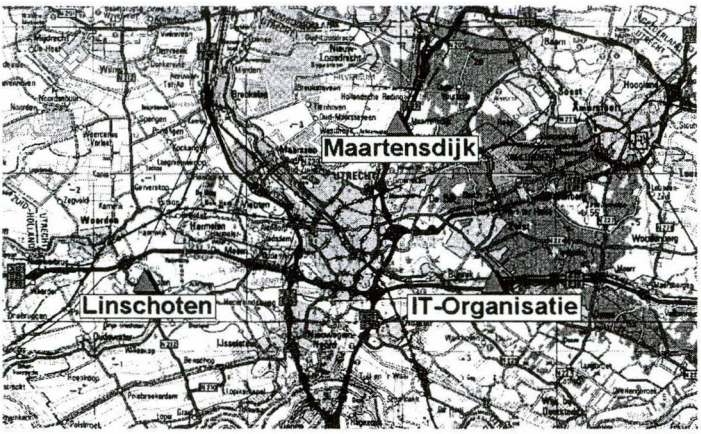


Figure 1: Map of the area.

a cellular system. The area where the cellular system operates (area coverage) is defined as an area where a **M**obile **S**tation (MS) is able to establish a connection with a **B**ase **S**tation (BS). The size of the cell is not defined in the TETRA standard. In [2, 3] we have studied the cell size for the network in the Netherlands, following a classical approach based upon fieldstrength and outage probability. As TETRA is mainly designed for group communication, capacity problems caused by the number of users is therefore not a major issue. Therefore, the size of the cells is limited by the quality and quantity of the received signal strength which is necessary to establish a connection. This strength is influenced by radio propagation losses, by fading, by signal interference of cells from other clusters and by noise. Those factors are heavily dependent on the area where the transmitter and receiver are situated. The radii of the different areas in The Netherlands are estimated in two steps. The first step is based upon a statistical approach from which the probability of outage is determined. The second step is to use the outage probability to calculate the actual field strength received. The radius is then determined for the different areas. The location and the number of base stations have been carefully chosen. The goal of our research is to objectively measure the decrease of speech quality as a function of BER. Based upon field strength measurements and pertinent BER values from the TETRA test system we assess the speech quality with a modified Perceptual Speech Quality Measure (PSQM) [4, 5].

BER MEASUREMENTS

In this section we describe only some of the BER masurements. If we look at the area where the (uplink) BER measurements are taken (Linschoten) we obtain the following cell sizes in Table 1, with different clutter factors β, indicating the extent of the urbanisation. Only the area between the MS and BS is taken into

Table 1: Approximation of cell-sizes for the Linschoten area.

Boundary coverage	β = -3dB	β = -8dB	β = -11dB
75%	16098 m	12072 m	10157 m
90%	11098 m	8929 m	7513 m
95%	9761 m	7320 m	6159 m
99%	6379 m	4784 m	4025 m

consideration, see Figure 2 for the route of the MS. In this case the TETRA MS consists of a 1W transmitter mounted on a BMW motorcycle with a Katrein 3.5 dBi antenna. The BS situated in Linschoten, has a three sector 12.5 dBi antenna at a heigth of 40 m. A characterization of the speed of the MS along the route



Figure 2: The route of the MS

is presented in Figure 3. The received field strength at the BS is characterized in Figure 4. The resulting BER is shown in Figure 5. The next section describes in short the speech source encoder and decoder.

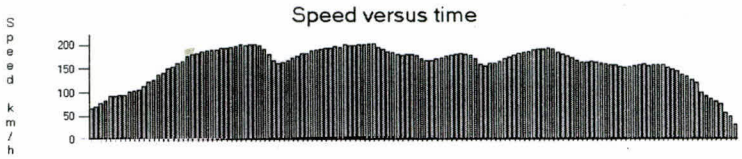


Figure 3: Speed of the MS.

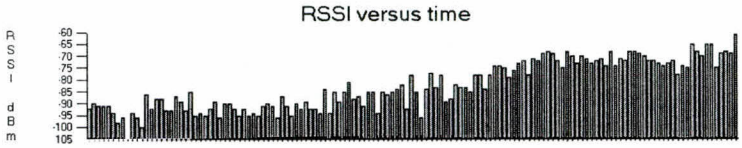


Figure 4: Received Signal Strength Indicator at the BS as function of time.

SPEECH CODEC

The TETRA speech codec belongs to class of vocoders and is based on the Code Excited Linear Prediction speech model. Due to limited space, we can only describe here some of the main issues. For details do consult [6]. A speech segment of N samples is generated by filtering a selected excitation sequence from a codebook with two time - varying filters and a scaling gain factor g_c. A simplified high level block diagram of the speech synthesis is shown in Figure 6. The pitch filter (long - term prediction) models the time - varying periodicity in the speech signal and the short - term filter shapes the spectral envelope. The pitch filter is given by:

$$H(z) = \frac{1}{B(z)} = \frac{1}{1 - g_p z^{-T}}, \tag{1}$$

where T is the pitch delay and g_p is the pertinent gain. The pitch synthesis filter is implemented as an adaptive codebook. For delays less than the length of a subframe of $N/4$ samples, the past excitation is repeated. The short - term synthesis filter is given by:

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 + \sum_{i=1}^{p} a_i z^{-i}}, \tag{2}$$

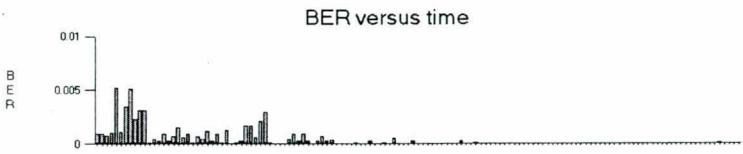


Figure 5: BER at the BS as a function of time.

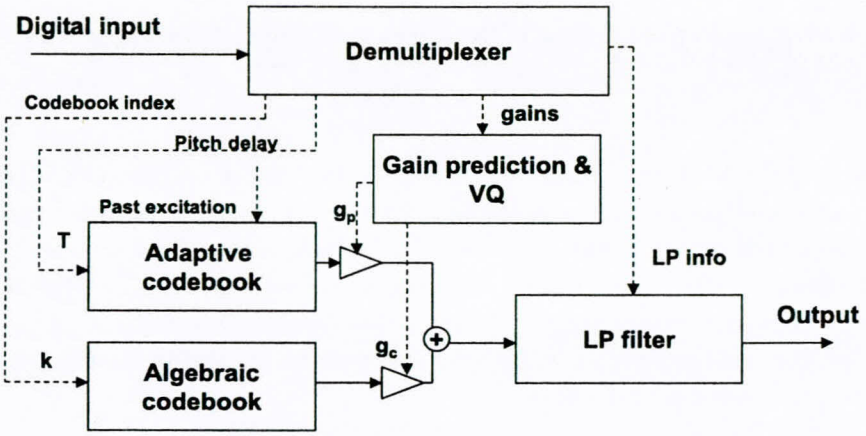


Figure 6: Block diagram TETRA decoder.

with $a_i, i = 1, \ldots, p$, the well - known Linear Prediction (LP) parameters of order p. Also in TETRA p is chosen to be equal to 10. The TETRA encoder uses the well - known analysis - by - synthesis technique to determine the pitch and excitation codebook parameters. A simplified block diagram of the speech encoder is shown in Figure 7. In the analysis - by - synthesis technique, the synthetic speech is computed

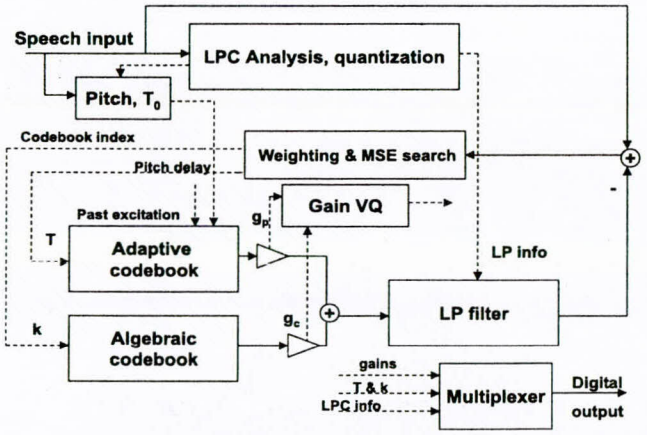


Figure 7: Block diagram TETRA encoder.

for all candidate excitation sequences. The pertinent innovation sequence, that produces the speech closest to the original speech frame according to a perceptually weighted distortion measure, is selected. This perceptual weighting filter attenuates

the error at the formant regions of the speech spectrum and is given by:

$$W(z) = \frac{A(z)}{A(z/0.85)}, \tag{3}$$

with $A(z)$ the LP inverse filter of Equation 2. In Algebraic CELP (ACELP) the innovation codebooks have an algebraic structure with advantages with respect to storage, search strategy and robustness. The TETRA codebook has fixed excitation vectors that are shaped by a dynamic shaping matrix which depends on $A(z)$, and its role is to concentrate the energy of the excitation vectors in the important frequency bands. The shaping matrix is a TOEplitz lower triangular matrix constructed from the impulse response of the filter:

$$F(z) = \frac{A(z/0.75)}{A(z/0.85)}. \tag{4}$$

In the codec the speech frames consist of 240 samples, with a sampling frequency of 8 kHz this is equal to are 30 ms. The LP parameters are computed and transmitted for each consecutive frame. The frame is divided in four subframes of 60 samples (7.5 ms) and the pitch and codebook parameters are obtained and transmitted per subframe. Table presents the bit allocation for the TETRA codec. Each frame is represented by 137 bits, resulting in a bit rate of 4567 bit / s.

Parameter	1st subfr.	2nd subfr.	3rd subfr.	4th subfr.	Total
LP filter					26
Pitch delay	8	5	5	5	23
Algebraic code	16	16	16	16	64
Gains (VQ)	6	6	6	6	24
Total per frame					137

Table 2: Bit allocation.

CHANNEL CODING FOR THE TETRA SPEECH CHANNEL

The structure of the channel coding is presented in figure 8. Two speech frames correspond to one transmission time slot and are encoded together. The size of the input block is therefore equal to 274 bits. The 137 bits representing the speech parameters of one speech frame are ordered in three sensitivity classes. Each sensitivity class is protected in a different way. Cyclic Redundancy Check (CRC) parity bits are companded only to the most sensitive parameters, such as the codebook

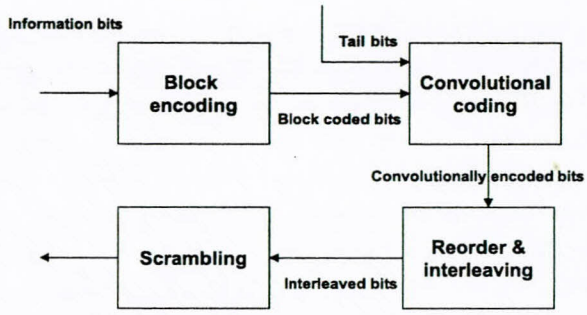


Figure 8: Flow diagram channel coding.

filter index and the gains. Per frame 30 bits are categorized in the most sensitive class. The CRC code is applied to these bits with the polynomial generator:

$$G(X) = 1 + X^3 + X^7, \tag{5}$$

generating 7 parity bits. The modulo 2 sum of the bits in the most sensitive class together with the 7 parity bits is added as an overall parity bit $p = 8$. With the addition of 4 tail bits, the size of the data block is now two times 51 bits at the lowest sensitivity level plus two times 56 bits at the intermediate level plus two times 30 bits at the highest level plus 8 parity bits, together with the 4 tail bits this adds up to 286 bits. The two most sensitive classes are convolutionally encoded. The 8 parity bits and the 4 tail bits are included in the highest sensitivity class. The bits of the least sensitive class remain unprotected, a convolutional 2/3 code is applied to the intermediate bits and the most sensitive bits are protected by a convolutional 8/18 code. This affects the frame size as follows: two times 51 unprotected bits plus two times 56 times 3/2 plus two times 42 times 18/8 equals 432 bits added to the block. The 432 bits are interleaved to increase the robustness of the code. The 432 bits can be represented in a matrix of 24 rows and 18 columns. The reordered bits are obtained by transposing the matrix. After this the scrambling for security takes place. In the decoding the inverse trajectory is followed. The CRC will be computed for the highest sensitivity bits and compared with the extra bits. In case differences occur, the Bad Frame Indicator (BFI) flag will be set and compensating measures are taken.

OUTLOOK

The aim of this research is to supplement the design and layout of the cellular TETRA network with information about the speech quality to be expected. The radii of the cells have high boundary coverage specifications because they will be used

for public order and safety. In case of a disaster there must be a guarantee of good communication among and within public service groups (police, fire-brigades and ambulances). The BER measurement shows the influence of the field strength. The speed of the MS appears not to have so much impact on the Bit Error Rate, this was to be expected for the pertinent environment with near line-of-sight communication. The PSQM measures shows a small decrease of quality in the area where the BER is maximaly 0.5 %. In this case only unprotected bits in the speech frame are affected. In the presentation we will show more details about the behavior of the voice coder in combination with the channel coding and present more measurements in urban areas at lower speed with typical public order ansd safety type of speech samples.

REFERENCES

[1] ETSI, TETRA Voice + Data - Designers' guide, ETR 0.1.0, European Telecommunication Standards Institute, F-06921 Sophia Antipolis - France, 1996.

[2] J. van Bussel, J.J.H. Tijdhof, H. Schurer, C.H. Slump, TETRA project - Optimum Cell Size for Public Order and Safety, University of Twente, EL-BSC-020N97, 1997.

[3] J. van Bussel, Schurer, H. Slump, C.H. and Rossou, E.K.L.(1997, September 30). Optimum Cell Size for Police Use. [Proceedings of the Second European Personal Mobile Communications Conference together with ITG-Fachtagung 'Mobile Kommunikation'].pp. (157-163). Bonn, Germany.

[4] ITU-T, Contribution COM 12-67, Draft recommendation P.861, Objective quality measurement of telephone-band (300-3400 Hz) speech codecs, study period 1993-1996.

[5] C.H. Slump. and Christoffers, J.W.(1998, March 26). A Perceptual Speech Quality Measure in the Design of a 2.4 kb/s speech codec. [Proceedings of Signal Processing Symposium SPS 98].pp. (183-186). Leuven, Belgium.

[6] ETSI, Radio Equipment and Systems, TETRA; Speech codec for full-rate traffic channel,ETS 300 395-2, European Telecommunications Standards Institute, F-06921 Sophia Antipolis - France, 1995.

PROPOSAL FOR AN ADAPTIVE MULTI-RATE CODER FOR GSM

A. Gerrits, A. Koppelaar, R. Taori, R. Sluijter, C. Baggen, E. Hekstra-Nowacka

Philips Research Eindhoven,
Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands
gerritsa@natlab.research.philips.com

To upgrade the existing GSM system in a backward compatible manner, ETSI launched a call for proposals for adaptive multi-rate (AMR) systems. The AMR system allows for adaptive allocation of the speech and channel coding rates to cope with varying channel conditions. This upgraded system would provide two benefits: a higher speech quality for the subscribers and a higher capacity for the network operators. In this paper, the Philips proposal is presented. A brief description of the speech coder, the channel coder and the AMR controller is provided and the obtained results are discussed.

INTRODUCTION

GSM has become a widely used system for mobile communications. A limitation of the current system is the rapid degradation of the speech quality under poor, yet frequently occurring, channel conditions. This is shown in Figure 1a, where the performance curves of the Full-Rate coder (FR) [1] and the Enhanced Full-Rate coder (EFR) [2] are given. The horizontal axis shows the channel condition, denoted by Carrier-to-Interference level (C/I), whereas the vertical axis shows the subjective speech quality expressed on a Mean Opinion Score (MOS) scale (5: excellent, ..., 1: poor) [3]. The speech quality of both coders is acceptable for C/I $\geq$ 10 dB, but degrades rapidly for poorer channel conditions. It is desirable to avoid this sudden break-down of the quality in favour of a graceful degradation of the speech quality in deteriorating channel conditions.

The Adaptive Multi-Rate (AMR) system attempts to achieve this by allocating the speech and channel coding rates as a function of the channel condition. In Figure 1b, the performance curves of several coders operating at different speech/channel bit rate allocations are put together. In order to obtain the performance curve indicated by the dashed line in Figure 1b, the best rate allocation has to be selected. A channel quality measure is needed to realize this.

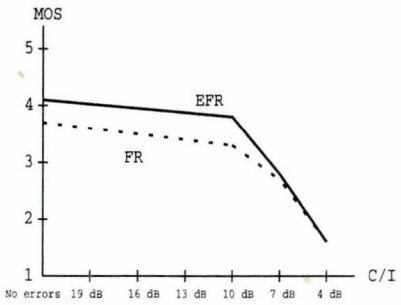


Figure 1a: FR/EFR performance curves.

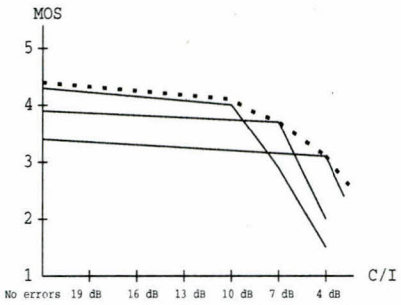


Figure 1b: Performance curves (conceptual) of several coders (solid lines). Desired performance curve (dashed line).

An overview of the AMR system is given in Figure 2. The system consists of a base station and a mobile, which communicate with each other via traffic channels (up-link and down-link). Two types of traffic channels are defined in the current GSM network: the Full Rate Traffic Channel (TCH/FR) with a bit rate of 22.8 kbit/s, and the Half Rate Traffic Channel (TCH/HR) with a bit rate of 11.4 kbit/s. Let us consider the down-link (the link from base station to mobile). In the mobile, a channel quality estimate for the down-link is derived from the channel decoder. This is sent via the up-link to the control system in the base station. The control system determines the rate allocation which suits the current channel condition on the down-link best. The up-link is controlled in a similar way.

An essential requirement for the AMR system is a robust channel quality measure. This measure should be able to track rapid variations in channel quality such that the rate allocation of the speech and channel coder can be changed adequately. It is also required that the speech coder should switch between different bit rates without introducing artefacts (seamless switching). Furthermore, transparency of the AMR system to various background noise conditions is required. In order to meet the quality requirements, most of the available bit rate of the traffic channel should be used for the speech and channel coder, and the amount of overhead bits (e.g. rate information, additional signalling bits) has to be reduced to a minimum. In the next sections, the speech coder, the channel coder and adaptation system will be discussed.

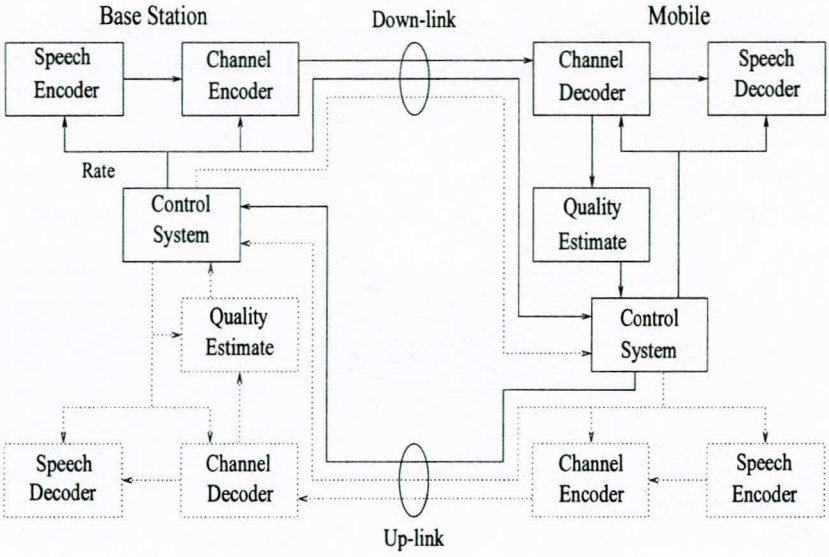


Figure 2: Schematic overview of AMR system.

SPEECH CODER

To meet the quality requirements over the entire C/I range, a number of bit rates have to be considered for the speech coder (Figure 1b). In order to facilitate seamless switching between these bit rates, one type of speech coding scheme is used. The speech coder of the proposal belongs to the family of LPC-based analysis-by-synthesis coders [4] using regular pulse excitation (RPE) [5].

The speech coder operates on speech segments of 20 ms (frames). A typical example is shown in the top trace of Figure 3. The short-term correlation in the original speech is removed by applying 10th order Linear Prediction (LP). The resulting signal is depicted in the middle trace, where long-term correlation is still present in the form of pitch contributions, which is removed using an adaptive code book (ACBK) with sub-sample pitch resolution [2]. The bottom trace in the figure shows the signal after removing long-term and short-term correlations. This signal is modelled using a code book of RPE sequences. An RPE sequence consists of regularly spaced pulses in time according to a down-sampling scheme. The amplitudes of the pulses are restricted to ternary values (+1, 0, -1). The encoder determines the ACBK and RPE parameters in an analysis-by-synthesis manner.

In Table 1 the bit allocations and update rates of the parameters are given. It lists the 5 bit rates used in TCH/FR and TCH/HR. Switching between the traffic channels is provided by the GSM system, while switching between the bit rates in the same traffic channel is provided by the AMR control system. The variability in bit rate is for the major part achieved by varying the decimation factor of the RPE. Switching between bit rates is possible every 20 ms.

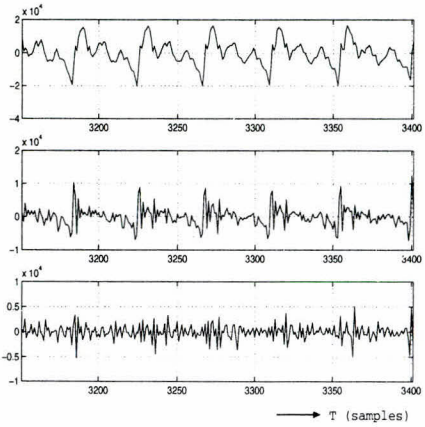


Figure 3: Typical signals.

Bit rate (kbit/s)	11.1	9.3	8.1	7.0	5.5
LPC (bits/20ms)	28	28	28	28	22
ACBK pitch (bits/10ms)	13	15	13	12	12
ACBK gain (bits/5ms)	4	4	4	3	2
RPE sequence (bits/5ms)	33	23	18	15	10
RPE gain (bits/5ms)	5	5	5	4	4
ACBK resolution	1/3	1/6	1/3	1/3	1/3
Decimation factor	2	3	4	5	8

Table 1: Bit allocation of the speech coder.

Another important requirement is the performance under background noise conditions. Under clean speech conditions, the coding emphasis is placed on the perceptually relevant low frequencies of the speech so that they are represented most accurately. When significant background noise is present, this emphasis is relieved so as to make the coder more transparent.

In the speech decoder, the speech quality of the synthesised speech is enhanced by using a post-filter. When applied on speech with background noise, the post-filter decreases the transparency. Therefore, when operating under speech with background noise, post-filtering is not applied. The presence of background noise is estimated from the synthesised speech, saving additional signalling.

CHANNEL CODER

Prior to transmission of the speech bits, the channel coder adds redundancy such that transmission errors can be corrected. Besides, two other types of information have to be communicated between mobile and base station, namely the

rate allocation and the channel quality estimate (see Figure 2). The rate allocation is communicated via the so called RA (Rate Assignment) bits. These bits are protected with a separate code, which is spread over 2 frames to reduce the amount of overhead per frame. The receiver (mobile or base station) decodes this code first in order to extract the rate allocation. There are 2 RA-bits encoded in a code word, such that 4 rate allocations can be addressed in a particular traffic channel. The channel quality estimate is also transmitted in two frames (1 bit per frame) to minimise overhead.

A small number of speech bits can be left unprotected. Most of the bits however, are protected by a channel code. An 8-bit Cyclic Redundancy Check (CRC) is added to a frame in order to detect the erroneous decoding of a frame. The CRC covers most of the speech bits and the channel quality estimate bit. The unprotected speech bits are not covered by the CRC.

The CRC-covered speech bits, the channel quality estimate bit and the CRC bits themselves are encoded with a single channel code. In order to enable efficient (near-) Maximum Likelihood (ML) decoding, convolutional codes are used [8]. To avoid the overhead due to zero-tailing, we use tail-biting convolutional codes. As a compromise between decoding complexity and performance, a 32-state encoder is deployed. The codes are tailored to the appropriate coding rate by using puncture tables [7].

A Viterbi decoder is used for decoding the tail-biting convolutional codes. Due to the tail-biting construction, start and end states are the same and the decoder has no information about this state. So, an ML decoder has to try all possible states, resulting in decoding the received frame 32 times. Since valid code words can be represented as closed paths in the circularly folded code trellis, the received word can be circularly extended. The employed decoder does this extension for a small fraction of the received word. The circularly extended word is decoded once by a Viterbi decoder using dedicated choices of begin and end states, resulting in a significantly reduced decoding complexity compared to ML decoding. Although the decoding is suboptimal, experiments have shown that the loss in performance is negligible.

ADAPTATION SYSTEM

The 4 rate allocations for each traffic channel are listed in Table 2. The results of listening tests conducted with the specified rates are shown in Figure 4. If the adaptation system selects the coding rate with the highest speech quality

TCH/HR		TCH/FR	
Speech (kbit/s)	R (ECC)	Speech (kbit/s)	R (ECC)
5.5	1/2	5.5	1/4
7.0	5/8	8.1	3/8
8.1	3/4	9.3	3/7
9.3	6/7	11.1	1/2

Table 2: Rate allocation.

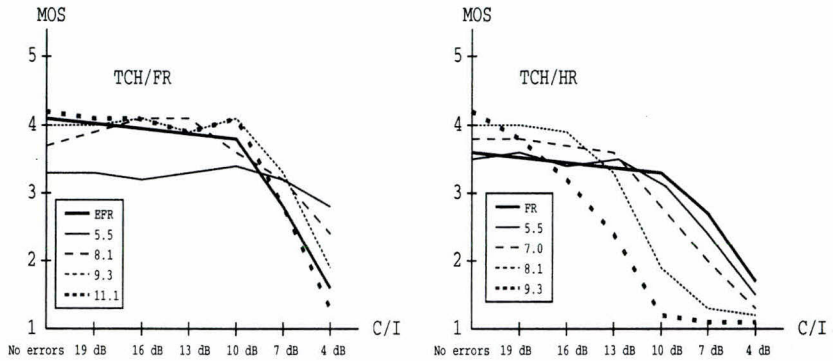


Figure 4: Listening tests results.

for each channel condition, the quality of the AMR system in TCH/FR for poor channel conditions is better than EFR. In TCH/HR, the quality of the AMR system in case of optimal allocation of rates compared to the FR coder is better for good channel conditions and slightly worse for poor conditions.

Measuring the channel quality was a difficult task. A good indicator for the channel quality is the bad frame rate. The detection of a bad frame can be done reliably using the CRC. However, estimating the bad frame rate in this way requires a considerable observation time, resulting in a slow response to rapidly changing channel conditions. It is desirable to have a channel quality indicator which is near-instantaneous for tracking these changes.

The minimum metric difference (MMD) [6], used in the AMR system, is one such measure. The MMD is obtained from the Viterbi decoder, where the metric of the most likely code word is tracked. When the second most likely code word has a metric which is close to the metric of the most likely word, the decision is unreliable, while a large metric difference indicates a reliable decision. By definition the metric difference between the best and the second best code word is the MMD. The calculation of the MMD is efficiently combined with the

Add-Compare-Select operations of the Viterbi decoder. Prior to transmission, the MMD is low-pass filtered and quantised using 2 bits, where one bit is transmitted per frame as the channel quality estimate.

Specific channel conditions and a given channel code result in typical values for MMDs. Figure 5 shows the MMD distribution for a C/I of 4, 7 and 10 dB in TCH/FR with a channel coding rate of 3/8. A clear distinction between the various channel conditions can be made.

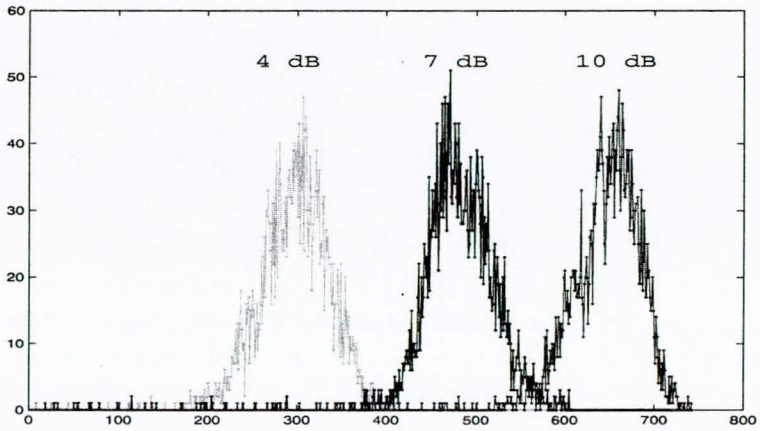


Figure 5: Distribution of MMD values for various channel conditions.

Listening tests with dynamically changing channel conditions support the MMD to be a reliable and an effective metric for measuring the channel quality. These tests also showed that the AMR system has a better speech quality than the EFR coder in TCH/FR, thereby showing that the adaptation system is functioning desirably. The speech quality of the AMR system in TCH/HR, under channel conditions with C/I $>$ 13 dB, has been judged superior in quality to the FR coder, which uses twice the gross bit rate. In cases where the C/I ratio becomes low, the quality of the FR coder is rated higher.

CONCLUSIONS

Using the AMR concept it is possible to offer graceful degradation. While the system is vulnerable in circumstances where the channel quality changes faster than the response time, it does give a clear overall improvement. Further, the proposed system outperforms the EFR coder in TCH/FR. However, the requirements set in TCH/HR could not be met. For worse channel conditions (i.e. C/I $<$

10 dB) a protection with rate 1/2 was found to be insufficient. Listening tests and real-time simulations showed, however, that a graceful degradation is achieved up to 4 dB C/I in TCH/FR in contrast to the 10 dB C/I for the current EFR and FR coders. In TCH/HR, a higher speech quality compared to the FR coder is possible down to 13 dB C/I. The real-time system will be demonstrated at the symposium.

REFERENCES

[1] P. Vary, R. Hoffman, K. Hellwig and R. Sluijter, *A regular-pulse excited linear predictive code.* Speech Comm., Vol. 7, no. 2, pp. 1054-1063, 1988.

[2] K.Järvinen, et al., *GSM Enhanced Full Rate Speech Codec*, Proc. Int. Conf. on Acoustics, Speech and Signal Proc., Munich, pp 771-774, 1997.

[3] P. Kroon, *Evaluation of Speech Coders*, Chapter 13 in *Speech Coding and Synthesis*, W.B. Kleijn and K.K. Paliwal (editors), Amsterdam, Elsevier, pp. 467-494, 1995

[4] M.R. Schroeder and B.S. Atal, *Code-excited linear prediction (CELP): high quality speech at very low bit rates.* Proc. Int. Conf. on Acoustics, Speech and Signal Proc., Tampa, pp. 937-940, 1985.

[5] P. Kroon, E.F. Deprettere and R.J. Sluijter, *Regular-pulse excitation - A novel approach to effective and efficient multipulse coding of speech*, IEEE Trans. on Acoustics, Speech, and Signal Proc., Vol. ASSP-34, no. 5, pp. 1054-1063, 1986.

[6] *A decoding and bad frame detection algorithm for TCH/HR.* Annex C of ETSI document ETS 300 575:1994 (GSM 05.03).

[7] I. E. Bocharova and B.D. Kudryashov, *Rational Rate Punctured Convolutional Codes for Soft-Decision Viterbi Decoding.* IEEE Trans. on Information Theory, Vol. 43, No. 4, pp. 1305-1313, 1997.

[8] A. Dholakia, *Introduction to Convolutional Codes with Applications.* Boston, Kluwer Academic Publishers, 1994.

LINEAR FEATURE TRANSFORMATIONS BASED ON MCE AND MMI

Kris Demuynck and Patrick Wambacq
Katholieke Universiteit Leuven, ESAT–PSI
Kard. Mercierlaan 94, B–3001 Heverlee, Belgium
Kris.Demuynck@esat.kuleuven.ac.be

This paper describes an alternative to the commonly used linear discriminant analysis (LDA) for finding linear transformations that map large feature vectors onto smaller ones while maintaining most of the discriminative power. The main problem with LDA is that it over-simplifies the situation by condensing all class information into only two scatter matrices, thereby losing important information on the individual class distributions. We therefore propose a new set of methods, based on the mutual information or minimum classification error paradigm, which take all information on the individual class distributions into account while searching an optimal sub-space, thus avoiding the crude approximations done by LDA. Experiments show that the proposed schemes provide more discriminative feature sets, leading to better classification results.

INTRODUCTION

The performance of a pattern classifier strongly depends on the input features: to obtain a good performance the feature vector needs to be both highly discriminative and compact. Linear discriminant analysis is a frequently used data-driven method to find linear transformations that map large feature vectors onto smaller ones with minimal loss in class discrimination according to some criterion.

Linear discriminant analysis (LDA) [1] thus searches for a linear transformation $T : x \rightarrow y, \mathcal{R}^q \rightarrow \mathcal{R}^r$ which maps a q dimensional input vector x onto a lower dimensional vector $y = Tx$ while retaining the maximum amount of class discrimination information. In order to do so LDA needs the individual conditional class density functions $f(x|c_i)$ and the a-priori class probabilities $W_i = p(c_i)$ for all N_c classes. In that, LDA assumes that each class density function $f(x|c_i)$ can be adequately modelled with a single multi-dimensional gaussian $\mathcal{N}(x;\mu_i,\Sigma_i)$. Next this information in condensed in two scatter matrices: (1) the within class scatter matrix $\Sigma_w = \sum W_i\Sigma_i$ and (2) the between class scatter

matrix $\Sigma_b = \sum W_i(\mu_i - \mu)(\mu_i - \mu)'$, with $\mu = \sum W_i \mu_i$, or alternatively the overall scatter matrix $\Sigma_t = \Sigma_b + \Sigma_w$. LDA then maximises the ratio of the overall variance to the within class variance.

$$\hat{T} = \underset{T}{\operatorname{argmax}} \frac{|T\Sigma_t T'|}{|T\Sigma_w T'|} = \underset{T}{\operatorname{argmax}} \frac{|T\Sigma_b T'|}{|T\Sigma_w T'|}$$

It can be shown that a solution to the above equation can be found by solving the generalised eigenvalue problem $\Sigma_b V = \Sigma_w V \Delta$ with Δ a diagonal matrix containing the eigenvalues of $\Sigma_w^{-1}\Sigma_b$. The r eigenvectors corresponding to the r largest eigenvalues then make up the rows of matrix $\hat{T}$.

The generalised eigenvalue decomposition can also be seen as a cascade of two eigenvalue decompositions, giving a more intuitive insight into what LDA does. A first transform T_1 converts the feature vectors so that the average within class scatter matrix $\Sigma_w^{(T)} = T_1 \Sigma_w T_1'$ equals the identity matrix. A second transform T_2 then looks for the principal axes of the transformed between class scatter matrix $\Sigma_b^{(T)} = T_1 \Sigma_b T_1'$, thus maximising the average distance between the class means $\mu_i^{(T)}$ while assuming an identity scatter matrix for all classes.

The problem with LDA is that it condenses all information into only two scatter matrices. The solution found by LDA is a good solution on average over all problems with the same two averaged scatter matrices, but the solution may not be particularly good for a specific case. See figure 1 for some examples where LDA fails to find a good solution.

Heteroscedastic discriminant analysis (HDA) [2, 3] improves upon LDA by also incorporating the individual scatter matrices into the optimisation. The individual class means are however still ignored. HDA is based on the observation that LDA can also be derived as a maximum likelihood (ML) method for normal populations with different means and common covariances [4]. In the ML approach we have to look for a transformation matrix $T : x \rightarrow y$, $\mathcal{R}^q \rightarrow \mathcal{R}^q$ and model parameters $\mu_i^{(y)}$ and $\Sigma^{(y)}$ (or $\Sigma_i^{(y)}$ in the case of HDA) that maximise the likelihood of the training data when evaluated in the original - non transformed - x domain. The model parameters that correspond to the last $q - r$ elements in the transformed domain are shared among all N_c classes, this way forcing as much class specific information as possible into the first r parameters.

HDA outperforms LDA if the classes have different covariances (heteroscedasity). HDA for example finds the optimal solution for the situation depicted at the left side of figure 1 while LDA fails. If all classes have the same covariance, HDA finds the same solution as LDA. HDA thus still fails on the situation

depicted at the right side of figure 1. Reason for this failure is the maximum likelihood criterion used in HDA: maximum likelihood does not directly target the most discriminative sub-space, good discrimination between classes is only a secondary effect, as demonstrated by the fact that maximum likelihood optimisation still ignores vital clues like the individual class means.

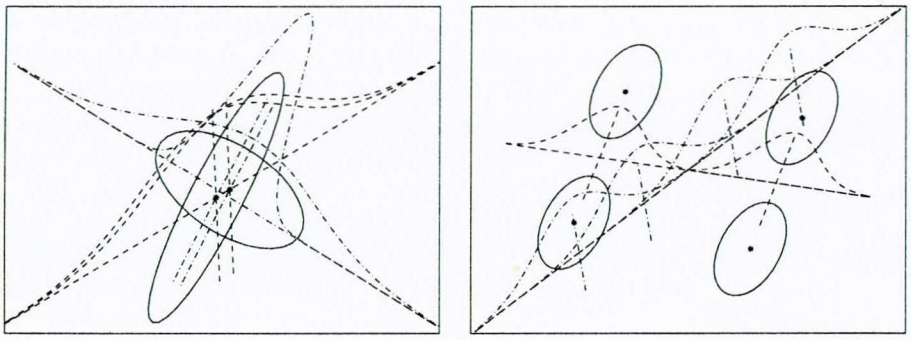


Figure 1: Even in simple cases LDA may fail to find a good solution. Each ellipse is the contour-line of a two dimensional gaussian. The dashed line shows the principal axis found by LDA and the projected class distributions. The dot-dashed curves show the solution found by the MMI-approximation.

APPROXIMATION OF THE MMI AND MCE CRITERION

Both LDA and HDA are sub-optimal because the criteria used - ratio of variances and maximum likelihood - are inadequate. In an effort to remedy this we have chosen to at least start with an adequate criterion while making concessions in the evaluation if necessary. The two criteria explored in this paper are mutual information and minimum classification error (MCE). As for the conditional class density functions $f(x|c_i)$, we no longer assume a normal distribution but allow any mixture of gaussians. Suppose we have a set of N_g gaussians for the N_c classes, then the class density functions can be expressed as follows:

$$f(x|c_i) = \sum_{j=1}^{N_g} \lambda_{ij} \mathcal{N}(x; \mu_j, \Sigma_j)$$

and

$$f(x, c_i) = \sum_{j=1}^{N_g} w_{ij} \mathcal{N}(x; \mu_j, \Sigma_j), \qquad (w_{ij} = W_i \lambda_{ij})$$

This formulation allows for one specific gaussian per class (cf. LDA) as well as for gaussian mixtures with or without shared gaussian.

Mutual information: The class density functions in the original q dimensional space contain all available information on the classes. If we want to reduce the dimensionality of the feature vector with minimal loss of information on the actual class densities, then the transformed density functions should resemble the original density functions as good as possible. In other words, the distance between the original and the transformed density functions should be minimal. Using the Kullback Leibler distance measure [5], following results were obtained:

$$\begin{aligned} D(p(C|X)||p(C|Y=TX)) &= \int_{\mathcal{R}^q} \sum_{i=1}^{N_c} f(c_i,x) \log_2 \frac{p(c_i|x)}{p(x_i|y=Tx)}\, dx \\ &= \int_{\mathcal{R}^q} \sum_{i=1}^{N_c} f(c_i,x) \log_2(p(c_i|x))\, dx - \int_{\mathcal{R}^q} \sum_{i=1}^{N_c} f(c_i,x) \log_2(p(c_i|y=Tx))\, dx \\ &= \int_{\mathcal{R}^q} \sum_{i=1}^{N_c} f(c_i,x) \log_2(p(c_i|x))\, dx - \int_{\mathcal{R}^r} \sum_{i=1}^{N_c} f(c_i,y) \log_2(p(c_i|y))\, dy \\ &= H(C|Y) - H(C|X) \end{aligned}$$

Finding a transformation $T : \mathcal{R}^q \rightarrow \mathcal{R}^r$ that minimises the Kullback Leibler distance is thus equal to finding a transformation that minimises the conditional class entropy [5] $H(C|Y)$. As x uniquely determines y ($y = Tx$) we can also write:

$$D(p(C|X)||p(C|Y)) = H(C|Y) - H(C|X,Y) = I(C;X|Y)$$

The transformation thus also has to *minimise* the conditional mutual information [5] $I(C;X|Y)$, i.e. the original feature vector x should provide minimal additional information on the classes on top of the reduced feature vector y.

The conditional entropy $H(C|Y)$ can be expressed as a function of the class density functions as follows:

$$\begin{aligned} H(C|Y) &= \int_{\mathcal{R}^r} \sum_{i=1}^{N_c} f(c_i,y) \log_2(p(c_i|y))\, dy \\ &= \sum_{i=1}^{N_c} \sum_{j=1}^{N_g} w_{ij} \int_{\mathcal{R}^r} \mathcal{N}(y;\mu_j,\Sigma_j) \log_2 \left(\frac{\sum_{k=1}^{N_g} w_{ik} \mathcal{N}(y;\mu_k,\Sigma_k)}{f(y)} \right) dy \end{aligned}$$

If we assume that the gaussians are highly localised functions (cf. Dirac impulses),

then we can move the term $\mathcal{N}(y;\mu_j,\Sigma_j)$ into the logarithm and into the fraction:

$$\begin{aligned} H(C|Y) &\approx \sum_{i=1}^{N_c}\sum_{j=1}^{N_g} w_{ij}\log_2\left(\frac{\sum_{k=1}^{N_g} w_{ik}\int_{\mathcal{R}^r}\mathcal{N}(y;\mu_j,\Sigma_j)\mathcal{N}(y;\mu_k,\Sigma_k)\,dy}{\sum_{k=1}^{N_g} W_k\int_{\mathcal{R}^r}\mathcal{N}(y;\mu_j,\Sigma_j)\mathcal{N}(y;\mu_k,\Sigma_k)\,dy}\right) \\ &\approx \sum_{i=1}^{N_c}\sum_{j=1}^{N_g} w_{ij}\log_2\left(\frac{\sum_{k=1}^{N_g} w_{ik}\mathcal{O}(j,k)}{\sum_{k=1}^{N_g} W_k\mathcal{O}(j,k)}\right) \end{aligned}$$

With $\mathcal{O}(i,j)$ an overlap measure between gaussians i and j:

$$\begin{aligned} \mathcal{O}(i,j) &= \int_{\mathcal{R}^r}\mathcal{N}(y;\mu_i,\Sigma_i)\mathcal{N}(y;\mu_j,\Sigma_j)\,dy \\ &= \frac{\exp\left(-\frac{1}{2}\left(\mu_i'\Sigma_i^{-1}\mu_i+\mu_j'\Sigma_j^{-1}\mu_j-\tilde{\mu}'(\Sigma_i^{-1}+\Sigma_j^{-1})^{-1}\tilde{\mu}\right)\right)}{\sqrt{(2\pi)^r|\Sigma_i^{-1}+\Sigma_j^{-1}||\Sigma_i||\Sigma_j|}} \end{aligned}$$

and

$$\tilde{\mu} = \Sigma_i^{-1}\mu_i + \Sigma_j^{-1}\mu_j$$

The gaussians are of course only moderately well localised, so a rather crude approximation is done. This is especially true when a large gaussian is moved into the logarithm. To compensate a bit for this mismatch, we replaced the overlap measure with a normalised variant $\mathcal{O}'(i,j)$, showing better results for heteroscedastic problems:

$$\begin{aligned} \mathcal{O}'(i,j) &= \frac{\mathcal{O}(i,j)}{\sqrt{\mathcal{O}(i,i)\times\mathcal{O}(j,j)}} \\ &= \frac{\exp\left(-\frac{1}{2}\left(\mu_i'\Sigma_i^{-1}\mu_i+\mu_j'\Sigma_j^{-1}\mu_j-\tilde{\mu}'(\Sigma_i^{-1}+\Sigma_j^{-1})^{-1}\tilde{\mu}\right)\right)}{\sqrt{|\frac{1}{2}(\Sigma_i^{-1}+\Sigma_j^{-1})|\sqrt{|\Sigma_i||\Sigma_j|}}} \end{aligned}$$

This normalised overlap measure was used in all further evaluations. Note that the overlap measure is invariant with respect to linear transformations, thereby making the whole criterion invariant to linear transformations.

Minimum classification error: If a minimum number of classification errors is of prime interest, then MCE is the optimal criterion. As opposed to the mutual information criterion, MCE does not provide optimal density functions, so confidence measures (e.g. the rate of confusion with the second best class) or any other measure that depends on the actual values of the density functions are less reliable. Using the same approximations as with mutual information, following

expression is obtained:

$$\begin{aligned}\mathrm{Err}(C|Y) &= 1 - \int_{\mathcal{R}^r} \max_i \{f(y, c_i)\}\, dy \;=\; 1 - \int_{\mathcal{R}^r} f(y) \max_i \{\frac{f(y, c_i)}{f(y)}\}\, dy \\ &\approx 1 - \sum_{j=1}^{N_g} W_j \max_i \left\{ \frac{\sum_{k=1}^{N_g} w_{ik} \mathcal{O}(j,k)}{\sum_{k=1}^{N_g} W_k \mathcal{O}(j,k)} \right\}\end{aligned}$$

In our implementation we made one more approximation: as we only use mixtures of gaussians with class specific gaussians (no sharing), we assumed that the best class i is the class corresponding to gaussian j. This way, the maximum operator could be removed, resulting in a smoother function which is easier optimise.

Implementation: As we were unable to find a closed form solution for either the MCE or the mutual information criterion, we resorted to numerical optimisation techniques. The multidimensional optimisation was converted into a sequence of uni-dimensional optimisations by splitting the transformation matrix T in a sequence of elementary Givens rotations: $T = \Theta_n \cdots \Theta_2 \Theta_1$. Each rotation swaps information between one of the r first vector components and one of the $q-r$ last vector components. All possible pairs of vector components are repeatedly handled, until global convergence. There is no need to check non orthonormal transformations or internal rotations between the first r vector components because the overlap measure $\mathcal{O}(i, j)$ is invariant to linear transformations. A good starting point for the optimisation is the LDA transformation matrix.

The most costly operations in the evaluation of both criteria are the matrix inversions and the determinants of the covariance matrices. Covariance matrices are symmetric and positive definite, and can be handled twice as fast as full matrices by using a decomposition in the form $\Sigma = L \times L'$, with L a lower triangular matrix. Despite our effort to speed up the calculations, the optimisation remains slow for large problems. This due to the quadratic dependency of the complexity with the number of gaussians N_g and the $r(q - r)$ dependency with the dimensionality of the input and target feature vectors.

EVALUATION ON A SPEECH RECOGNITION TASK

Evaluations on small artificial problems like the ones shown in figure 1 clearly demonstrate the benefits of our new approach over LDA and HDA. We also evaluated our approach on a complex classification problem, namely speech recognition [6].

Current speech recognisers are firmly based on the principles of statistical pattern recognition [7]. The task of a speech recogniser in such a statistical

framework is to find the most likely sequence of words given the acoustic signal. The recogniser therefore relies on statistical models for the knowledge sources involved in speech recognition: acoustic models, language model and pronunciation models. We focused on the acoustic models. Words in a recogniser are modelled as a sequence of basic sounds (phones) which are in turn modelled as a sequence of states in a hidden Markov model (HMM). The task of the acoustic models is to determine the probability of all states for a given acoustic feature vector. We applied different feature reduction techniques to simplify the acoustic models.

The feature vectors used in speech recognition consist of a set of cepstral coefficients with their first and second order time derivatives calculated every 10ms. Adding the derivatives allows us to model non stationary events in the states. The inverse cosine transform (IDCT) that converts the spectral coefficients into cepstral coefficients is an ad-hoc method that is - based on some physical considerations - expected to concentrate all relevant information in the first coefficients. We replaced the IDCT with the transformation matrices obtained by LDA and by the proposed (minimum) mutual information (MMI) criterion. As input features we took all spectral coefficients augmented with their time derivatives. As classes we used the 139 HMM-states (46 phones with 3 states, and one silence state). Each class was modelled with one full covariance gaussian for calculating the transformation matrix, and with a mixture of shared diagonal covariance gaussians in the final acoustic models. The MMI criterion was preferred to MCE because the goodness of a speech recognition system is judged based on the number of word errors. Thus errors in the classification of states are not really important, as long as the cumulated state probabilities over a word are accurate.

nr. of	word error rate/state perplexity/state error rate		
parameters	IDCT	LDA	MMI
9	12.52/5.02/59.43	12.23/5.18/57.61	9.92/4.50/56.52
16	8.33/3.11/52.96	7.58/3.15/51.67	6.94/2.86/51.30
25	6.29/2.38/49.74	6.21/2.40/48.93	5.66/2.26/49.10
39	5.88/1.95/48.26	6.09/1.95/47.49	6.00/1.88/47.94

Table 1: Results on the ARPA RM test-sets (feb89 + oct89 + feb91 + sep92) with the standard Word Pair Grammar (context-independent modelling).

Table 1 shows the results obtained on the ARPA Resource Management speech recognition task. The first number is the percentage of misrecognized words, the second number is the state perplexity, and the last number is the state error rate. The perplexity equals $2^{-H(C|Y)}$ and can be interpreted as the number of equiprobable choices that would give rise to the same entropy. The

state perplexity and state error rate are calculated on the training data-base and are in fact the quantities that are to be minimised by the MMI and MCE criterion respectively. The substantial decrease in perplexity and word error rate demonstrates the effectiveness of the proposed method, and shows that the approximation involved in the evaluation of the mutual information criterion does not really harm the optimisation.

CONCLUSIONS

In this paper we described an alternative to the commonly used linear discriminant analysis for finding linear transformations that map large feature vectors onto smaller ones while maintaining most of the discriminative power. We showed that LDA and HDA are inadequate in some situations, and showed that the main reason for this failure are the criteria on which LDA and HDA are based. We then introduced a new set of methods based on more powerful criteria like mutual information and minimum classification error, and showed that these criteria can be converted, with some approximations indeed, to cost functions that can be numerically optimised. Evaluations on small artificial problems clearly demonstrated the benefits of the new approach. The new approach also performed substantially better than LDA when applied to a speech recognition task. Drawbacks of the new method are the approximations involved in the evaluation of the cost functions, and the complex numerical optimisation needed to solve the problem.

REFERENCES

[1] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*, pages 114–121. John Wiley & Sons, 1973.

[2] N. Kumar. *Investigation of Silicon-Auditory Models and Generalization of LDA for Improved Speech Recognition.* PhD thesis, John Hopkins Univ., 1997.

[3] N. Kumar and A.G. Andreou. Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition. *Speech Communication*, 26(4):283–297, Dccember 1998.

[4] N.A. Campbell. Canonical variate analysis - a general formulation. *Australian Journal of Statistics*, 26:86–96, 1984.

[5] T. Cover and J. Thomas. *Elements of Information Theory.* John Wiley & Sons Inc., 1991.

[6] S. Young. A review of large-vocabulary continuous-speech recognition. *IEEE Signal Processing Magazine*, pages 45–57, September 1996.

[7] F. Jelinek. *Statistical Methods for Speech Recognition.* MIT Press, Cambridge, Massachusetts, 1997.

GENERALIZATION CAPACITY OF NEURAL NETWORKS FOR THE CLASSIFICATION OF OVARIUM TUMOURS

E. Lerouge, S. Van Huffel

Katholieke Universiteit Leuven, div. E.S.A.T.–SISTA/COSIC
K.Mercierlaan 94, B–3001 Heverlee, Belgium
Ellen.lerouge@esat.kuleuven.ac.be

An experienced surgeant is able to make a prediction about the malignancy of an ovarium tumour by means of the presented medical information (such as medical images, morphology and demographic features). To support this classification process, we try to automate the prediction by means of a supervised neural network. A neural network is able to deduce unknown relations from the data by means of its learning capacity and to generalise towards unseen cases (not presented in the training set). Therefore, it is suited for this biomedical classification task. To obtain a good generalization, large amounts of data have to be presented to the neural network. Unfortunately, we only have a small data set at our disposal, such that the risk of memorising the examples in the training set increases. This phenomenon is called overtraining. Since a wrong diagnosis has very strong consequences on the welfare of the patient, the generalization capacity of the network has to be examined thoroughly. We will study two techniques to avoid overtraining: early stopping *and* regularization.

INTRODUCTION

A reliable classifier for preoperative discrimination between benign and malignant ovarium tumours would be of considerable help to clinicians. It would assist them to select patients for whom minimally invasive surgery or conservative management suffices versus those for whom urgent referral to a gynecologic oncologist is needed.

There have already been several attempts to automate the classification process, such as the risk of malignancy index and logistic regression [1]. In this paper, we will make use of a new type of classifier based on a supervised neural network. Neural networks have already been succesfully applied for diagnosis in the biomedical area. The reason for their success lies in the learning capacity to

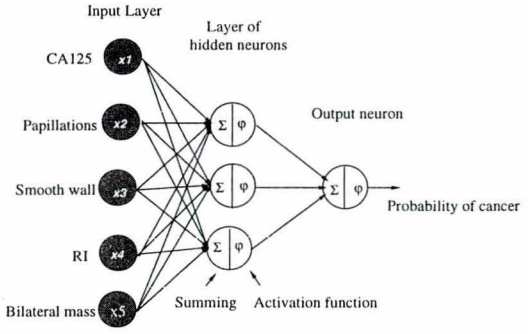
extract unknown relationships from the data and the **generalization** possibility towards unseen cases. As opposed to logistic regression, a neural network is able to make a nonlinear classsification.

The most important unresolved problem in practical applications of feedforward neural networks is how to determine the appropriate complexity of the network, in order to avoid overtraining. In this paper, two techniques are presented to overcome overtraining and their better performance is confirmed via simulation results.

METHOD

The neural network design and training

The most commonly used neural network structures for classification tasks, such as the discrimination between benign and malignant tumours, are multilayer feedforward networks. As illustrated in figure 1, these networks have a layered *architecture*: the information enters the network in the input layer, is fed forward into the hidden neuron layers where it is processed to result in the output layer. Given a certain architecture, the interconnection weights and biases parametrically determine the functional dependence of the output from the input. The functional dependency, we use in the simulations is given in figure 1.



$$y_k = \sigma(\sum_{j=1}^{n_h} w_j(tanh(a_j)))$$

$$a_j = \sum_{i=1}^{m} v_{ij} x_k^i + \beta_j$$

$$\sigma(z) = \frac{1}{1+e^{-z}}$$

m = number of inputs

n_h = number of hidden units

Figure 1: A supervised multi-layer feedforward neural network with one hidden layer

There is a discrepancy between the output of the classifier y_k and the desired output d_k. Adjusting the weights and biases via examples of correct classification, so as to minimise this discrepancy in an iterative process, is referred to as supervised learning. To measure the discrepancy, the cost fucntion is often chosen to

be the squared difference between the desired output d_k and the actual response y_k, averaged over all N samples, $J(\theta) = \frac{1}{N} \sum_{k=1}^{N} (d_k - y_k(\theta))^2$.

If enough examples are available and if appropriate learning methods (Simulated annealing [2]) are used, the output of the overall neural network can be trained to approximate e.g. the *a posteriori probability* of malignancy given the clinical data of patients.

Overtraining

During learning, the outputs of a supervised neural net approximate the target values better and better given the inputs in the training set. This ability may be useful in itself, but more often the purpose of using a neural net is to generalize (e.g. the outputs of the net must approximate the target values as closely as possible given inputs that are not in the training set).

Generalization is *not always possible.* There are several conditions that are typically necessary for good generalization. The inputs to the network have to contain sufficient information pertaining to the target. The network can not learn a nonexistent function between inputs and outputs. Moreover, the training cases have to be a sufficiently large and representative subset of the set of all cases that you want to generalize to. Cases outside the range of the training data or cases inside large 'holes' in the training data require extrapolation. It is important to have sufficient training data to avoid extrapolation, since it is often unreliable.

Even if generalization is possible, neural networks do not automatically generalize. A general rule of thumb is that a network, which is too complex for the available number of examples will easily become overtrained and therefore will have a poor *generalization capacity.* However, if the network is not complex enough, it will not have a good *learning capacity* and the test results will be equally suboptimal. There are basically two ways to avoid *overfitting*: reducing the number of dimensions of the parameter space (e.g. greedy constructive learning, pruning or weight sharing) or reducing the effective size of each dimension (e.g. early stopping or regularisation such as weight decay). In this paper, we will use the second approach, namely early stopping and regularization.

Stopped training [4] is the most common solution to the problem of overfitting. The training of non-linear network models corresponds to an iterative reduction of the error function defined with respect to a set of training data.

During a typical training session, the error with respect to the training set decreases as a function of the number of iterations. The error with respect to an independent test set often shows a decrease at first, followed by an increase as the network starts to over-fit. Early stopping aims to achieve a good generalization by terminating the training when the error on the test set starts to increase again.

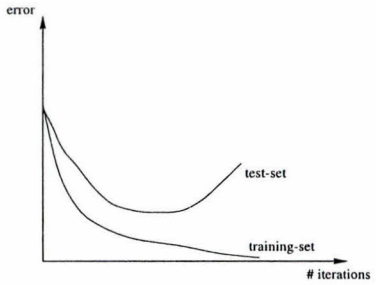


Figure 2: Overtraining in a neural network: the error in the training set decreases monotonically, while the error in the test set starts to increase after a number of iterations

The technique of **regularization** [4] favors smoother network mappings by adding a penalty Ω to the error function $\tilde{E} = E + \nu\Omega$, with ν a parameter called a regularisation coefficient which controls the model complexity. One of the simplest forms of regularizer is called *weight decay* and consists of the sum of the squares of the adaptive parameters $\Omega = \frac{1}{2}\|w\|^2$, where w denotes the vector of all weights and biases in the network. It has been found empirically that a regularizer of this form can lead to significant improvement in network generalization.

This is illustrated for the particular case of a quadratic error function $E = E_0 + \frac{1}{2}(w - w^*)^T H(w - w^*)$ for which the relationship between the original and the regularized error functions is

$$\tilde{w}_j = \frac{\lambda_j}{\lambda_j + \nu} w_j^*,$$

[4] with λ_j the eigenvalues along the principal directions of the quadratic error surface, w^* the minimum of the error function without regularization and $\tilde{w}$ the minimum of the error function with regularization. Along those directions in which the corresponding eigenvalues are relatively large ($\lambda_j \gg \nu$), the minimum

of the error function is shifted very little. Conversely, along directions for which the eigenvalues are relatively small ($\lambda_j \ll \nu$), the corresponding components of the minimum weight vector are suppressed. Thus only those directions for which $\lambda_j \gg \nu$ contribute significantly to fitting the data, with the weight vector components in remaining directions being set to small values by the regularizer. The quantity

$$\gamma = \sum \frac{\lambda_j}{\lambda_j + \nu},$$

defines the *effective* number of parameters in the model. Clearly as ν is increased, the effective number of parameters is reduced.

We can demonstrate a relationship between early stopping and regularization for the case of a quadratic error function. Suppose the initial weight vector $w^{(0)}$ is chosen to be at the origin, and is updated using simple gradient descent $w^{(\tau)} = w^{(\tau-1)} - \eta \Delta E$, where τ denotes the step number and η the learning rate. One can deduce the following relationship between the original and the regularized error functions:

$$\begin{aligned} w_j^\tau &\simeq w_j^* \quad when \quad \lambda_j \gg (\eta\tau)^{-1} \\ |w_j^\tau| &\ll |w_j^*| \quad when \quad \lambda_j \ll (\eta\tau)^{-1} \end{aligned}$$

This proves that regularization and early stopping lead to similar solutions and the quantity $(\eta\tau)^{-1}$ is analogous to the regularization parameter ν.

Evaluation

The most commonly used method for estimating generalization error in neural networks is to reserve part of the data as a test set, which must not be used in any way during training. In the case of the classification of ovarium cancers however, the availability of labelled data is severely limited so that we can not afford to keep part of the data set for model comparison purposes. Cross validation allows us to use all of the data for training. In *k-fold cross-validation*, we divide the data into k subsets of equal size. We train the network k times, each time leaving out one of the subsets from training, using only the omitted subset to compute whatever error criterion interests you. Finally, the test errors are averaged over all k results.

To make an objective validation of the performance of the different classifiers (with and without complexity control), we will use the *Receiver Operating Characteristic* as test criterion. The ROC-curve [5] plots the percentage of correctly

classified of malignant tumours (sensitivity), versus the percentage of false positives ($y_{benign} > T$)(specificity) for varying values of the threshold T. The area under the ROC-curve is a one-value measure of the accuracy of the classifier. The higher this area, the better the classifier. Moreover, this criterium is independent of the fraction of malignant samples in the population. Also for the clinician, it is an interesting criterium. A clinician is not interested in the averaged deviation from desired numbers, but thinks in terms of percentages found, missed or misclassified. The threshold is not fixed, so he can chose a specific threshold corresponding with a certain sensitivity and specificity. To compare different classifiers, we need to compare the area under the ROC-curves in a statistical way [6], based on a z-test.

SIMULATIONS

Inputs and architecture

The data were derived from a study group of 173 patients of which 124 patients had a benign tumour and 49 patients had a malignant tumour. Several indicators such as demographic, serum marker, color Doppler imaging and morphologic indicators were considered. It is beyond the scope of this paper to give a complete description of the meaning of the different indicators that are used. We refer the reader to [1] for a more detailed explanation. The continuous inputs were standardised by subtracting their mean and dividing by their standard deviation. Binary valued inputs were left unchanged. The desired output was labeled 0 for benign samples, 1 for malignant cases.

Since the training set is not large, there is a risk of overtraining. We have already mentioned two approaches to avoid overtraining: reducing the number of dimensions of the parameter space and reducing the effective size of each dimension. We will limit the number of hidden neurons to $n_h = 5$. Then we will use early stopping and regularization to avoid overtraining even more.

Training

We first performed an input selection [8] to determine the most important features. We retain 5 features (log(CA125), Papillations, Smooth wall, Resistance index, Bilateral mass and Ascites).

The neural networks consist of 5 hidden neurons, 5 inputs and 1 output. The parameter vector θ is randomly initialised (zero mean Gaussian distribution,

standard deviations $\sigma = 0.01$). Training is done using Simulated annealing. The cost-function to be minimised is the mean squared classification error. In order to bias the accuracy towards malignancy, a design parameter λ is introduced.

$$J(\theta) = \lambda \frac{1}{N_b} \sum_{k=1}^{N_b} (d_k^b - y_k)^2 + (1 - \lambda) \frac{1}{N_m} \sum_{k=1}^{N_m} (d_k^m - y_k)^2$$

By taking $\lambda = \frac{1}{3}$ in this expression, the role of malignant examples is more emphasised.

Results

In a first neural network (NN1), we do not care about overtraining and do not apply early stopping, nor regularization. In a second neural network (NN2), we try to overcome overtraining by applying early stopping and in a third neural network (NN3) we make use of an extra regularisation term. The performance of the second and third neural network are represented in table 1, respectively in function of the number of iterations and in function of the value of the regularization constant ϱ.

# iterations	$\text{ROC}_{training}$	ROC_{test}
50	96.20	95.40
100	97.45	96.13
200	**97.40**	**97.93**
500	98.00	95.10

ϱ	$ROC_{training}$	ROC_{test}
0	98.17	95.25
0.2	**97.29**	**98.02**
0.4	97.15	97.10
0.8	96.60	92.99

Table 1: Results (left) early stopping and (right) regularization

For the first neural network (NN1), the ROC-curve spans about 95.25% of the total square area of the plot, whereas with early stopping (NN2) and regularization (NN3) it spans respectively 97.93% and 98.02%. Table 2 states the results of the statistical analysis. There is a significant difference between training with and without complexity control. A significant difference between early stopping and regularization was not found.

REFERENCES

[1] D. Timmerman, "Ultrasonography in the assessment of ovarian and

class1 versus class2	z-value	p-value
NN1 versus NN2	3.6678	$< \mathbf{0.003}$
NN1 versus NN3	4.0124	$< \mathbf{0.003}$
NN2 versus NN3	1.3372	$= 0.1817$

Table 2: The influence of early stopping and regularization

tamoxifen-associated endometrial pathology", Katholieke Universiteit Leuven, Dienst Verloskunde en Gynaecologie,1997

[2] Mathworks Inc.,"Optimisation toolbox User's Guide", Mathworks Inc., 1992

[3] Mathworks Inc.," Neural Network toolbox User's Manual", Mathworks Inc., 1992

[4] C. Bishop, "Artificial neural networks for pattern recognition", OUP, Oxford, 1996.

[5] J. A. Hanley and B. J. McNeil, "The meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve", Radiology, Vol. 143, no. 1, p. 29-36, 1982

[6] J. A. Hanley and B. J.McNeil, "A Method of Comparing the Areas under Receiver Operating Characteristic Curves Derived from the Same Cases", Radiology, vol. 148, no. 3, p. 839-843, 1983

[7] S. Haykin, "Neural networks", Macmillan College Publishing Company, New York, 1994.

[8] H. Verrelst, J. Vandewalle, B. De Moor, "Bayesian input selection for neural network classifiers", Katholieke Universiteit Leuven, Departement Elektrotechniek, 1998

GENERALIZED MOMENT INVARIANTS FOR VIEWPOINT AND ILLUMINATION INDEPENDENT COLOR PATTERN RECOGNITION

F. Mindru, T. Moons, and L. Van Gool*

Katholieke Universiteit Leuven, E.S.A.T. / PSI
K. Mercierlaan 94, B–3001 Heverlee, Belgium
Florica.Mindru@esat.kuleuven.ac.be

New invariant features are presented that can be used for the recognition of planar color patterns irrespective of the viewpoint or the illumination conditions, and without the need for error prone contour extraction. The new features are based on moments of powers of the intensities in the individual color bands and combinations thereof. The paper gives a classification of all functions of such moments which are invariant under both affine deformations of the pattern (thus achieving viewpoint invariance) as well as affine changes of the intensity values of the color bands (hence, coping with changes in the irradiance pattern due to different lighting conditions and/or viewpoints). The discriminant power and classification performance of the new invariants for color pattern recognition is tested on a data set of images of real, outdoors advertising panels.

INTRODUCTION

This paper contributes to the viewpoint and illumination independent recognition of planar color patterns such as labels, postcards, pictograms, etc., which typically have a high pictorial content. Much research has been put into invariants for planar shapes under geometrical deformations (see [8, 9] for an overview). For the patterns considered here, the pictorial content usually is too complicated to robustly extract object contours from it. Color information has proven very useful in pattern recognition as well. Much attention has been paid to the illumination independent characterization of the color distribution of the pattern [2, 4, 5, 6, 13, 14]. Color histograms, however, do not exploit the spatial distribution of the colors within the pattern. Another strand of research has focussed

* We would like to thank the financial support from the Belgian IUAP4/24 'IMechS' project and the EU Esprit project 20.515 IMPROOFS

on moment invariants for greyvalue intensity patterns under different types of geometric and / or photometric changes [3, 11, 12, 15]. A limitation of this approach is that one may have to let grow the order of the moments beyond the point where they remain stable in order to create sufficient discriminant power for the invariants. In [17] it was proposed to use powers of the intensities in the individual color bands and combinations thereof in the expressions for the moments to overcome these problems. The resulting so-called *generalized color moments* implicitly characterize the shape, the intensity and the color distribution of the pattern in a uniform manner; and, a broad set of moment invariants can be extracted that never call upon high powers of intensities or spatial coordinates. Invariants of such moments for affine transformations of the pattern and linear photometric changes were derived and tested. Under real world conditions, however, photometric changes include significant non-linear effects, especially in the form of offsets in the color bands. This caused the need for normalisation against intensity offset before computing the value of the invariants in [17].

This paper extends the results of [17] by presenting a classification of color moment invariants under full affine transformations of the intensities in the color bands, thus making intensity normalisation superfluous. The discriminant power and classification performance of these new invariants for color pattern recognition is tested on a data set of images of real outdoors advertising panels taken from different viewing angles and under different weather conditions.

GENERALIZED COLOR MOMENTS

A color pattern can mathematically be represented as a vector-valued function I defined on a region Ω in the (image) plane and assigning to each image point $(x, y) \in \Omega$ the 3-vector $I(x, y) = (R(x, y), G(x, y), B(x, y))$ containing the RGB-values of the corresponding pixel. The *generalized color moments* M_{pq}^{abc} are defined by

$$M_{pq}^{abc} = \iint_{\Omega} x^p y^q \, [R(x, y)]^a \, [G(x, y)]^b \, [B(x, y)]^c \, dxdy \ .$$

M_{pq}^{abc} is said to be a (generalized color) moment of *order* $p+q$ and *degree* $a+b+c$. Observe that generalized color moments M_{pq}^{000} of degree 0 in fact are the (p, q)-shape moments of the image region Ω ; and, that the generalized color moments of degree 1, viz. M_{pq}^{100}, M_{pq}^{010}, M_{pq}^{001}, are just the (p, q)-intensity moments of respectively the R-, G- and B-color band. On the other hand, the generalized color

moments M_{00}^{abc} of order 0 are the non-central (a, b, c)-moments of the (multivariate) color distribution of the RGB-values of the pattern. Hence, these generalized color moments generalize shape moments of planar shapes, intensity moments of greylevel images, and non-central moments of the color distribution in the image. In this paper, only generalized color moments up to the first order and the second degree are considered. In the sequel they will be called "moments" for short.

GEOMETRIC AND PHOTOMETRIC CHANGES

The geometrical deformations of the pattern caused by changes in viewpoint are modelled by *affine transformations*:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

with $|A| = a_{11}a_{22} - a_{12}a_{21} \neq 0$. This implies that the camera is assumed to be relatively far from the object. This assumption together with the fact that the object is planar can greatly simplify the analysis of the photometric changes. Typically, light sources are far from the objects as well. The geometry of light reflection is the same for all points in that case, i.e. they share the same angles of light incidence and camera viewing direction. The change in camera or light position will then result in an overall scaling of the intensity [10]. Furthermore, the combined effect of diffuse and specular reflection can be modeled by an offset in the intensities [16]. Thus, the photometric changes to be considered comprise *scaling combined with an offset* for each color band: i.e. $K'(x, y) = s_K\, K(x, y) + u_K$ with $K \in \{ R, G, B \}$ and $0 \neq s_K, u_K \in I\!R$. The combined effect of the geometric and photometric transformations of the moments M_{pq}^{abc} is given by

$$\begin{aligned} \left(M_{pq}^{abc}\right)' = \iint_\Omega & [\, a_{11}x + a_{12}y + b_1 \,]^p\, [\, a_{21}x + a_{22}y + b_2 \,]^q\, [\, s_R\, R(x, y) + u_R \,]^a \\ & [\, s_G\, G(x, y) + u_G \,]^b\, [\, s_B\, B(x, y) + u_B \,]^c\ \mathrm{abs}(|A|)\ dxdy\ , \end{aligned}$$

which, after expansion, yields a linear combination of moments of order $\leq p + q$ and degree $\leq a + b + c$.

The actions of the geometric and photometric changes on the moments come out to commute. Hence, one might first normalize against one type of transformation and then switch to the use of invariants for the other. In [17] a compromise was made by normalizing against photometric offset and using invariance under

affine transformations and photometric scaling. The resulting invariants may look simpler than the ones given here, but the inherent complexity is at least comparable. In this paper, full invariance under geometric as well as photometric transformations is envisaged.

CLASSIFICATION OF THE COLOR INVARIANTS

As only moments up to the 1st order and the 2nd degree are considered here, the resulting invariants are functions of the generalized color moments M_{00}^{abc}, M_{10}^{abc} and M_{01}^{abc} with $(a,b,c) \in \{(0,0,0), (1,0,0), (0,1,0), (0,0,1), (2,0,0), (0,2,0), (0,0,2), (1,1,0), (1,0,1), (0,1,1)\}$. The invariants can be classified according to 3 degrees of freedom: the *order*, the *degree* and the *number of color bands* of the moments involved. To allow maximal flexibility for the user in the choice of the color bands and to ensure the highest possible robustness of the invariants in the classification, the following 2 criteria were used to construct a basis for the moment invariants: (1) Keep the number of color bands as low as possible; and (2) include as much as possible low-order invariants. This results in the following classification which is obtained by Lie group theoretical methods (see e.g. [7]).

Theorem 1 (1-BAND INVARIANTS) *All geometric / photometric invariants involving generalized color moments up to the 1st order and 2nd degree in only 1 color band are functions of the following basis invariant:*

$$S_{12} = \frac{\{M_{10}^2 M_{01}^1 M_{00}^0 - M_{10}^2 M_{00}^1 M_{01}^0 - M_{01}^2 M_{10}^1 M_{00}^0 + M_{01}^2 M_{00}^1 M_{10}^0 + M_{00}^2 M_{10}^1 M_{01}^0 - M_{00}^2 M_{01}^1 M_{10}^0\}^2}{(M_{00}^0)^2 \left[M_{00}^2 M_{00}^0 - (M_{00}^1)^2\right]^3}$$

where M_{pq}^i stands for either M_{pq}^{i00}, M_{pq}^{0i0} or M_{pq}^{00i}.

Theorem 2 (2-BAND INVARIANTS) *All geometric / photometric invariants involving generalized color moments up to the 1st order and 2nd degree in 2 of the 3 color bands are functions of the following 8 basis invariants:*

0th order, 2nd degree :

$$D_{02} = \frac{\left[M_{00}^{11} M_{00}^{00} - M_{00}^{10} M_{00}^{01}\right]^2}{\left[M_{00}^{20} M_{00}^{00} - (M_{00}^{10})^2\right] \left[M_{00}^{02} M_{00}^{00} - (M_{00}^{01})^2\right]}$$

1st order, 2nd degree : S_{12}^1, S_{12}^2,

$$D_{12}^1 = \frac{\left\{ \begin{array}{l} M_{10}^{10} M_{01}^{01} M_{00}^{00} - M_{10}^{10} M_{00}^{01} M_{01}^{00} - M_{01}^{10} M_{10}^{01} M_{00}^{00} \\ + M_{01}^{10} M_{00}^{01} M_{10}^{00} + M_{00}^{10} M_{10}^{01} M_{01}^{00} - M_{00}^{10} M_{01}^{01} M_{10}^{00} \end{array} \right\}^2}{(M_{00}^{00})^4 \left[M_{00}^{20} M_{00}^{00} - (M_{00}^{10})^2\right] \left[M_{00}^{02} M_{00}^{00} - (M_{00}^{01})^2\right]}$$

$$D_{12}^{2} = \frac{\left\{ \begin{array}{l} M_{10}^{20} M_{01}^{01} (M_{00}^{00})^2 - M_{10}^{20} M_{00}^{01} M_{01}^{00} M_{00}^{00} - M_{01}^{20} M_{10}^{01} (M_{00}^{00})^2 + M_{01}^{20} M_{00}^{01} M_{10}^{00} M_{00}^{00} \\ + M_{00}^{20} M_{10}^{01} M_{01}^{00} M_{00}^{00} - M_{00}^{20} M_{01}^{01} M_{10}^{00} M_{00}^{00} + 2 M_{01}^{10} M_{10}^{01} M_{00}^{10} M_{00}^{00} - 2 M_{10}^{01} (M_{00}^{10})^2 M_{01}^{00} \\ + 2 M_{01}^{01} (M_{00}^{10})^2 M_{10}^{00} - 2 M_{10}^{10} M_{01}^{01} M_{00}^{10} M_{00}^{00} + 2 M_{10}^{10} M_{00}^{10} M_{00}^{01} M_{01}^{00} - 2 M_{01}^{10} M_{00}^{10} M_{00}^{01} M_{10}^{00} \end{array} \right\}^2}{(M_{00}^{00})^4 \left[M_{00}^{20} M_{00}^{00} - (M_{00}^{10})^2 \right]^2 \left[M_{00}^{02} M_{00}^{00} - (M_{00}^{01})^2 \right]}$$

$$D_{12}^{3} = \frac{\left\{ \begin{array}{l} M_{10}^{02} M_{01}^{10} (M_{00}^{00})^2 - M_{10}^{02} M_{00}^{10} M_{01}^{00} M_{00}^{00} - M_{01}^{02} M_{10}^{10} (M_{00}^{00})^2 + M_{01}^{02} M_{00}^{10} M_{10}^{00} M_{00}^{00} \\ + M_{00}^{02} M_{10}^{10} M_{01}^{00} M_{00}^{00} - M_{00}^{02} M_{01}^{10} M_{10}^{00} M_{00}^{00} + 2 M_{10}^{10} M_{01}^{01} M_{00}^{01} M_{00}^{00} - 2 M_{10}^{10} (M_{00}^{01})^2 M_{01}^{00} \\ + 2 M_{01}^{10} (M_{00}^{01})^2 M_{10}^{00} - 2 M_{10}^{01} M_{01}^{10} M_{00}^{01} M_{00}^{00} + 2 M_{10}^{01} M_{00}^{10} M_{00}^{01} M_{01}^{00} - 2 M_{01}^{01} M_{00}^{10} M_{00}^{01} M_{10}^{00} \end{array} \right\}^2}{(M_{00}^{00})^4 \left[M_{00}^{20} M_{00}^{00} - (M_{00}^{10})^2 \right] \left[M_{00}^{02} M_{00}^{00} - (M_{00}^{01})^2 \right]^2}$$

$$D_{12}^{4} = \frac{\left\{ \begin{array}{l} M_{10}^{11} M_{01}^{10} (M_{00}^{00})^2 - M_{10}^{11} M_{00}^{10} M_{01}^{00} M_{00}^{00} - M_{01}^{11} M_{10}^{10} (M_{00}^{00})^2 + M_{01}^{11} M_{00}^{10} M_{10}^{00} M_{00}^{00} \\ + M_{00}^{11} M_{10}^{10} M_{01}^{00} M_{00}^{00} - M_{00}^{11} M_{01}^{10} M_{10}^{00} M_{00}^{00} + M_{10}^{10} M_{01}^{01} M_{00}^{10} M_{00}^{00} - M_{10}^{10} M_{00}^{10} M_{00}^{01} M_{01}^{00} \\ + M_{01}^{10} M_{00}^{10} M_{00}^{01} M_{10}^{00} - M_{10}^{01} M_{01}^{10} M_{00}^{10} M_{00}^{00} + M_{10}^{01} (M_{00}^{10})^2 M_{01}^{00} - M_{01}^{01} (M_{00}^{10})^2 M_{10}^{00} \end{array} \right\}^2}{(M_{00}^{00})^4 \left[M_{00}^{20} M_{00}^{00} - (M_{00}^{10})^2 \right]^2 \left[M_{00}^{02} M_{00}^{00} - (M_{00}^{01})^2 \right]}$$

$$D_{12}^{5} = \frac{\left\{ \begin{array}{l} M_{10}^{11} M_{01}^{01} (M_{00}^{00})^2 - M_{10}^{11} M_{00}^{01} M_{01}^{00} M_{00}^{00} - M_{01}^{11} M_{10}^{01} (M_{00}^{00})^2 + M_{01}^{11} M_{00}^{01} M_{10}^{00} M_{00}^{00} \\ + M_{00}^{11} M_{10}^{01} M_{01}^{00} M_{00}^{00} - M_{00}^{11} M_{01}^{01} M_{10}^{00} M_{00}^{00} - M_{10}^{10} M_{01}^{01} M_{00}^{01} M_{00}^{00} + M_{10}^{10} (M_{00}^{01})^2 M_{01}^{00} \\ - M_{01}^{10} (M_{00}^{01})^2 M_{10}^{00} + M_{10}^{01} M_{01}^{10} M_{00}^{01} M_{00}^{00} - M_{10}^{01} M_{00}^{10} M_{00}^{01} M_{01}^{00} + M_{01}^{01} M_{00}^{10} M_{00}^{01} M_{10}^{00} \end{array} \right\}^2}{(M_{00}^{00})^4 \left[M_{00}^{20} M_{00}^{00} - (M_{00}^{10})^2 \right] \left[M_{00}^{02} M_{00}^{00} - (M_{00}^{01})^2 \right]^2}$$

where S_{12}^{i} is the 1-band invariant S_{pq} defined in Theorem 1 and evaluated in the i-th (of the 2) color band(s).

Note that the invariants D_{12}^{1}, D_{12}^{2} and D_{12}^{3} are build of moments involving 1 band only, whereas D_{12}^{4} and D_{12}^{5} also include moments of the form M_{pq}^{11}.

Theorem 3 (3-BAND INVARIANTS) *All geometric / photometric invariants involving generalized color moments up to the 1st order and 2nd degree in all 3 color bands are functions of the following 18 basis invariants:*

0th order, 2nd degree : $D_{02}^{(RG)}$, $D_{02}^{(RB)}$, $D_{02}^{(GB)}$;

1st order, 2nd degree : $S_{12}^{(R)}$, $S_{12}^{(G)}$, $S_{12}^{(B)}$, $D_{12}^{1(RG)}$, $D_{12}^{1(RB)}$, $D_{12}^{1(GB)}$, $D_{12}^{2(RB)}$, $D_{12}^{3(RG)}$, $D_{12}^{3(GB)}$, $D_{12}^{4(RG)}$, $D_{12}^{4(RB)}$, $D_{12}^{4(GB)}$, $D_{12}^{5(RG)}$, $D_{12}^{5(GB)}$, $D_{12}^{5(RB)}$,

where $S_{ij}^{(K)}$ and $D_{pq}^{i(KL)}$ are the 2-band invariants defined in Theorem 2 evaluated in the color bands K and L. In particular, every invariant involving all 3 color bands is a function of invariants involving only 2 of the 3 bands.

RECOGNITION PERFORMANCE OF THE NEW INVARIANTS

The usefulness and the discriminant power of these new invariants has been tested on a data set of images with real changes in viewpoint and illumination. Color



Figure 1: *Examples of color images in the data set of advertising panels.*

	1	2	3	4	5	6	7	8	9	10	11	mean
A	67	0	40	11	7	86	70	88	62	22	0	41
B	87	100	90	100	60	0	90	0	50	89	86	68
C	73	50	80	72	80	93	70	50	75	44	100	72
D	87	67	80	61	93	93	80	25	62	67	86	72
E	80	83	100	72	73	86	67	56	62	62	75	74
F	93	92	100	89	87	79	80	88	62	67	57	81
G	100	92	100	89	93	71	80	75	87	89	86	88
H	100	100	90	100	100	100	100	75	100	100	86	96

Table 1: *Recognition rates (in %) of different types of feature vectors.*

images of different outdoors advertising panels were taken with a betacam camera from various viewpoints at different times of the day and under different weather conditions. Physically different panels representing the same pattern were included as well. Figure 1 shows some examples of such images.

Recognition of a pattern is performed by means of a *nearest neighbour* classifier based on feature vectors consisting of a number of moment invariants. Since our image data set is medium size (approx. 200 samples), the recognition performance of the system is estimated by means of a *leave-one-out* procedure.
For the tests, 11 classes of different patterns were used, each containing 10 to 18 images. The panels were manually cut out from the images. Theoretically, 18 (basis) invariants are available for a complete distinction between the patterns (Theorem 3). But, due to the limited amount of samples and classes, statistically significant conclusions can only be drawn for feature vectors composed of 5 invariants at most [1]. Therefore, numerous feature vectors combining 3 to 5 invariants were selected and tested. The feature vectors are selected according to the types of the constituting invariants; and, the best performing combinations of invariants are selected by a Principal Component Analysis.

As an example, Table 1 shows the recognition rates of 8 of these feature vectors which are representative for the feature vectors used in the test:

$$A = \{ S_{12}^{(R)}, S_{12}^{(G)}, S_{12}^{(B)} \} , \qquad E = \{ S_{12}^{(R)}, D_{02}^{(BG)}, D_{12}^{1(RB)} \} ,$$
$$B = \{ D_{02}^{(RG)}, D_{02}^{(RB)}, D_{02}^{(GB)} \} , \quad F = \{ S_{12}^{(R)}, D_{02}^{(BG)}, D_{12}^{1(RB)}, D_{12}^{4(RG)} \} ,$$
$$C = \{ S_{12}^{(B)}, D_{02}^{(BG)}, D_{12}^{1(BG)} \} , \quad G = \{ S_{12}^{(R)}, D_{02}^{(BG)}, D_{12}^{1(RB)}, D_{12}^{4(RG)}, D_{12}^{2(RG)} \} ,$$
$$D = \{ S_{12}^{(G)}, D_{02}^{(BG)}, D_{12}^{1(BG)} \} , \quad H = \{ D_{02}^{(RG)}, D_{02}^{(RB)}, D_{02}^{(BG)}, D_{12}^{1(RG)}, D_{12}^{4(RG)} \} .$$

The columns in Table 1 (numbered from 1 to 11) give for each pattern the number (in %) of corresponding images in the data set that are correctly recognized by the given feature vector. The average performance of each vector over all the 11 different patterns is indicated in the last column. The outcome of the experiments is discussed in more detail in the subsequent section.

Recognition performance on real data

First, the recognition rates of the 2-band invariants in Theorem 2 were investigated by considering the feature vectors formed by applying every invariant to each (combination of the) color band(s) (cf. A and B in Table 1). The average recognition rate of the different triplets ranges from 41% to 68% correct matches. Taking combinations of 3 different invariants (cf. C, D, E in Table 1) improves the recognition performance. Their average recognition rates now vary between 55% and 76%. Feature vectors C and D are examples of combinations of 3 different invariants but applied to only 1 pair of color bands. Note that they perform better than the best triplet formed with the same type of invariant, viz. B. Furthermore, it is important to observe the differences in the recognition rates of C and D on individual patterns, although they only differ in one component and have the same average recognition rate. This proves that each invariant captures different characteristics of a color pattern. When taking combinations of 3 different invariants applied to different combinations of the color bands, the average recognition rates are still improving, as can be seen from E in Table 1. One reason for having a better performance when using all 3 bands is that these feature vectors cope better with correlations between the bands. Therefore, the best way of capturing the information contained in all 3 bands in a uniform manner is to use "symmetric" groups of invariants, which are evaluated on the 3 bands and combining different types of invariants. Finally, increasing the length of the feature vectors significantly improves the recognition rate as well, as can be seen from E, F and G in Table 1. The best recognition performance for the given data set with a feature vector of length less than 6 is obtained by the invariant vector H in Table 1. Its average recognition rate is 96%. Hence, acceptable recognition rates can be obtained with only 5 of the 18 basis invariants in Theorem 3. Using

all 18 invariants is expected to make these new invariants well suited for use with large data sets.

REFERENCES

[1] R. O. Duda, P. E. Hart, *Pattern classification and scene analysis*, John Wiley, 1973.

[2] G. Finlayson, B. Schiele, J. Crowley, Comprehensive colour image normalisation, *Proc. ECCV '98*, LNCS 1406, Springer, 1998, pp 475 – 490.

[3] J. Flusser,T. Suk, Pattern recognition by affine moment invariants, *Pattern Recognition*, Vol. 26 (1993), pp. 167 – 174.

[4] D. Forsyth, A novel algorithm for color constancy, *IJCV 5 (1990)*, pp. 5 – 36.

[5] B. Funt, G. Finlayson, Color constant color indexing, *IEEE Trans. PAMI*, Vol. 17 (1995), pp. 522 – 529.

[6] T. Gevers, A. W. M. Smeulders, A comparative study of several color models for color image invariant retrieval, *Proc. Intern. Workshop on Image Database and Multimedia Search*, 1996, pp. 17 – 23.

[7] T. Moons, E. Pauwels, L. Van Gool, A. Oosterlinck, Foundations of semi-differential invariants, *IJCV*, Vol. 14 (1995), pp. 25–47.

[8] J. L. Mundy, A. Zisserman A (eds.), *Geometric invariance in computer vision*, MIT Press, 1992.

[9] J. L. Mundy, A. Zisserman, D. Forsyth (eds.). *Applications of invariance in computer vision*, LNCS 825, Springer, 1994.

[10] M. Oren, S. Nayar, Seeing beyond Lambert's law, *Proc. ECCV '94*, LNCS 800, Springer, pp. 269–280.

[11] R. Prokop, A. Reeves, A survey of moment-based techniques for unoccluded object representation and recognition, *CVGIP: Image Proc.*, Vol. 54 (1992), pp. 438–460.

[12] T. Reiss, *Recognizing planar objects using invariant image features*, LNCS 676, Springer, 1993.

[13] D. Slater, G. Healey, The illumination-invariant recognition of 3D objects using local color invariants, *IEEE Trans. PAMI*, Vol. 18 (1996), pp. 206 – 210.

[14] M. Swain, D. Ballard, Color indexing, *IJCV*, 7 (1991), pp. 11–32.

[15] L. Van Gool, T. Moons, D. Ungureanu, Affine / photometric invariants for planar intensity patterns, *Proc. ECCV '96*, LNCS 1065, Springer, 1996, pp. 642 – 651.

[16] L. Wolff, On the relative brightness of specular and diffuse reflection, *Proc. CVPR '94*, IEEE Comp. Soc. Press, 1994, pp. 369 – 376.

[17] F. Mindru, T. Moons, L. Van Gool, Recognizing color patterns irrespective of viewpoint and illumination, accepted for publication in *Proc. CVPR '99*, to appear.

ON CERTAIN CHANNELS WITH A RANDOM PARAMETER

Mark S. Pinsker*, **Vyacheslav V. Prelov*** and **Edward C. van der Meulen**†

A general class of stationary channels with a random parameter is considered. It is assumed that the random parameter is a completely singular stationary process $U = \{U_n\}$ independent of the input signal $X = \{X_n\}$. Some rather general sufficient conditions are found under which the information rate $\overline{I}(X;Y)$ coincides with the conditional information rate $\overline{I}(X;Y|U)$, where $Y = \{Y_n\}$ is the output signal of the channel considered. Some special cases of such channels are investigated in more detail.

INTRODUCTION

Consider a general model of a stationary channel with a random parameter described by means of a triple of discrete-time jointly stationary stochastic processes ($X = \{X_j\}$, $U = \{U_j\}$, $Y = \{Y_j\}$). The processes X and Y are the input and output signal of the channel, respectively, and the process $U = \{U_j\}$ is a random parameter independent of X. Unlike in previous papers where it was assumed that the random parameter is a *random variable,* we consider here the more general case where the random parameter is a *stochastic process.* Channels with an additive noise or with a multiplicative noise or with an additive-multiplicative noise, where $Y = X + U$, $Y = UX$, or $Y = UX + Z$ ($Z = \{Z_j\}$ is a stationary process independent of X and U), respectively, are some examples of such channels.

We will assume that the random parameter $U = \{U_j\}$ is a completely singular stationary process [1, Chap. 2, §5], i.e., a zero-entropy process. This means, from a descriptive point of view, that the process U is "slowly varying". Note that this case is a natural generalization of the one considered earlier, where it was assumed that the random parameter is a random variable independent of the time instants, i.e., a random variable, which remains fixed during the transmission over the channel. In the case where U is a completely singular process, and under certain additional conditions connected with the requirement of finiteness of the

This work was supported in part by INTAS under Grant 94-469.

* Institute for Information Transmission Problems of the RAS, Moscow, Russia.

† Department of Mathematics, Katholieke Universiteit Leuven, Leuven, Belgium.

corresponding mutual informations, we show that the information rate $\overline{I}(X;Y)$ coincides with the conditional information rate $\overline{I}(X;Y|U)$. The exact result is formulated in Section 2. In certain special cases, this equality allows us to obtain explicit expressions for $\overline{I}(X;Y)$ and to investigate the asymptotic behavior of $\overline{I}(X;Y)$ for weak signals X. The corresponding results are described in Section 3, and in Section 4 the information rates in various models of stationary channels with an additive-multiplicative noise are compared. Preliminary results for the case $Y = UX + Z$ were reported in [2]. The full paper [3], containing the proof of the results described below, has appeared in *Problemy Peredachi Informatsii.*

STATEMENT OF THE RESULTS

Let us first recall some definitions and notions which are necessary to formulate the main results of the paper.

A stochastic process $U = \{U_j\}$ is called *singular* [1, Chap. 2, §5], if $\mathcal{F}(-\infty, t) = \mathcal{F}(-\infty, s)$ for any s and t, where $\mathcal{F}(-\infty, t)$ is the minimal σ-algebra that contains all the events $\{U_{j_1} \in E_1, \ldots, U_{j_n} \in E_n\}$ for any positive integer n, any integers $j_1, \ldots, j_n \in (-\infty, t)$, and any Borel sets $E_1, \ldots, E_n$ of real numbers.

A stationary stochastic process $U = \{U_j\}$ is called *completely singular* [1, Chap. 2, §5] if any stationary stochastic process $U^q = \{U_j^q\}$ with a finite number of values obtained by means of quantization from the process $U = \{U_j\}$ is singular.

Some properties of completely singular processes are pointed out in [4]. Here we only note that the entropy (per unit time) of any singular process with a finite number of values is equal to zero. Therefore, the entropy of any stationary process with a finite number of values obtained by means of quantization from a completely singular process, is equal to zero.

The information rate $\overline{I}(X;Y)$ and the conditional information rate $\overline{I}(X;Y|U)$ in the channel (X, U, Y) described in the introduction are defined as

$$\overline{I}(X;Y) = \lim_{n\to\infty} \frac{1}{n} I(X_1^n; Y_1^n) \tag{1}$$

and

$$\overline{I}(X;Y|U) = \lim_{n\to\infty} \frac{1}{n} I(X_1^n; Y_1^n | U_1^n), \tag{2}$$

provided the limits exist, where $I(\cdot;\cdot)$ and $I(\cdot;\cdot|\cdot)$ are the mutual information and the conditional mutual information, respectively, $X_1^n = (X_1, \ldots, X_n)$, $Y_1^n = (Y_1, \ldots, Y_n)$, and $U_1^n = (U_1, \ldots, U_n)$.

The following theorem contains the main statement on the information rate in stationary channels with a random parameter.

Theorem 1. *Let $X = \{X_j\}$, $U = \{U_j\}$ and $Y = \{Y_j\}$ be arbitrary discrete-time jointly stationary stochastic processes such that X and U are independent. Then the following statements are valid:*

(1) *If $\overline{I}(X;Y)$ and $\overline{I}(X;Y|U)$ are defined, then the inequality*

$$\overline{I}(X;Y) \leq \overline{I}(X;Y|U) \tag{3}$$

holds.

(2) *If $U = \{U_j\}$ is a completely singular process and*

$$I(Y_{-\infty}^{0}; Y_1) < \infty \tag{4}$$

then $\overline{I}(X;Y)$ and $\overline{I}(X;Y|U)$ are defined, $\overline{I}(U;Y) = 0$, and in the case where $\overline{I}(X;Y|U) < \infty$ the equalities

$$\overline{I}(X;Y) = \overline{I}(X;Y|U) \tag{5}$$

and

$$\overline{I}(U;Y|X) = \overline{I}(U;Y) = 0 \tag{6}$$

hold.

For channels with an additive-multiplicative noise the following statement can be derived from this theorem.

Theorem 2. *Let $X = \{X_j\}$, $U = \{U_j\}$ and $Z = \{Z_j\}$ be independent, discrete-time, stationary stochastic processes such that U is a completely singular process and $Y = UX + Z$. Then the following statements are valid:*

(1) *If*

$$I(U_0 X_0; Y_0) < \infty \tag{7}$$

and

$$I(Z_{-\infty}^{0}; Z_1) < \infty \tag{8}$$

then $\overline{I}(X;Y)$ and $\overline{I}(X;Y|U)$ are defined, $\overline{I}(U;Y) = 0$, and in the case where $\overline{I}(X;Y|U) < \infty$ equalities (5) and (6) hold, i.e., the conclusion of the second statement of Theorem 1 is correct.

(2) *If* $X = \{X_j\}$ *and* $Z = \{Z_j\}$ *are stationary sequences of independent random variables and*

$$I(U_0, X_0; Y_0) < \infty \tag{9}$$

then $\overline{I}(X;Y)$, $\overline{I}(X;Y|U)$, $\overline{I}(U;Y)$, *and* $\overline{I}(U;Y|X)$ *are defined and equalities* (6) *and*

$$\overline{I}(X;Y) = \overline{I}(X;Y|U) = I(X_0;Y_0|U_0) \tag{10}$$

hold.

It should be noted that equality (5) may be incorrect if condition (4) is not fulfilled. Indeed, assume $Y = X + U$, where the processes $X = \{X_j\}$ and $U = \{U_j\}$ satisfy the conditions that $X_j \equiv X_0$, $U_j \equiv U_0$ for all integer j (such that both processes X and U are completely singular), X_0 and U_0 are independent and have Gaussian distributions. Then $I(Y_{-\infty}^0; Y_1) = I(X_0 + U_0; X_0 + U_0) = \infty$, i.e., condition (4) is violated. At the same time, we have $\overline{I}(X;Y) = \lim\limits_{n\to\infty} \frac{1}{n} I(X_0; X_0 + U_0) = 0$, and $\overline{I}(X;Y|U) = \lim\limits_{n\to\infty} \frac{1}{n} I(X_0; X_0 + U_0|U_0) = \infty$ which shows that in this case equality (5) is incorrect.

Note also that conditions (4) and (7)–(9) are satisfied in most cases of practical interest. For example, these conditions are fulfilled if all processes considered take a finite number of values or denumerable sets of values with finite entropies.

ASYMPTOTICS

Formulas (5) and (10) allow us to establish explicit expressions for $\overline{I}(X;Y)$ in certain special cases. For example, set $Y = UX + Z$ and assume that X, U, and Z are independent stationary processes such that X and Z are Gaussian with spectral densities $f_X(\lambda)$ and $f_Z(\lambda)$, respectively, and $U_j \equiv U_0$ for any j, where U_0 is a random variable with distribution function $F(u)$. Then the conditions of Theorem 2 are satisfied. Therefore, taking into account the well-known expression for the information rate of Gaussian stationary processes [1, Chap. 3, §10] and formula (5) (see also [5]), we have

$$\overline{I}(X;Y) = \overline{I}(X;Y|U) = \frac{1}{2} \int_{-\infty}^{\infty} dF(u) \int_{-1/2}^{1/2} \ln\left(1 + \frac{u^2 f_X(\lambda)}{f_Z(\lambda)}\right) d\lambda. \tag{11}$$

In particular, if X and Z are stationary Gaussian sequences of independent random variables with parameters (a_X, σ_X^2) and (a_Z, σ_Z^2), respectively, then (11)

turns into the following well-known formula:

$$\overline{I}(X;Y) = \frac{1}{2}\int_{-\infty}^{\infty} \ln\left(1 + \frac{u^2\sigma_X^2}{\sigma_Z^2}\right) dF(u). \tag{12}$$

Moreover, formulas (5) and (10) together with the results of papers [6, 7] (see also [8]) allow us to investigate the asymptotic behavior of $\overline{I}(X;Y)$ in the case of weak input signals for certain models of channels with a random parameter and, in particular, for channels with an additive-multiplicative noise.

To formulate the corresponding results, let us recall that a stationary stochastic process $X = \{X_j\}$ is called *weakly regular* (i.e., a process with completely positive entropy) [1, Chap. 2, §5] if any nontrivial (different from a constant) stationary stochastic process $X^q = \{X_j^q\}$ with a finite number of values obtained by means of quantization from the original process $X = \{X_j\}$ is regular (see [9, Chap. 5, §8]). Some properties of such processes are also pointed out in [4]. Here we only note that the class of weakly regular processes is one of the most extensive classes of processes possessing some property of "regularity". In particular, any regular process is weakly regular, and any nontrivial transformation (not only linear) of a weakly regular process leads again to a weakly regular one. Any Gaussian stationary process having a spectral density is weakly regular.

Before the next theorem will be stated, we recall that the quantity

$$J = J(Z_0) = \int_{-\infty}^{\infty} \frac{[p'(z)]^2}{p(z)}\, dz$$

is called the Fisher information of the random variable Z_0 provided Z_0 has a differentiable density function $p(z)$.

Theorem 3. *Assume $Y = \varepsilon XU + Z$, where $X = \{X_j\}$, $U = \{U_j\}$ and $Z = \{Z_j\}$ are independent stationary processes such that $U = \{U_j\}$ is a completely singular process taking values in a bounded set of the real axis and $Z = \{Z_j\}$ is a sequence of i.i.d. random variables having a finite second moment and a differentiable density function $p(z)$ such that $p(z)$ and $p'(z)$ are bounded and the Fisher information $J(Z_0)$ is finite.*

Assume also that the formula

$$I(\varepsilon X_0; \varepsilon X_0 + Z_0) = \frac{1}{2}J(Z_0)\sigma^2\varepsilon^2 + o(\varepsilon^2), \quad \varepsilon \to 0 \tag{13}$$

holds where $\sigma^2 = var(X_0) < \infty$. *Then for the information rate* $\overline{I}(\varepsilon X; Y)$ *the following asymptotic representation*

$$\overline{I}(\varepsilon X; \varepsilon XU + Z) = \frac{1}{2} J(Z_0)(\mathbf{E}U_0^2)\sigma^2\varepsilon^2 + o(\varepsilon^2), \quad \varepsilon \to 0 \tag{14}$$

is valid if

(i) $X = \{X_j\}$ *is a sequence of i.i.d. random variables*

or

(ii) $X = \{X_j\}$ *is a weakly regular stationary process and* $U_j \equiv U_0$ *for all* $j = 0, \pm 1, \ldots$.

Some conditions under which the asymptotic relation (13) is valid, can be found in [6, 7].

COMPARISON WITH OTHER MODELS

It is interesting to compare the information rates $\overline{I}(X; Y)$, $\overline{I}(X; Y')$ and $\overline{I}(X; Y'')$ in the following models of stationary channels:

$$Y = \varepsilon UX + Z, \tag{15}$$

$$Y' = \varepsilon U'X + Z, \tag{16}$$

and

$$Y'' = \varepsilon mX + Z, \tag{17}$$

where the stationary processes $X = \{X_j\}$, $U = \{U_j\}$, $U' = \{U_j'\}$ and $Z = \{Z_j\}$ are mutually independent, $U = \{U_j\}$ is a completely singular process, $U' = \{U_j'\}$ is a sequence of i.i.d. random variables whose probability distributions coincide with the probability distribution of U_0, and $m^2 = \mathbf{E}U_0^2$.

It seems natural to expect that under rather general conditions, $\overline{I}(X; Y) \geq \overline{I}(X; Y')$ for any ε. We cannot claim that this inequality is always correct. Nevertheless, as can be easily seen, it is valid if $X = \{X_j\}$ and $Z = \{Z_j\}$ are stationary sequences of independent random variables.

Using the results of [6], one can obtain the asymptotic representation for $\overline{I}(\varepsilon X; Y')$ as $\varepsilon \to 0$ in the case where $X = \{X_j\}$ and $Z = \{Z_j\}$ are stationary sequences of independent random variables. Namely, assume that the conditional

density function

$$f(x,y') = p_{Y'|X}(y'|x) = \int_{-\infty}^{\infty} p_U(u) p_Z(y'-ux)\,du$$

satisfies conditions $I'_{(0)} - IV'_{(0)}$ of [6, Theorem 1]. Then, under the assumption that $|X_0| \le K < \infty$ where K is a constant, it follows from this theorem that

$$\overline{I}(X;Y') = I(\varepsilon X_0; \varepsilon U_0 X_0 + Z_0) = \frac{1}{2} J(Z_0)(\mathbf{E}U_0)^2 \sigma_X^2 \varepsilon^2 + o(\varepsilon^2), \quad \varepsilon \to 0. \quad (18)$$

Comparing (14) and (18), we observe that in the case considered $\overline{I}(X;Y) > \overline{I}(X;Y')$ for all sufficiently small ε if $\sigma_U^2 > 0$. Moreover, if $\mathbf{E}U_0 = 0$, then $\overline{I}(X;Y') = o(\varepsilon^2)$, $\varepsilon \to 0$ and, at the same time, $\overline{I}(X;Y)$ has order ε^2.

Comparing models (15) and (17), we note that under rather general assumptions on the processes X, U, and Z, the inequality $\overline{I}(X;Y) \le I(X;Y'')$ holds. For example, it is easy to see that this inequality is always correct if X and Z are stationary sequences of independent Gaussian random variables. At the same time, it is possible to construct an example where $\overline{I}(X;Y) > I(X;Y'')$. To do so, consider the case where $X = \{X_j\}$ and $Z = \{Z_j\}$ are sequences of i.i.d. random variables such that $P\{X_0 = -1\} = P\{X_0 = 1\} = \frac{1}{2}$ and

$$P\{Z_0 = k\} = p_k > 0,\ k = 0, \pm 1, \ldots,\ \sum_k p_k = 1.$$

Assume also that $P\{U_0 = \frac{1}{\sqrt{2}}\} = P\{U_0 = \frac{\sqrt{3}}{\sqrt{2}}\} = \frac{1}{2}$ and $\varepsilon = 1$. Then we have that

$$\overline{I}(X;Y'') < H(X_0) = \ln 2 = \overline{I}(X;Y).$$

Assume now that the processes X, U, and Z satisfy the conditions of Theorem 3. Then

$$\begin{aligned} \overline{I}(\varepsilon X; \varepsilon U X + Z) &= \overline{I}(\varepsilon X; \varepsilon m X + Z) + o(\varepsilon^2) \\ &= \frac{1}{2} J(Z_0) m^2 \sigma^2 \varepsilon^2 + o(\varepsilon^2), \quad \varepsilon \to 0, \end{aligned}$$

i.e., the information rates in the models (15) and (17) are asymptotically equivalent up to terms of order $o(\varepsilon^2)$ if $\varepsilon \to 0$.

As a concluding remark, we mention that the results of this paper concerning

model (15) can be extended to the more general case where the output signal Y is equal to the sum $Y = UX + \alpha U + \beta X + Z$ and where the processes U, X, and Z are the same as in the model (15), and α and β are some constants.

REFERENCES

[1] M. S. Pinsker, *Information and Information Stability of Random Variables and Processes*, Holden-Day, San Francisco (1964).

[2] M. S. Pinsker, V. V. Prelov, and E.C. van der Meulen, "Information transmission over channels with additive-multiplicative noise," *Proc. 1998 IEEE ISIT*, Cambridge, MA, USA, August 16-21, 1998, 239.

[3] M. S. Pinsker, V. V. Prelov, and E. C. van der Meulen, "Stationary channels with a random parameter which is a completely singular process," *Probl. Peredachi Inf.*, **35**, No. 1, 3-12 (1999).

[4] M. S. Pinsker and V. V. Prelov, "On error-free filtering of certain stationary processes," *Usp. Mat. Nauk*, **52**, No. 2, 109–118 (1997).

[5] I. A. Ovseevich, "Capacity and the information rate in channels with a random parameter," *Probl. Peredachi Inf.*, **8**, No. 4, 28–39 (1972).

[6] I. A. Ibragimov and R. Z. Khasminsky, "Weak signal transmission in a memoryless channel," *Probl. Peredachi Inf.*, **8**, No. 4, 28–39 (1972).

[7] M. S. Pinsker, V. V. Prelov, and E. C. van der Meulen, "Information rates in certain stationary non-Gaussian channels in weak-signal transmission," *Probl. Peredachi Inf.*, **34**, No. 1, 3–17 (1998).

[8] M. S. Pinsker, V. V. Prelov, and E.C. van der Meulen, "Information transmission over stationary channels with additive non-Gaussian noise by means of weak input signals", *Proc. 19th Symposium on Information Theory in the Benelux*, Veldhoven, The Netherlands, May 28-29, 1998, 143-147.

[9] I. I. Gikhman and A. V. Skorokhod, *An Introduction to the Theory of Random Processes* [in Russian], Nauka, Moscow (1965).

ON ERROR CORRECTION IN INFORMATION FEEDBACK SCHEMES

M.S. Bargh and J.P.M. Schalkwijk

Eindhoven University of Technology,
Department of Electrical Engineering,
Group on Information and Communication Theory,
PO Box 513, 5600 MB Eindhoven, The Netherlands.

mortaza@ei.ele.tue.nl, piet@ei.ele.tue.nl

Most information feedback schemes yield highly reliable communication by efficiently repeating messages or message chips. In order to gain an insight into such information feedback schemes, this contribution provides a unified explanation and a deeper understanding about the repetition mechanisms in these information feedback schemes for Discrete Memoryless Channels (DMC's) and Additive White Gaussian Noise (AWGN) channels. Our objective is to give a general insight into these repetition mechanisms not to exactly evaluate them.

INTRODUCTION

Repetition of code words or channel input symbols can improve the error performance of one-way coding systems or, in other words, lower the coding complexity for the same performance. For example, consider the situation where every code word of an (M, N) block code with maximum decoding error probability $P_{e,max}^{N}$ is repeated L times. Based on a majority decision rule at the receiver, a decoding error might occur when half or more of the blocks are received erroneously. The decoding error can be upper-bounded by $g_L(P_{e,max}^{N})^{\frac{L}{2}}$, where $g_L = \frac{L}{2}2^L$ and $L \ll N$. This shows an increase in the reliability of the decoded messages as L increases. Meanwhile, the transmission rate falls accordingly since the mentioned repetition mechanism is not efficient.

In information feedback communication systems, the transmitter is able to observe every received symbol via a noiseless and delayless feedback channel and use this information in the coding process. In fact, *most information feedback schemes yield highly reliable communication by efficiently repeating messages or message chips (i.e. message segments).* In such information feedback schemes we can distinguish two efficient repetition mechanisms which prevent any rate

loss as mentioned for the case of one-way communication systems. These two mechanisms are referred to as "repeat to resolve uncertainty" and "repeat to correct erroneous receptions" in this contribution. Generally speaking, an encoder using the first mechanism knows **what** to repeat and that of the latter one knows **when** to repeat. Note that our objective is to obtain a general insight into the repetition mechanisms of such information feedback schemes not to exactly evaluate them.

REPEAT TO RESOLVE UNCERTAINTY

In the schemes using the first mechanism, the transmitter sends the entire message in the initial transmission. Then, from the second transmission onwards, the transmitter sends corrections needed to resolve the *difference* between the true message and the receiver's latest estimate of this message. Note that the transmitted difference is sufficient to completely specify the true message provided that the difference is received correctly. Based on data received previously, the receiver has a *list* (or a range) of candidates for the true message. Every transmitted correction can be interpreted as the index of the true message on receiver's list. As the transmission process continues, the size of the list decreases or, in other words, the correcting data is compressed relative to the previous transmissions. In this way, the scheme saves on transmission energy or, alternatively, on the number of transmissions. In the following, we explain a few schemes for DMCs and AWGN channels that operate based on this principle.

Savage's scheme

Savage [7] presented a block coding strategy which explicitly uses the concept of the list coding. The strategy for transmitting each message is a two-step procedure. For each step a one-way block code is used. In the first step, a block code is used to transmit one of the M messages. The decoder receives a distorted version of the codeword and makes a list of L messages which are the most likely messages given the received code word, i.e. a list decoding of the first block code is performed. Then the feedback channel is used to inform the transmitter about the list and the order in which messages appear on the list. In the second step, i.e. the first repetition of the message, the transmitter chooses a code word from the second code to indicate which message on the list of the L messages is the true message. Notice the efficient repetition mechanism employed in the second

step where one out of L entries on the list is transmitted instead of retransmitting one out of M messages, $L \ll M$.

Ooi's scheme

In [6], James Ooi used Ahlswede's idea employed in a constructive proof of the coding theorem for DMC's with feedback [1], and introduced the so called compressed-error-cancellation framework for transmitting information over DMC's with feedback. The very basic principle of Ooi's scheme was presented as an information feedback scheme for Binary Symmetric Channels (BSC's) in a Bell Labs memorandum dated back in 1962 [14] by Weldon. Ooi's scheme is a low complexity capacity achieving feedback scheme for DMC's though it gives a good performance for discrete finite state channels, discrete memoryless multiple access channels and for channels that vary with time in an unknown way over a set of possible channels, see [5].

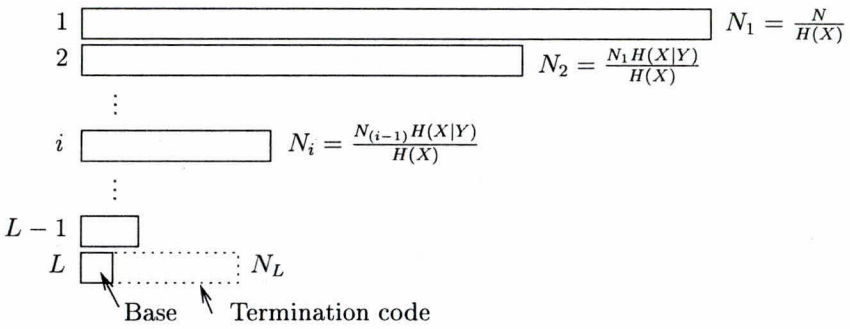
Assume a DMC with capacity achieving input distribution q_X. Message $w \in \{0,1\}^N$ is mapped into $N_1 = \frac{N}{H(X)}$ independent and identically distributed (iid) channel input symbols according to distribution q_X. The resulting block, i.e. the whole message, is transmitted through the channel. The channel distorts the message and the transmitter observes the received block via the feedback channel. For the first (efficient) repetition of the message, the transmitter compresses the distortion into $N_1 H(X|Y)$ bits and then maps the result into $N_2 = \frac{N_1 H(X|Y)}{H(X)}$ iid channel inputs and transmits the resulting message through the channel. In the ith iteration (or the $(i-1)^{\text{st}}$ effective repetition of the message), the distortion of the previous block is compressed to $N_{(i-1)}H(X|Y)$ bits, then the result is mapped into $N_i = \frac{N_{(i-1)}H(X|Y)}{H(X)}$ iid channel inputs to be transmitted through the channel. When the process continuous indefinitely, the number of transmitted symbols is

$$\frac{N}{H(X)} + \frac{N_1 H(X|Y)}{H(X)} + \cdots + \frac{N_{(i-1)}H(X|Y)}{H(X)} + \cdots =$$
$$\frac{N}{H(X)}\left(1 + \frac{H(X|Y)}{H(X)} + \cdots + \left(\frac{H(X|Y)}{H(X)}\right)^{(i-1)} + \cdots\right) = \frac{N}{H(X) - H(X|Y)}$$

resulting in an average rate equal to the channel capacity. The length of the transmitted blocks decreases as the process proceeds, see the figure. In practice, the iteration stops after a fixed number of times. The last block or, in other words, the base block which contains a small number of symbols is reliably sent

over the channel using a strong termination (feedback) code. In addition to the information about the channel noise, every block contains some other information about the previous block, e.g. the length of the previous block. Before the termination block, a synchronization sequence is sent to mark the beginning of the termination code. This framework shows the problem of channel coding with feedback to be strongly related to the problem of source coding for which there are both rich theory and efficient lossless source coding algorithms.

For decoding, the decoder of the scheme looks for the synchronization sequence in order to find the termination code, decodes the last block and uses the decoded information to decode the block before the synchronization sequence. In this way, the decoded information of the ith iteration is used to resolve the uncertainty in the $(i-1)^{\text{th}}$ iteration. This process continues until the uncertainty in the first block, corresponding to the original message, is resolved.



Center-of-gravity methods

In [8], Schalkwijk gave the center-of-gravity information feedback model as an unifying principle for many information feedback schemes designed for AWGN channels. In these channels, transmitted signals are constrained in average power and sometimes in peak power. Every message $w \in \mathcal{W}$ corresponds to a D-dimensional vector $\mathbf{x}_w^D$ as the input of a vector channel. The center-of-gravity method is a repetition type scheme, however, it does not simply repeat the input signal N times. Instead, before the nth transmission, $n = 1, \ldots, N$, a common signal $\mathbf{u}_{(n-1)} \in I\!R^D$ is computed at the receiver from already received signals and sent back to the transmitter. On the nth transmission, or the $(n-1)^{\text{st}}$ repetition of the message, $\mathbf{u}_{(n-1)}$ is subtracted at the transmitter and added again at the

receiver. Therefore, the transmitted vector is

$$\mathbf{x}^N = \left((\mathbf{x}^w - \mathbf{u}_0), (\mathbf{x}^w - \mathbf{u}_1), \ldots, (\mathbf{x}^w - \mathbf{u}_{(N-1)})\right).$$

The optimum sequence $\mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_{(N-1)}$ minimizes the average transmission power. From $(n-1)$ received signals $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{(n-1)}$, the receiver can compute the center of gravity $\mathbf{u}_{(n-1)}$ that minimizes the moment of inertia of the signal set around the tip of the vector $\mathbf{u}_{(n-1)}$. In these methods, the center of gravity approaches the true message point in the D dimensional signal space and, therefore, the energy of transmissions decreases as the process proceeds. On the other hand, subtracting and then again adding common signal vectors $\mathbf{u}_{(n-1)}$, does not affect the probability of error if we compare it to the scheme that repeats the original message N times.

A suboptimum version of center-of-gravity uses the *receiver's best* guess in the place of the center of gravity. The best $(n-1)^{\text{st}}$ guess is the message point $w \in \mathcal{W}$ in the D dimensional signal space, which maximizes $p_{\mathbf{X}|\mathbf{Y}^{(n-1)}}(\mathbf{x}^w|\mathbf{y}_1, \ldots, \mathbf{y}_{(n-1)})$. The best guess is inferior to the center-of-gravity, however, feeding back one out of $|\mathcal{W}|$ possible signals needs less powerful feedback channel than that needed in the center-of-gravity method. Many feedback schemes for AWGN channels can be explained by the center-of-gravity principle, see [8, 9] and references therein for an in depth study.

REPEAT TO CORRECT ERRONEOUS RECEPTIONS

The information feedback schemes using the second mechanism repeat messages or message chips whenever necessary. If a message (chip) is received correctly, the transmitter moves on to the next message (chip). Otherwise, the entire message (chip) is repeated. From this point of view, these schemes are similar to the automatic repeat request schemes. However, here the transmitter side which observes the received message (chip) via the feedback channel initiates the acknowledgement or rejection decision. Therefore, there is a need for a mechanism to convey the acknowledgement or rejection status to the receiver for every message (chip). In the schemes using this mechanism, the rejected message is totally neglected and it is not possible to compress the correcting repetitions or to optimize their transmission power (in fact, sometimes there is an expansion of transmissions in order to correct an erroneous reception). The optimum performance in these schemes is gained due to immediately repeating the erroneous

receptions. In the following we explain a few information feedback schemes that use this mechanism to correct transmission errors.

Schalkwijk-Barron's scheme

This is a sequential signalling over AWGN channels subject to a peak power constraint [11]. The signalling has two modes: the message mode and the control mode. In the message mode, one of M messages is sent through the channel using a set of M orthogonal signals. Then the receiver sends back the decoded message or the transmitter deduces it by observing the received signal. In the control mode, the transmitter sends the acknowledgement or the rejection signal depending on whether the decoder is correct or not, respectively. The receiver decodes the control part and if the decoded result is an acknowledgement, the receiver accepts the decoded message as correct. If the control part is decoded as a rejection, the receiver discards the decoded message and waits for a retransmission. For the control part, Schalkwijk and Barron used the sequential decision feedback scheme of Viterbi for AWGN channels [13].

Yamamoto-Itoh's scheme for AWGN channels [15] is a modification of Schalkwijk-Barron's scheme. In the control mode of Schalkwijk-Barron's scheme, Yamamoto and Itoh employed a new blockwise decision scheme instead of Viterbi's sequential decision scheme. In this way, a totally fixed length transmission method is obtained with the same asymptotic error exponent as that of the original scheme. Yamamoto and Itoh also generalized Schalkwijk-Barron's scheme for use in DMC's. In a block of N transmissions, a block code of length γN is used to send the message part, $\gamma < 1$. They used the rest of the block for the control part to convey the acknowledgement or the rejection signals to the receiver using two codewords $\mathbf{x}_1^{(1-\gamma)N} = (x, x, \ldots, x)$ and $\mathbf{x}_1^{(1-\gamma)N} = (x', x', \ldots, x')$, respectively, where x and x' are two input symbols of the channel.

Multiple Repetition Feedback Coding (MRFC) schemes

To explain MRFC schemes [12, 10] in the framework of the repeat to correct error mechanism, we assume that every message consists of message chips to be transmitted to the receiver. Every message chip corresponds to a channel input symbol. Suppose that the transmitter sends symbol x over the DMC. Upon correct reception of the symbol, the next symbol is transmitted. However, if x is erroneously received as $y \neq x$, then x is retransmitted k_{xy} times. In the decoder,

a received subsequence $yx^{k_{xy}}$, i.e. a y followed by k_{xy} x's, indicates the transmission error $x \rightarrow y$. The error is subsequently corrected by replacing subsequence $yx^{k_{xy}}$ with x to retrieve the corresponding message chip. In order to prevent removing message chips, the sequence of message chips must be constrained or precoded such that the corresponding channel inputs do not constitute the so called forbidden subsequences $yx^{k_{xy}}$, $x \neq y$. In MRFC schemes there is no explicit acknowledgement signal but the rejection signal, which does not occur so often, is a few repetitions of the symbol in error. In fact, the acknowledgement signals are hidden in the sequence of the message chips, where a correctly received message chip, i.e. its corresponding channel input symbol, can not be the beginning of a forbidden subsequence. This is guaranteed by the precoding and repetition rules.

Horstein's scheme [4] is a sequential scheme to transmit a message through a BSC with noiseless feedback. The transmission of message chips is very similar to that of MRFC schemes. Interesting to know is that Schalkwijk derived MRFC schemes for BSCs by detailed study of Horstein's scheme. Zigangirov and Burnashev used Horstein's idea in [18, 16, 17] and in [2, 3], respectively, where they modified Horstein's schemes and improved some shortcomings of Horstein's presentation.

REFERENCES

[1] Rudolf Ahlswede. A constructive proof of the coding theorem for discrete memoryless channels with feedback. In *proceeding of the sixth Prague conference on information theory, statistical decision functions and random processes*, pages 39–59, 1971.

[2] M.V. Burnashev. Data transmission over a discrete channel with feedback. random transmission time. *Problemy Peredachi Informatsii*, 12(4):10–30, 1976.

[3] M.V. Burnashev. Reliability function of a binary symmetrical channel with feedback. *Problemy Peredachi Informatsii*, 24(1):3–10, 1988.

[4] Michael Horstein. Sequential transmission using noiseless feedback. *IEEE Transactions on Information Theory*, 9:136–143, July 1963.

[5] James M. Ooi. *A framework for low-complexity communication over channels with feedback.* PhD thesis, Massachusetts Institute of Technology, 1997.

[6] James M. Ooi and Gregory W. Wornell. Fast iterative coding for feedback channels. In *In the proceeding of IEEE International Symposium on Information Theory, Ulm, Germany*, page 133, July 1997.

[7] J.E. Savage. A bound on the reliability of block coding with feedback. *The Bell System Technical Journal*, 45:967–977, 1966.

[8] J. Pieter M. Schalkwijk. Center-of-gravity information feedback. *IEEE Transactions on Information Theory*, 14(2):324–331, March 1968.

[9] J. Pieter M. Schalkwijk. Recent developments in feedback communication. *Proceedings of the IEEE*, 57(7):1242–1249, July 1969.

[10] J. Pieter M. Schalkwijk. A class of simple and optimal strategies for block coding on the binary symmetric channel with noiseless feedback. *IEEE Transactions on Information Theory*, 17(3):283–287, May 1971.

[11] J. Pieter M. Schalkwijk and Mark E. Barron. Sequential signaling under a peak power constraint. *IEEE Transactions on Information Theory*, 17(3):278–282, May 1971.

[12] Thijs Veugen. *Multiple repetition coding for channels with feedback.* PhD thesis, Eindhoven University of Technology, 1997.

[13] A.J. Viterbi. The effect of sequential decision feedback on communication over the gaussian channel. *Information and control*, 8:80–92, February 1965.

[14] Edward J. Weldon, Jr. An upper bound on the probability of a word error for block codes used with the memoryless binary symmetric channel with noiseless information feedback. Memorandum No. 38763-15, Bell telephone laboratories, August 1962.

[15] Hirosuke Yamamoto and Kohji Itoh. Asymptotic performance of a modified schalkwijk-barron scheme for channels with noiseless feedback. *IEEE Transactions on Information Theory*, 25(6):729–733, November 1979.

[16] K.Sh. Zigangirov. Information transmission over a binary symmetrical channel with noiseless feedback (random transmission time). *Problemy Peredachi Informatsii*, 4(3):38–47, 1968.

[17] K.Sh. Zigangirov. Upper bounds for the error probability for channels with feedback. *Problemy Peredachi Informatsii*, 6(2):87–92, 1970.

[18] K.Sh. Zigangirov. Recurrent transmission through the binary symmetric channel with feedback. *Problems of Control and Information Theory*, 6(4):189–205, 1977. English translation at pages 12–25.

MORE ON THE DUALITY BETWEEN SOURCE AND CHANNEL CODING

Valery N. Koshelev* and **Edward C. van der Meulen†**

The duality between source and channel coding as pointed out by Shannon [2] is further explored. We consider both multi-level source coding subject to a sequence of fidelity constraints and multi-level channel coding subject to a sequence of input cost constraints. Efficient multi-level source coding and efficient multi-level channel coding can both be phrased in terms of the concept of divisibility. Whereas earlier necessary and sufficient conditions were found for source divisibility, sofar only sufficient conditions have been established for channel divisibility.

INTRODUCTION

There exist well-known remarks by A. N. Kolmogorov and V. M. Tikhomirov [1] and C.E. Shannon [2] on a certain duality between source and channel coding. Mathematically, these reflect the duality between two main information-theoretical quantities known as the rate-distortion function

$$R(\varepsilon) \stackrel{\Delta}{=} \min_{Y:D(X;Y)\leq\varepsilon} I(X;Y) \tag{1}$$

and the capacity-cost function

$$C(\tau) \stackrel{\Delta}{=} \max_{X:T(X)\leq\tau} I(X;Y). \tag{2}$$

This work was supported in part by INTAS under Grant No. 94-469.

* Scientific Council for Cybernetics of the RAS, Moscow, Russia.

† Department of Mathematics, Katholieke Universiteit Leuven, Leuven, Belgium.

Here $(X, Y) = \{\mathcal{X} \times \mathcal{Y}; \mathbf{P} \circ \mathbf{Q}\}$, $\mathbf{P} = [p(x)], \mathbf{Q} = [q(y|x)]$, is a pair of random variables defined over the alphabets $\mathcal{X}$ and $\mathcal{Y}$, and I is the average mutual information of the pair; in (1)

$$D(X;Y) \triangleq \sum_{xy} p(x)q(y|x)d(x;y) \tag{3}$$

is the average distortion at which the source output is reproduced by a reproducer output Y; in (2)

$$T(X) \triangleq \sum_{x} p(x)t(x) \tag{4}$$

is the average cost of the channel input X; the values (3) and (4) are calculated with respect to a distortion $|\mathcal{X}| \times |\mathcal{Y}|$-matrix

$$\mathbf{d} = [d(x;y)] \tag{5}$$

and a cost $|\mathcal{X}|$-vector

$$\mathbf{t} = [t(x)], \tag{6}$$

respectively.

For the discrete memoryless source (DMS) defined by the pair

$$(\mathbf{P}; \mathbf{d}) \tag{7}$$

the function $R(\varepsilon)$ shows the minimal amount of information sufficient to reproduce the source output with an average distortion less than or equal to ε. For the discrete memoryless channel (DMC) given by the pair

$$(\mathbf{Q}; \mathbf{t}) \tag{8}$$

the function $C(\tau)$ shows the maximal amount of information one can reliably transmit over the channel paying a cost of no more than τ per channel input.

SUCCESSIVE SOURCE CODING

The coding theorems showing that values (1) and (2) can really be achieved were proved in 1959 by Shannon [2]. Since then the theorems have been applied intensively to various schemes in the field of source and channel coding.Particularly, in a series of works starting in 1978, a scheme of hierachical source coding was formulated and studied [3, 4, 5, 6, 7, 9, 10]. It was shown that source coding

with a given distortion measure can be realized as a kind of scheme of successive refinement and that under definite conditions the rate-distortion function can be achieved simultaneously for a sequence of distortions

$$0 \leq \varepsilon_L \leq \varepsilon_{L-1} \leq \ldots \leq \varepsilon_1, \quad L \geq 2. \tag{9}$$

The successive coding procedure is based on a hierarchical code which is an L-level tree whose nodes are source reproducer code words of a block length N. In this tree, each mother node generates daughter nodes with exponential rates of branching equal to $R_l, l \in \overline{1,L}$. Then the theorem of successive source coding shows the exchange between the L-vector of rates of branching and the L-vector of distortions. Let $L = 2$ and consider a stochastic scheme

$$(X, Y_1, Y_2) \in \mathcal{X} \times \mathcal{Y}^2 \tag{10}$$

where X is a generic source letter and Y_l is its copy of the l-th refinement, $l \in \overline{1,2}$.

Theorem 1 (Two-level successive source coding). Let a DMS be defined by the pair (7) where $\mathbf{P} = \mathbf{P}_X$. Then the point $(R_1, R_2, \varepsilon_1, \varepsilon_2)$, $\varepsilon_2 \leq \varepsilon_1$, is achievable in the scheme of successive coding iff there exists a triplet (10) such that

$$I(X;Y_1) \leq R_1, \qquad I(X;Y_2|Y_1) \leq R_2$$

$$D(X_1;Y_1) \leq \varepsilon_1, \qquad D(X_2;Y_2) \leq \varepsilon_2.$$

The proof of Theorem 1 follows a standard technique of random source coding [8] modified with respect to a multi-level mode [4].

Theorem 1 can easily be generalized to the case

$$(R_1, \ldots, R_L, \varepsilon_1, \ldots, \varepsilon_L), \qquad L \geq 3. \tag{11}$$

The scheme of successive source coding achieves its maximal efficiency with

$$R_l = R(\varepsilon_l) - R(\varepsilon_{l-1}), \forall l \in \overline{1,L}. \tag{12}$$

Definition 1. A DMS (7) with rate-distortion function (1) is called *divisible* for a sequence of distortions (9) iff the point (11) is achievable for R_l, $l \in \overline{1, L}$, defined in (12).

In other words, if the source is divisible, then the optimal scheme of successive source coding can be viewed as a step-by-step climbing up along the rate-distortion curve.

Let

$$\mathbf{Q}_\varepsilon \triangleq \arg \min_{\mathbf{Q}:\mathcal{D}(\mathbf{P};\mathbf{Q};\mathbf{d})\leq\varepsilon} \mathcal{I}(\mathbf{P};\mathbf{Q})$$

be the test-channel of the source $(\mathbf{P};\mathbf{d})$, i.e. the stochastic $|\mathcal{X}| \times |\mathcal{Y}|$-matrix that achieves $R(\varepsilon)$; here $\mathcal{I}$ and $\mathcal{D}$ are the same as I and D in (1) and (3), but in the notation of $\mathcal{I}$ and $\mathcal{D}$ their dependence on $\mathbf{P}, \mathbf{Q}$ and $\mathbf{d}$ is explicitly shown.

Theorem 2 (Source divisibility criterion). The DMS (7) is divisible for a pair of distortions $0 \leq \varepsilon_2 \leq \varepsilon_1$ iff the matrix equation

$$\mathbf{Q}_{\varepsilon_2}\mathbf{S} = \mathbf{Q}_{\varepsilon_1} \tag{13}$$

has a stochastic solution.

Thus, for the class of discrete memoryless sources (7) the problem of source divisibility can be considered as completely solved, because it is reduced to determining whether the equation (13) has a stochastic solution or not.

SUCCESSIVE CHANNEL CODING

Now we turn to a similar problem for the class of memoryless input-constrained channels. Namely, for a DMC (8) with capacity-cost function (2) we wish to construct a scheme of successive channel coding which would look like a climbing up along the curve of $C(\tau)$. It is well-known that the latter is an increasing, concave, and continuous function in its region of interest.

Successive channel coding, as we have developed it, is also based on an L-level code tree with nodes as channel-input code words. Each mother node X_1^N generates a number of daughter nodes $\left\{X_2^N\right\}$. The process of mother-to-daughter transformation $X_1^N \to X_2^N$ is paid at the cost $r\left(X_1^N; X_2^N\right) = \sum\limits_{n=1}^{N} r(x_{1n}; x_{2n})$ defined by a special letter-to-letter cost matrix

$$\mathbf{r} = [r(x_1; x_2)] \tag{14}$$

of the size $|\mathcal{X}| \times |\mathcal{X}|$. To make cost matrix (14) compatible with the original cost vector (6) we let $r(0;x) \equiv t(x)\ \forall x \in \{0,1,\ldots,|\mathcal{X}|\} \triangleq \mathcal{X}$. Thus the dummy code word 0^N plays the role of a root node of the code tree. Let R_l be the rates of the exponential branching of the tree, $l \in \overline{1,L}$. Then the scheme of hierarchical channel coding can be interpreted as a communication system assigned to serve a society of L users, where the l-th user wishes to communicate at the rate $R_1 + \ldots + R_l$. The specific rule of this society is that the basic rate $R_1 + \ldots + R_{l-1}$ is paid for by the $l-1$ preceding users at the cost τ_{l-1} so that the l-th user is to pay only for the additional rate R_l at the cost $\tau_l - \tau_{l-1}$. Again, let $L = 2$. In this case the scheme of the two-level successive channel coding can be described by the four-tuple of random variables

$$\begin{array}{lcl} \mathbf{P} & X_1 & \xrightarrow{\mathbf{Q}} Y_1 \\ \mathbf{W} & \downarrow & \\ & X_2 & \xrightarrow{\mathbf{Q}} Y_2, \end{array} \tag{15}$$

defined by the stochastic triplet $(\mathbf{P}, \mathbf{W}, \mathbf{Q})$.

In (15), X_1 denotes the mother letter distributed according to the stochastic $|\mathcal{X}|$-vector $\mathbf{P}$, X_2 denotes the daughter input letter distributed conditionally on X_1 according to a stochastic $|\mathcal{X}| \times |\mathcal{X}|$-matrix $\mathbf{W} = [w(x_2|x_1)]$, and Y_1, Y_2 are the channel output letters. Y_1 and Y_2 are conditionally independent with the same channel matrix $\mathbf{Q}$. In particular, the triplet

$$X_1 \to X_2 \to Y_2$$

forms a Markov chain. Then

$$T(X) \triangleq \sum_x p(x)\, t(x) \equiv \sum_x p(x)\, r(0;x)$$

denotes the average input cost paid by the first user and

$$S(X_1;X_2) \triangleq \sum_{x_1 x_2} p(x_1)\, w(x_2|x_1)\, r(x_1;x_2) \tag{16}$$

is a super cost paid by the second user. With these notations we prove

Theorem 3 (Two-level successive channel coding). Let a DMC be given by the pair

$$(\mathbf{Q};\mathbf{r})\,. \tag{17}$$

For this channel, the point

$$(R_1, R_2; \tau_1, \tau_2) \tag{18}$$

is achievable with a scheme of two-level successive coding iff there exists a stochastic scheme (15) such that

$$T(X_1) \le \tau_1 \tag{19}$$

$$S(X_1; X_2) \le \tau_2 - \tau_1 \tag{20}$$

$$I(X_1; Y_1) \ge R_1 \tag{21}$$

$$I(X_2; Y_2) \ge R_1 + R_2 \tag{22}$$

$$I(X_2; Y_2|X_1) \ge R_2. \tag{23}$$

Theorem 3 can easily be extended to the general case

$$(R_1, \ldots, R_L, \tau_1, \ldots, \tau_L)\,, \quad L \ge 3. \tag{24}$$

Definition 2. Let the pair (17) define a DMC with a capacity-cost function (2). We say that the channel is *divisible* for a sequence of costs $0 \le \tau_1 \le \tau_2 \le \ldots \le \tau_L$ iff in the scheme of the L-level successive channel coding the point (24) is achievable with $R_l = C(\tau_l) - C(\tau_{l-1}) \quad \forall l \in \overline{1, L}$.

Let

$$\mathbf{P}_\tau \triangleq \arg \max_{\mathbf{P}:\mathcal{T}(\mathbf{P};\mathbf{t})\le\tau} \mathcal{I}(\mathbf{P};\mathbf{Q})$$

be the stochastic $|\mathcal{X}|$ -vector that achieves $C(\tau)$; here $\mathcal{I}$ and $\mathcal{T}$ are the same as I and T in (2) and (4), except that in $\mathcal{I}$ the dependence on $\mathbf{P}$ and $\mathbf{Q}$, and in $\mathcal{T}$ the dependence on $\mathbf{P}$ and $\mathbf{t}$ is exhibited. Denote also (16) as $S(X_1; X_2) \equiv \mathcal{S}(\mathbf{P};\mathbf{W};\mathbf{r})$. Then we prove

Theorem 4 (Sufficient conditions of channel divisibility). The DMC (17) is divisible for a pair of costs $0 \le \tau_1 \le \tau_2$ if there exists a stochastic $|\mathcal{X}| \times |\mathcal{X}|$-matrix $\mathbf{W}$ such that

$$\mathbf{P}_{\tau_1}\mathbf{W} = \mathbf{P}_{\tau_2} \tag{25}$$

$$\mathcal{S}(\mathbf{P}_{\tau_1};\mathbf{W};\mathbf{r}) \le \tau_2 - \tau_1 \tag{26}$$

$$\mathcal{I}(\mathbf{P}_{\tau_1}; \mathbf{Q}) \geq \mathcal{I}(\mathbf{P}_{\tau_1}; \mathbf{WQ}). \tag{27}$$

The proof of Theorem 4 follows from putting $R_1 = C(\tau_1)$ and $R_2 = C(\tau_2) - C(\tau_1)$ in the right-hand sides of (21)-(23).

We conjecture that equation (27) is true for any stochastic $|\mathcal{X}| \times |\mathcal{X}|$-matrix $\mathbf{W}$. If so, then equations (25)-(26) form a system of conditions necessary and sufficient for channel divisibility.

REFERENCES

[1] A.N. Kolmogorov and V.M. Tikhomirov, "Epsilon-entropy and epsilon-capacity of sets in functional spaces" , *Usp. Mat. Nauk*, 14, No. 2, 3-86, 1959. See also *Amer. Math. Soc. Trans.*, 33, 277-367.

[2] C.E. Shannon, "Coding for a discrete source with a fidelity criterion" , *IRE National Conv. Rec.*, pt. 4, 142-163, Mar. 1959. Reprinted in "Key Papers in the Development of Information Theory" (ed. D. Slepian), IEEE Press, 1974, 245-266.

[3] V.N. Koshelev, "Multi-level source coding and data transmission theorem", in : *Proc. 7th All-Union Conference on Coding Theory and Data Transmission* [in Russian], Vilnius, Part 1, 85-92, 1978.

[4] V.N. Koshelev, "Hierarchical source coding", *Probl. Peredachi Inf.*, 16, No. 3, 31-49, 1980.

[5] V.N. Koshelev, "An evaluation of the average distortion for a discrete scheme of sequential approximations", *Probl. Peredachi Inf.*, 17, No. 3, 20-33, 1981.

[6] W.H.R. Equitz and T.M. Cover, "Successive refinement of information", *IEEE Trans. Inform. Theory*, 37, No. 2, 268-275, Mar. 1991.

[7] V.N. Koshelev, "On the divisibility of discrete sources with an additive single-letter distortion measure", *Probl. Peredachi Inf.*, 30, No. 1, 31-50, 1994.

[8] J.K. Omura, "A coding theorem for discrete time sources", *IEEE Trans. Inform. Theory*, 19, No. 4, 490-497, July 1973.

[9] V.N. Koshelev and E.C. van der Meulen, "Two-level coding for input-constrained channels", *Proc. 1998 IEEE ISIT*, Cambridge, MA, USA, August 16-21, 1998, 430.

[10] J. Chow and T. Berger, "Failure of successive refinement for symmetric Gaussian mixtures", *IEEE Trans. Inform. Theory*, 43, No.1, 350-352, Jan. 1997.

AN EFFICIENT ALGORITHM FOR SELECTING OPTIMAL CONFIGURATIONS OF AR-COEFFICIENTS

Rudy Moddemeijer

University of Groningen, Department of Computing Science,
P.O. Box 800, NL-9700 AV Groningen, The Netherlands,
rudy@cs.rug.nl

There exists an essential difference between the correct Auto Regressive (AR) model and the optimal AR-model. We try to find an optimal model balancing between flexibility, using many AR-parameters, and low variance, using only a few AR-parameters. We select an optimal AR-parameter configuration consisting of zero and non-zero parameters given a maximum AR-order. This optimal configuration will be selected using a Modified Information Criterion (MIC) which is closely related to Akaike's criterion (AIC). This MIC allows an a priori selection of the probability of estimating too many parameters.

We present the method and a verification by simulations. The method is based on pivoting the Hessian matrix by Gauß-Jordan pivots. As a result we can now select an optimal parameter configuration with an a priori probability of selecting a configuration with a too large number of parameters given an a priori selected maximum AR-order.

INTRODUCTION

With respect to Auto-Regressive (AR) modeling we distinguish the *correct* model, which generated the signal, and the *optimal* model. It can be a better strategy to estimate the *optimal* model, which has a simpler structure and can more reliably be estimated. Due to the finite sample size not all aspects of the correct model can be estimated with a sufficient accuracy, so we balance between a minimum estimation error (variance) and a minimum modeling error (bias).

Assume an unbiased statistic s, $\mathrm{BIAS}\{\underline{s}\} = 0$, so the mean square error is $\mathrm{MSE}\{\underline{s}\} = \mathrm{VAR}\{s\}$. Can we construct a better statistic? In mean square sense it can be done by $s' = \lambda s$. We search for the minimum mean square error as a function of λ and find:

$$\begin{aligned} \mathrm{BIAS}\{\underline{s'}\} &= (\lambda - 1)\mathrm{E}\{\underline{s}\} \\ \mathrm{MSE}\{\underline{s'}\} &= (\lambda - 1)^2\mathrm{E}\{\underline{s}\} + \lambda^2\mathrm{VAR}\{\underline{s}\} \end{aligned} \quad \text{where } \lambda = \frac{\mathrm{E}\{\underline{s}\}^2}{\mathrm{E}\{\underline{s}\}^2 + \mathrm{VAR}\{\underline{s}\}} \qquad (1)$$

This shows the principle of balancing bias and variance. The decrease of the mean square error by the reduction of the variance dominates the increase by the bias. This means that we can construct better statistics by accepting some bias.

Assume the correct AR-model has a small parameter. The value of this parameter can be negligible with respect to the variance; in this case its better to omit the parameter in the optimal model. This introduces some bias but also reduces the variance. We try to find a balance between flexibility, using many parameters, and low variance, using only a few parameters. We use the large parameter space of a Jth order AR-model with J independently adjustable AR-parameters. Of every parameter we decide during the estimation process whether the parameter is essential or not. We only estimate the essential parameters and keep the other parameters equal to zero.

Neither the AR-order nor the number of negligible parameters are a priori known. Therefore we search for an efficient strategy to estimate the configuration of significant parameters. Our algorithm is based on an Auto Regressive and Moving Average (ARMA) parameter estimation algorithm [1] which was used to find alternative ARMA-parameter configurations providing a better fit to the time-series published in the book of Box & Jenkins [2]. This algorithm has been combined with the AR-order estimation algorithm based on the *Modified Information Criterion* (MIC) [3], which is capable of estimating an AR-order given an a priori selected probability of selecting a too high order.

THEORY

Assume the discrete time stochastic signal $\underline{x}$, generated by AR-filtered, stationary, normally distributed white noise with zero mean and fixed variance σ^2 and a sequence of N observations $x_1, x_2, \ldots, x_N$. Estimate a model of $\underline{x}$; i.e. estimate the configuration and the values of the non-zero coefficients of the AR-model. Assume the Jth order AR-model and its conditional probability density function:

$$f_x(x_n|x_{n-1}, \ldots, x_{n-J}; \mathbf{a}, \sigma) = f_\varepsilon(\varepsilon_n(\mathbf{a}); \sigma) \text{ where } \varepsilon_n(\mathbf{a}) = x_n - \sum_{j=1}^{J} a_j x_{n-j} \quad (2)$$

and where $\mathbf{a} = (a_1, \ldots, a_J)$. This AR-model has a parameter vector $\mathbf{p} = (a_1, a_2, \ldots, a_J, \sigma)$ of $J+1$ independently adjustable parameters. If we fixate some AR-parameters at zero, the number of independently adjustable parameters will be less. The models with different configurations of AR-coefficients are called the hypotheses. The dimension $\dim \mathbf{p} \leq J+1$ for a given configuration is the number of independently adjustable parameters, i.e. the number of non-zero parameters. Given a sequence of observations we determine the parameter configuration by selecting the *optimal* hypothesis.

Testing two composite hypotheses is elaborately discussed in our earlier publication [3]. Given the *Maximum of the Average Log Likelihood* (MALL), the

Average Log Likelihood (ALL) function at the Maximum Likelihood (ML) parameter estimate $\widehat{\mathbf{p}}$, defined by

$$\widehat{\mathrm{E}}\{\underline{l}(\widehat{\mathbf{p}})\} = \tfrac{1}{N}\sum_{n=1}^{N} \log f_x(x_n|\widehat{\mathbf{p}}) \tag{3}$$

where $\widehat{\mathrm{E}}\{\underline{l}(\widehat{\mathbf{p}})\}$ denotes the average of the stochastic variable $\underline{l}(\widehat{\mathbf{p}}) = \log f(\underline{x}|\widehat{\mathbf{p}})$. The quantity $N\ \widehat{\mathrm{E}}\{\underline{l}(\widehat{\mathbf{p}})\}$ equals the well-known maximum of the log likelihood function in the method of ML. In earlier work [3] we have argued that the problem of AR-order estimation can be solved by subsequently testing pairs of hypotheses using the difference in ALL at the ML-estimates as a criterion:

$$\widehat{\mathrm{E}}\{\triangle\underline{l}(\widehat{\mathbf{p}}_0, \widehat{\mathbf{p}}_1)\} = \widehat{\mathrm{E}}\{\underline{l}(\widehat{\mathbf{p}}_1)\} - \widehat{\mathrm{E}}\{\underline{l}(\widehat{\mathbf{p}}_0)\} > \eta_{high} \tag{4}$$

We accept the alternative hypothesis with parameter vector $\widehat{\mathbf{p}}_1$ if (4) is fulfilled. The constant η_{high} depends on $(\dim \widehat{\mathbf{p}}_1 - \dim \widehat{\mathbf{p}}_0)$ and the a priori selected *False Alarm Probability* (FAP) α; i.e. the probability on selecting a too high order.

Within our concept we have the same null hypotheses but a set of alternative hypotheses instead of only one alternative hypothesis. We want to extend the successful concept of our earlier work to this situation.

MODEL SELECTION BY GAUSS-JORDAN PIVOTS

A Gauß-Jordan pivot is an operation on a matrix V resulting in the pivoted matrix $\tilde{V}$. Performing a Gauß-Jordan pivot with pivot element V_{ij}:

$$\begin{aligned}
\tilde{V}_{mn} &= V_{mn} - V_{mj}V_{in}/V_{ij} && \text{where } m \neq i \text{ and } n \neq j \\
\tilde{V}_{mi} &= V_{mj}/V_{ij} && \text{where } n \neq j \\
\tilde{V}_{in} &= V_{in}/V_{ij} && \text{where } m \neq i \\
\tilde{V}_{ij} &= 1/V_{ij}
\end{aligned} \tag{5}$$

exchanges the elements a_i and b_j in the matrix equation $\mathbf{a} = V\mathbf{b}$. Performing Gauß-Jordan pivot using all main diagonal elements as pivot element is equivalent to inversion of the matrix. Using a selected subset of main diagonal element leads to a *partial inverse.*

Due to the Gaussian noise we determine the ALL-function given the parameter vector $\mathbf{p}$ by:

$$\widehat{\mathrm{E}}\{\underline{l}(\mathbf{p})\} = \tfrac{1}{N}\sum_{n=1}^{N}\left(-\tfrac{1}{2}\ln(2\pi\sigma^2) - \tfrac{\varepsilon_n^2(\mathbf{a})}{2\sigma^2}\right) \tag{6}$$

Searching the maximum as a function of the parameter vector $\mathbf{p}$ leads to the ML estimate $\widehat{\mathbf{p}}$. We can separate the estimation of σ and $\mathbf{a}$. Consequently we find:

$$\widehat{\mathbf{a}} = -v_{\mathbf{aa}}^{-1}(\mathbf{0})v_{\mathbf{a}}(\mathbf{0}) \text{ where } v(\mathbf{a}) = \frac{1}{N}\sum_{n=1}^{N}\varepsilon_n^2(\mathbf{a}) \tag{7}$$

and where $v_{\mathbf{a}}(\mathbf{0})$ and $v_{\mathbf{aa}}(\mathbf{0})$ are respectively the vector of first derivatives and the matrix of second derivatives (Hessian matrix) at $\mathbf{a} = \mathbf{0}$, and

$$\widehat{\sigma}^2 = v(\widehat{\mathbf{a}}) \text{ where } v(\widehat{\mathbf{a}}) = v(\mathbf{0}) - \tfrac{1}{2} v_{\mathbf{a}}(\mathbf{0}) v_{\mathbf{aa}}^{-1}(\mathbf{0}) v_{\mathbf{a}}(\mathbf{0}) \tag{8}$$

We construct the matrix V with elements V_{ij} $(0 \leq i \leq J \text{ and } 0 \leq j \leq J)$:

$$V = \begin{bmatrix} v(\mathbf{0}) & -\frac{1}{2} v_{\mathbf{a}}{}^{\mathsf{T}}(\mathbf{0}) \\ -\frac{1}{2} v_{\mathbf{a}}(\mathbf{0}) & \frac{1}{2} v_{\mathbf{aa}}(\mathbf{0}) \end{bmatrix} \text{ where } V_{ij} = \sum_{n=1}^{N} x_{n-i} x_{n-j} \tag{9}$$

Performing J Gauß-Jordan pivots on this matrix subsequently using V_{jj} $(1 \leq j \leq J)$ as pivot element leads to:

$$\begin{bmatrix} v(\widehat{\mathbf{a}}) & \widehat{\mathbf{a}}^{\mathsf{T}} \\ -\widehat{\mathbf{a}} & -2 v_{\mathbf{aa}}(\mathbf{0})^{-1} \end{bmatrix} = \begin{bmatrix} v(\mathbf{0}) - \frac{1}{2} v_{\mathbf{a}}(\mathbf{0}) v_{\mathbf{aa}}^{-1}(\mathbf{0}) v_{\mathbf{a}}(\mathbf{0}) & -v_{\mathbf{a}}{}^{\mathsf{T}}(\mathbf{0}) v_{\mathbf{aa}}^{-1}(\mathbf{0}) \\ v_{\mathbf{aa}}^{-1}(\mathbf{0}) v_{\mathbf{a}}(\mathbf{0}) & 2 v_{\mathbf{aa}}(\mathbf{0})^{-1} \end{bmatrix} \tag{10}$$

Notice that V_{00} is the only element of the main diagonal that is not used as pivot element. Similarly we can determine partial inverses of the matrix representing a ML-estimate by using only a subset of the full parameter space. E.g. we find $v(\widehat{\mathbf{a}})$ where $\widehat{\mathbf{a}} = (\widehat{a_1}, \widehat{a_2}, 0, \ldots, 0)$ by only using V_{11} and V_{22} as pivot element.

If we have J AR-parameters, there exist 2^J partial inverses representing ML-estimates with 0 to J AR-parameters. If J becomes large, the selection of the optimal partial inverse becomes rather computationally intensive. To reduce the computational effort, we only use the main diagonal element V_{kk} which lead to a maximum reduction of V_{00} as pivot element. This reduction can be predicted by evaluation of $V_{00} - \tilde{V}_{00} = V_{0k} V_{k0} / V_{kk}$. During the partial inversion process we use the element V_{kk} which has a maximum $V_{0k} V_{k0} / V_{kk}$ as the next pivot element. Now the decision remains whether the inclusion of the parameter a_k to the set of non-zero parameters leads to a better model. Any V_{kk} $(1 \leq k \leq J)$ which is used for the first time as pivot element during the partial inversion process leads to a decrease of V_{00} because $V_{0k} V_{k0} / V_{kk} \geq 0$. Any additional parameter leads to a model with smaller $\widehat{\sigma}$. Searching the *optimal* model, and the model with minimum $\widehat{\sigma}$, we use MIC to construct a threshold to determine whether an additional parameter leads to a better model.

Substitution of the ML parameter estimates (7) and (8) into the ALL-function 6 leads to the Maximum Average Log Likelihood (MALL) of the hypothesis:

$$\widehat{\mathrm{E}} \{ \underline{l}(\widehat{\mathbf{p}}) \} = -\tfrac{1}{2} \ln (2\pi \widehat{\sigma}^2) - \tfrac{1}{2} \tag{11}$$

We determine the increase of the MALL in case of the introduction of one additional parameter. Assume we have an estimated parameter vector $\widehat{\mathbf{p}}_0$, given the parameter configuration of the null hypothesis, and similar estimated parameter vectors $\widehat{\mathbf{p}}_k$, given parameter configurations of the kth alternative hypotheses

	0	1	2	3	4	5	6	7	8	corr.
$N = 100$										
4	0	197	8052	1719	32					0
6	0	263	8450	1273	14	0	0			0
8	0	340	8645	1003	12	0	0	0	0	0
10	0	379	8758	852	11	0	0	0	0	0
$N = 1000$										
4	0	0	0	2572	7428					0
6	0	0	0	3004	6777	217	2			0
8	0	0	0	3737	6138	122	3	0	0	0
10	0	0	0	4245	5660	92	3	0	0	0
$N = 10000$										
4	0	0	0	0	10000					0
6	0	0	0	0	256	7532	2212			2212
8	0	0	0	0	431	8523	1014	30	2	725
10	0	0	0	0	572	8685	714	28	1	431

Table 1: The observed number of parameter configurations as a function of the number of AR-parameters (horizontal) and the size of the AR-parameter space J (vertical) given 10.000 sequences of N observations generated according to the AR-model $x_n = \frac{1}{2}x_{n-1} + \frac{1}{4}x_{n-2} + \frac{1}{8}x_{n-3} + \frac{1}{16}x_{n-4} + \frac{1}{32}x_{n-5} + \frac{1}{64}x_{n-6} + \varepsilon_n$. The probability on a too high number of AR-parameters α is 5%. In the last column we present the number of times the model has correctly been identified.

having one additional AR-parameter with respect to the null hypothesis. The difference is MALL depends on k and equals:

$$\widehat{\mathrm{E}}\left\{\triangle\underline{l}(\widehat{\mathbf{p}}_0, \widehat{\mathbf{p}}_k)\right\} = \widehat{\mathrm{E}}\left\{\underline{l}(\widehat{\mathbf{p}}_k)\right\} - \widehat{\mathrm{E}}\left\{\underline{l}(\widehat{\mathbf{p}}_0)\right\} = \ln\left(\frac{\widehat{\sigma}_0}{\widehat{\sigma}_k}\right) = -\tfrac{1}{2}\ln\left(1 - \frac{V_{0k}V_{k0}}{V_{00}V_{kk}}\right) \quad (12)$$

Compare this equation with (4). If the additional parameter is superfluous ($\widehat{\sigma}_0 \approx \widehat{\sigma}_k$) this statistic is chi-squared distributed [3] with one degree of freedom. If the additional parameter is a significant contribution to the model ($\widehat{\sigma}_0 \gg \widehat{\sigma}_k$), the expectation of (12) differs significantly from zero and the statistic is normally distributed [3]. We are only interested in the first case because the second case is a typical indication that the algorithm should proceed in adding more parameters.

Assume there are K different parameters which can possibly be added to the space of non-zero parameters. If $K = 1$ we have a situation which equals the situation with AR-order estimation where you possibly add one pre-selected parameter in every iteration step. The threshold η_{high} for the FAP α in case of one additional parameter can be solved from the equation [3]:

$$\alpha = \int_{2N\eta_{high}}^{\infty} f_{\chi^2}(x)\mathrm{x} = 1 - \mathrm{erf}\left(\sqrt{N\eta_{high}}\right) \text{ so } \eta_{high} = \frac{\left(\mathrm{erf}^{-1}\left(1-\alpha\right)\right)^2}{N} \quad (13)$$

If $K > 1$ the threshold should be higher because there are two or more possibilities of adding one parameter to the space of non-zero parameters.

Assume that the differences in MALL, $\widehat{\mathrm{E}}\left\{\triangle\underline{l}(\widehat{\mathbf{p}}_0, \widehat{\mathbf{p}}_k)\right\}$ for different k, in case of adding different superfluous parameters are statistically independent amongst

each other. At this moment there is no theoretical evidence for this assumption. Empirically we obtain good results, so this assumption seems to be reasonable.

Given this assumption, the maximum of $\widehat{\mathrm{E}}\{\triangle\underline{l}(\widehat{\mathbf{p}}_0, \widehat{\mathbf{p}}_k)\}$ has the distribution:

$$f_{\Pi}(x) = K \frac{\left(\mathrm{erf}\left(\sqrt{x/2}\right)\right)^{K-1}}{\exp x/2\sqrt{2\pi x}} \tag{14}$$

Instead of (4) we use as a criterion:

$$\max \widehat{\mathrm{E}}\{\triangle\underline{l}(\widehat{\mathbf{p}}_0, \widehat{\mathbf{p}}_k)\} > 2N\eta_{high} \tag{15}$$

where we search the maximum as a function of the K alternative hypotheses. Similar to (13) we determine the threshold as a function of the FAP by solving η_{high} from the equation:

$$\alpha = \int_{2N\eta_{high}}^{\infty} f_{\Pi}(x)\mathrm{x} = 1 - \left(\mathrm{erf}\left(\sqrt{N\eta_{high}}\right)\right)^K \text{ so } \eta_{high} = \frac{\left(\mathrm{erf}^{-1}\left(\sqrt[K]{1-\alpha}\right)\right)^2}{N} \tag{16}$$

If the statistical dependence of the $\widehat{\mathrm{E}}\{\triangle\underline{l}(\widehat{\mathbf{p}}_0, \widehat{\mathbf{p}}_k)\}$ plays a significant role, we should use a lower threshold η_{high}, because the effective number of parameters decreases. Keeping η_{high} unchanged will reduce the FAP.

Assume we have J AR-parameters and an empty space of non-zero parameters; i.e. all parameters are zero. In the jth iteration ($1 \leq j \leq J$) of the estimation process we can add one parameter to the space of non-zero parameters by performing a Gauß-Jordan pivot, so this space contains exactly $j-1$ different AR-parameters. This means that we can choose between $K = J - j + 1$ different parameters (pivot-elements) to be added. We select the parameters with the largest $V_{0k}V_{k0}/V_{kk}$; this is the parameter which leads to a maximum increase of the MALL. We compare with (15) the K results of (12) with η_{high} of (16); if this difference in MALL is larger than η_{high} we add the parameter to the space of non-zero parameters, by pivoting the matrix V with pivot element V_{kk}, and continue the iteration. If the difference in MALL is smaller we stop the iteration and the optimal model is reached.

RESULTS

First we have tested the behavior of our method in case of no AR-model. In this case we are fitting AR-models to white noise; the results were satisfactory.

In table 1 we test our method is case of an AR-model where the importance of the parameters a_i decreases with i. We expect an underestimation of the number of parameters which decreases with increasing sample size N. For $N = 10000$ the five AR-parameter models seems to be optimal. As expected the number of estimated models with 6 parameters decreases with increasing J due to the increasing threshold.

	0	1	2	3	4	5	6	7	8	corr.
$N = 100$										
2	1	9286	713							713
4	1	9448	535	16	0					235
6	2	9494	493	11	0	0	0			144
8	5	9505	479	10	1	0	0	0	0	103
10	5	9470	513	11	1	0	0	0	0	94
$N = 1000$										
2	0	6570	3430							3430
4	0	7618	2311	69	2					1869
6	0	8056	1881	62	1	0	0			1444
8	0	8274	1657	66	3	0	0	0	0	1205
10	0	8308	1628	62	2	0	0	0	0	1131
$N = 10000$										
2	0	9	9991							9991
4	0	50	9509	428	13					9473
6	0	80	9495	407	17	1	0			9449
8	0	103	9473	414	10	0	0	0	0	9421
10	0	122	9460	405	13	0	0	0	0	9407

Table 2: As table 1 using the AR-model $x_n = 0.55x_{n-1} + 0.05x_{n-2} + \varepsilon_n$ and $\alpha = 5\%$.

A model with a possibly irrelevant parameter $a_2 = 0.05$ is presented in table 2. We expect that for small N the model with and without the parameter a_2 are indistinguishable. This is confirmed by the clear preference for the one AR-parameter model for $N \leq 1000$. Using $N = 10000$ we are capable of correctly identifying the correct model. Our claim of a FAP less than 5% is confirmed by the simulation results although in a significant number of cases a too simple model has been found.

To verify the results in a fourth order model with two zero parameters we studied the model $x_n = 0.50\ x_{n-1} - 0.25\ x_{n-4} + \varepsilon_n$ (see table 3). This model shows the advantages of our method. Our claim of less than 5% wrong identifications with irrelevant additional parameters has never been violated. For $N \geq 1000$ all identified two AR-parameter models correspond with the correct model.

Finally we have used the model of table 3 in table 3 to verify the behavior of our method for other values of α. In this table we also find an excellent agreement between the simulations and the theory.

CONCLUSIONS

The *correct* model is not always the *optimal* model. It is a better strategy to estimate the AR-parameter configuration instead of the AR-order, accepting that the AR-parameters are not all non-zero. From a statistical point of view, parameters with a value negligible with respect to the estimation error, can better be fixated at zero. This leads to accepting some bias in the optimal model in order to decrease the estimation error.

The algorithm performs satisfactory; in most applications the claim of an a priori selected probability of estimating a model with too many parameters

	0	1	2	3	4	5	corr.
$N = 100$							
4	26	2737	7006	219	12		6487
6	32	3261	6460	241	6	0	5682
8	47	3645	6076	228	4	0	5293
10	45	3764	5969	218	4	0	5143
$N = 1000$							
4	0	0	9514	477	9		9514
6	0	0	9528	466	6	0	9528
8	0	0	9532	458	10	0	9532
10	0	0	9546	446	8	0	9546
$N = 10000$							
4	0	0	9503	487	10		9503
6	0	0	9499	483	17	1	9499
8	0	0	9529	456	13	2	9529
10	0	0	9505	480	15	0	9505

	0	1	2	3	4	5	6	corr.
$\alpha = 1\%$								
4	0	0	9912	88	0			9912
6	0	0	9907	93	0	0	0	9907
8	0	0	9907	93	0	0	0	9907
10	0	0	9918	82	0	0	0	9918
$\alpha = 2\%$								
4	0	0	9812	187	1			9812
6	0	0	9807	191	2	0	0	9807
8	0	0	9811	187	2	0	0	9811
10	0	0	9822	178	0	0	0	9822
$\alpha = 10\%$								
4	0	0	9060	889	51			9060
6	0	0	9022	934	43	1	0	9022
8	0	0	9070	892	38	0	0	9070
10	0	0	9112	854	33	1	0	9112
$\alpha = 20\%$								
4	0	0	8118	1664	218			8118
6	0	0	8114	1669	199	17	1	8114
8	0	0	8155	1639	196	8	2	8155
10	0	0	8199	1613	175	13	0	8199

Table 3: As table 1 using the AR-model $x_n = 0.50x_{n-1} - 0.25x_{n-4} + \varepsilon_n$ and $\alpha = 5\%$ (left). Instead of the sample size N we also modify α for a given sample size $N = 1000$ (right).

is fulfilled. The only theoretical weakness is the assumption of the statistical independence of the differences in MALL (12). Although this assumption seems to be confirmed by simulations, a better theoretical foundation is necessary.

In some cases our method fails systematically; we believe that this failure is not caused by a wrong estimation method, but has a theoretical foundation. Some models are given a certain number of observations N indistinguishable. The problem of indistinguishable models has been neglected in the literature.

The computational effort will mainly be determined by the number of samples N, the number of evaluated parameters J and the actually estimated number of parameters M. If the number of samples is large, the most costly operation is the construction of V and the complexity behaves like $O\{N \times J\}$. So, for large N the algorithm has a linear complexity. The pivoting process, which costs $O\{M \times J^2\}$ units of time, dominates for small N.

REFERENCES

[1] R. Moddemeijer, "An ARMA model identification algorithm," in *Seventh Symposium on Information Theory in the Benelux*, D.E. Boekee, Ed. May 22-23 1986, pp. 151–159, Werkgemeenschap Informatie- en Communicatietheorie.

[2] G.E.P. Box and G.M. Jenkins, *Time series analysis: forecasting and control*, Holden-Day series in time series analysis. Holden-Day, San Francisco, 1970.

[3] R. Moddemeijer, "Testing composite hypotheses applied to AR order estimation; the Akaike-criterion revised," Resubmitted to IEEE Transactions on Signal Processing, 1997.

OPTIMAL RESOURCE MANAGEMENT ALGORITHM FOR ADAPTIVE MODELLING

J. Levendovszky[1], L. Kovács[1], I. Koller[1], E.C. van der Meulen[2]

[1] Department of Telecommunications, Technical University of Budapest, Budapest, Hungary

[2] Department of Mathematics, Katholieke Universiteit Leuven, Leuven, Belgium

Adaptive modelling play a central role in information theory and modern information technologies (e.g. adaptive predictive coding, adaptive channel equalization, ... etc.). In order to achieve high performance, the modelling capability of the adaptive system should be of the same degree as of the unknown system. Undermodelling results in loss of performance, whereas overmodelling uses the modelling resources inefficiently. This is typically the case in adaptive noise cancellation when multi-channel cancellation must be performed by a single DSP. Thus, the filterlength belonging to a single channel should not exceed the degree of the modelled propagation medium, otherwise there is not enough capacity left for other channels. As a result, traditional modelling algorithms such as recursive LMS must be modified in order to be able to model the degree of the system properly. In this paper a new algorithm, called Adaptive Filter Degree Adaptation (AFDA), is developed for this purpose. AFDA not only adapts the weights of an FIR filter according to the well known LMS algorithm but also adaptively determines the filter degree needed for modelling the system. In this way, optimal resource management is guaranteed by striking a good balance between modelling degree and performance. In practice, this makes possible to solve multi-channel modelling tasks by a single piece of hardware.

Besides proposing AFDA as a new algorithm and revealing its performance, the other main results of the paper is to ascertain the residual error analytically as a function of the modelling degree.

Keywords: stochastic approximation, adaptive signal processing, linear architectures, hypermatrices

THE MODEL

The problem of adaptive modelling is described as follows:

- Given a random sequence x_k, which is the primary input sequence. We assume that x_k is short term weakly stacionary and ergodic on a smaller time scale (which does not exclude the possibility of x_k being highly non-stacionary on a larger time scale).
- The unknown system to be modeled is considered to be linear, thus at the reference point the following sequence appears $d_k = \sum_{j=0}^{M} a_j x_{k-j}$, where $a_j, \quad j = 0, ..., M$ represent the coefficients of the unknown system. As a result, the exact values of $a_j, \quad j = 0, ..., M$ and M are unknown to us.
- The input sequence x_k is fed into an adaptive FIR architecture, which yields an estimated system output sequence y_k at the reference point, given as $y_k = \sum_{j=0}^{L} w_j x_{k-j}$.
- At the reference point an error signal is made by substracting the system output from the estimated output $e_k = y_k - d_k$. The task is to find the optimal weights and degree of the adaptive FIR filter, which minimizes the residual mean square error, which is

$$\min_{w_{opt}, L_{opt}} : \quad \frac{1}{N} \sum_{k=1}^{N} e^2(k). \tag{1}$$

In our novel algorithm we include L into the optimization process, making the degree of the filter also adaptive.

THE TRADITIONAL APPROACH TO ADAPTIVE IDENTIFICATION

The traditional method to minimize the mean square error as a function of the filter length is usually performed by the LMS algorithm given in the form of

$$w_l(k+1) = w_l(k) + \delta(d_k - \sum_{j=0}^{L} w_j(k) x_{k-j}) x_{k-l}, \tag{2}$$

where $w_l(k)$ denotes the weight belonging to the lth tap at time instant k and δ_k is the so-called learning constant. In order to ascertain the residual error one can use the following approximation (see [1, 2])

$$\lim_{N \to \infty} \frac{1}{N} \sum_{k=1}^{N} e^2(k) = E \left(d_k - \sum_{j=0}^{L} w_j x_{k-j} \right)^2,$$

therefore $\lim_{N\to\infty} E \parallel \mathbf{w_{opt}} - \mathbf{w_{opt}^{(N)}} \parallel^2 = 0$, where $\mathbf{w_{opt}}$ is given by the following equation

$$\mathbf{R}\mathbf{w_{opt}} = \mathbf{r}. \tag{3}$$

Here matrix $\mathbf{R}$ is defined as $R_{ij} = E(x_{k-i}x_{k-j})$ and vector $\mathbf{r}$ is $r_i = E(d_k x_{k-i})$ respectively. Using this approximation the residual error can be calculated as the minimum of $J(\mathbf{w}) = E(d_k^2) - 2\mathbf{w^T r} + \mathbf{w^T R w}$, which results in

$$J_{res}(\mathbf{w_{opt}}) = E(d_k^2) - \mathbf{w_{opt}^T R w_{opt}}. \tag{4}$$

The transient time of algorithm (2) can be upperbounded by the expression

$$CT \sim M\frac{\log\delta}{log\beta_{max}} - \frac{\log\left(\prod_{i=0}^{M}\log(v(0)-v_{opti})\right)}{\log\beta_{max}}, \tag{5}$$

where β_{max} denotes the maximum eigenvalue of matrix $\mathbf{R}$. With regard to algorithm (2), one can pose the following questions:

1. What is the effect of undermodelling ($L < M$) on the system performance? (In other words, we would like to derive a formula which expresses the residual error as a function of modelling filter degree $J_{res}(L)$.)
2. Without having exact knowledge about M, how to set (or adaptively modify) the modelling degree L in order to get a close approximation of the unknown M ?

THE RESIDUAL ERROR AS A FUNCTION OF THE MODELLING DEGREE

In this section, we derive a formula which expresses the residual error as a function of modelling filter degree $J_{res}(L)$. This function is summarized in the following theorem.

Theorem:
The residual error as a function of the degree of the identification filter, $J_{res}(L)$ decreases as $J(L) \approx O(const(M-L)\gamma_{min}a_{min}^{m2})$, where γ_{min} is the minimal eigenvalue of matrix $\mathbf{R^m} - \mathbf{R^{cT}R^{-1}R^c}$ and a_{min}^{m2} is the minimal component of vector $\mathbf{a^m}$.

The notations used in the theorem is given as follows:

1. the M dimension coefficient vector $\mathbf{a}$ from expression $\sum_{j=0}^{M} a_j x_{k-j}$ is partitioned into $\mathbf{a} = (\mathbf{a^L}, \mathbf{a^m})$ where $\mathbf{a^L} = (a_0, ..., a_L)$ and $\mathbf{a^m} = (a_{L+1}, ..., a_M)$, respectively;
2. the corresponding correlation matrix $\mathbf{R}$, which is of type MxM, is partitioned into

$$\mathbf{R} = \begin{pmatrix} \mathbf{R^L} & \mathbf{R^{cT}} \\ \mathbf{R^c} & \mathbf{R^m} \end{pmatrix},$$

where $\mathbf{R^L}$ is of type LXL, $\mathbf{R^c}$ is of type $(M-L)XL$ and $\mathbf{R^m}$ is of type $(M-L)X(M-L)$.

To prove the theorem one can rewrite expression (4) of the residual error into the following form

$$J(\mathbf{w_{opt}}) = \mathbf{a^{mT}}(\mathbf{R^m} - \mathbf{R^{cT}}\mathbf{R^{L^{-1}}}\mathbf{R^c})\mathbf{a^m}. \tag{6}$$

This form can easily be derived as follows:
First let us determine the optimal weights based on equation (3). Here again

$$R_{ij} = E(x_{k-i}x_{k-j})$$

and

$$r_i = E(d_k x_{k-i}) = E\left((\sum_{j=0}^{L} a_j x_{k-j} + \nu_k)x_{k-i}\right)^2 =$$

$$\sum_{j=0}^{L} a_j E(x_{k-j}x_{k-l}) + E(\nu_k x_{k-i}) = \sum_{j=0}^{L} a_j R_{ij} + b_i,$$

or in vectorial notation $\mathbf{r} = \mathbf{Ra} + \mathbf{b}$. Note that the matrix $\mathbf{R^L}$ is of type LxL. Thus, $\mathbf{w_{opt}}$ can be obtained as follows:

$$\mathbf{R^L w_{opt}} = \mathbf{R^L a} + \mathbf{b}. \tag{7}$$

From this $\mathbf{w_{opt}} = \mathbf{a} + \mathbf{R^{L^{-1}}b}$. Using this expression, the residual error given by (4) can be expressed as

$$(\mathbf{a}-\mathbf{w})^T\mathbf{R^L}(\mathbf{a}-\mathbf{w}) + 2(\mathbf{a}-\mathbf{w})^T\mathbf{b} + E(\nu_k^2) =$$

$$\mathbf{b^T}\mathbf{R^{L^{-1T}}}\mathbf{R^L}\mathbf{R^{L^{-1}}}\mathbf{b} - 2\mathbf{b^T}\mathbf{R^{L^{-1}}}\mathbf{b} + E(\nu_k^2) =$$

$$E(\nu_k^2) - \mathbf{b^T}\mathbf{R^{L^{-1}}}\mathbf{b} \tag{8}$$

By using the definition of **b**

$$b_i = E(\nu_k x_{k-i}) = E\left(\sum_{j=L+1}^{M} a_j x_{k-j} x_{k-i}\right) = \sum_{j=L+1}^{M} a_j R_{ij}^{L}$$

and

$$E(\nu_k^2) = E\left(\sum_{j=L+1}^{M}\sum_{i=L+1}^{M} a_j a_i x_{k-j} x_{k-i}\right) =$$

$$\sum_{j=L+1}^{M}\sum_{i=L+1}^{M} a_j a_i E(x_{k-j} x_{k-1}) = \sum_{j=L+1}^{M}\sum_{i=L+1}^{M} a_j a_i R_{ij}^{m} = \mathbf{a^{mT}R^m a^m}.$$

hypermatrices (the interested readers can turn to [2]), one can arrive at $J(L) \approx O\left(const(M-L)\gamma_{min} a_{min}^{m2}\right)$.

As a result we can conclude that if the degree of undermodelling of the unknown system is $M - L$ then the residual error decreases proportionally.

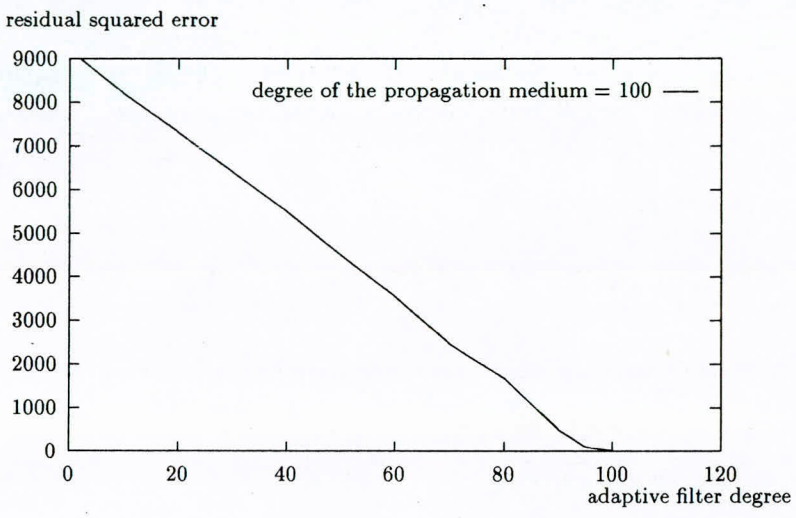


Figure 1 *The residual error as a function of the modelling degree*

FILTER DEGREE ADAPTATION

As was stressed before, the success of adaptive identification hinges upon the difference between the model degree M and the modelling degree L. In the case of undermodelling, when $M > L$, the residual error can remain high after the

convergence (see Theorem 1). In the case of overmodelling, when $M < L$ the resource management is not optimal.
Traditional methods, such as proposed by Akaike [6] and Rissanen [5] used information theoretic measures to obtain this degree. Based on the minimization of the Kullback Leibler mean information, the following estimation was obtained:

$$L = -R(\theta) + 0.5m \log N \tag{9}$$

where $R(\theta) = \max \sum_{i=1}^{N} \log g(x_i \mid \mathbf{w})$ Equation (8) is obtained form minimizing

$$\int_{-\infty}^{\infty} g(\mathbf{y} \log g(\mathbf{x}) d\mathbf{x} - \int_{-\infty}^{\infty} g(\mathbf{x} \log g(\mathbf{x} \mid \mathbf{w}) d\mathbf{x}.$$

This estimation though, proves to be rough in practice. Therefore, one may wish to find out the exact modelling degree in the course of the weight adaptation. This can be done by using the Adaptive Filter Degree Algorithm (AFDA), described as follows:

Given: An initial filter degree L, the initial filter coefficients $w_0(0), w_1(0), ..., w_L(0)$, the learning coefficient δ, a stabilization factor α, an averaging window K, and a pre-set value of the residual error ε.

1. Start to perform the recursion

$$w_l(k+1) = w_l(k) - \delta \left(\sum_{j=0}^{L} w_j(k) x_{k-j} - d_k \right) x_{k-l}$$

   until the weights are stabilized which is measured by the following criterion

$$\sum_{j=0}^{L} (w_j(k+1) - w_j(k))^2 \leq \alpha$$

2. When the weights are stabilized measure the main square error by the formula $\frac{1}{K} \sum_{k=1}^{K} e_k^2$ and
   if $\frac{1}{K} \sum_{k=1}^{K} e_k^2 \leq \varepsilon$ then stop the algorithm or
   if $\frac{1}{K} \sum_{k=1}^{K} e_k^2 > \varepsilon$ then increment the filter degree by setting $L := L + 1$ and go Back to *Step 1*!

Since the degree of the filter (the number of coefficients) is gradually increased until it reaches the degree of the model, AFDA is expected to yield very small

residual error. Simulations prove that this algorithm is capable to find the unknown model degree, therefore optimal resource management is guaranteed.

APPLICATION TO ADAPTIVE NOISE CANNCELLING AND NUMERICAL RESULTS

A straightforward application of AFDA is in the field of active noise cancellation. In this case the unknown system to be modelled is a propagation medium (e.g. air) between the primary noise source and the cancellation point. The modelling filter is an adaptive FIR architecture implemented on a DSP. To find the exact number of filter coefficients of the FIR filter (modelling degree) is essential in order to leave enough room for other applications running on the DSP (e.g., multichannel noise cancellation). Therefore, optimal resource management is crucial when implementing a multichannel active noise cancellation system.
Figure 1 shows the residual error as a function of the FIR degree while obtained recursively by AFDA. It can be seen in Figure 3 that the unknown degree of the propagation medium is found by AFDA.

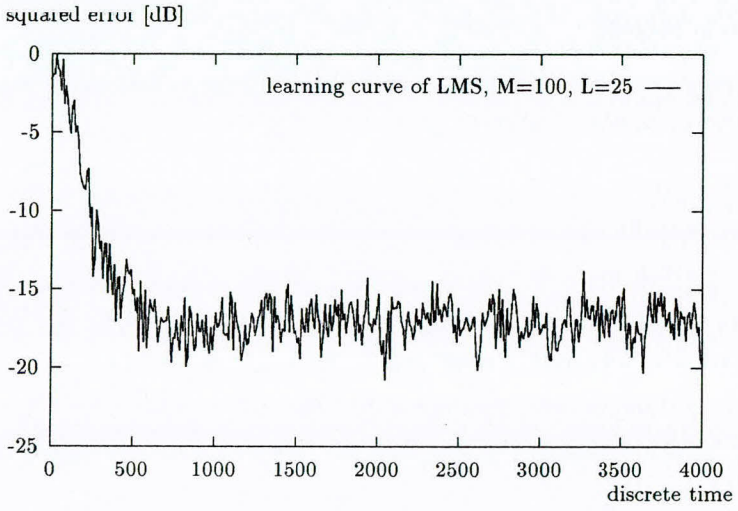


Figure 2 *Learning curve of an undermodelled system*

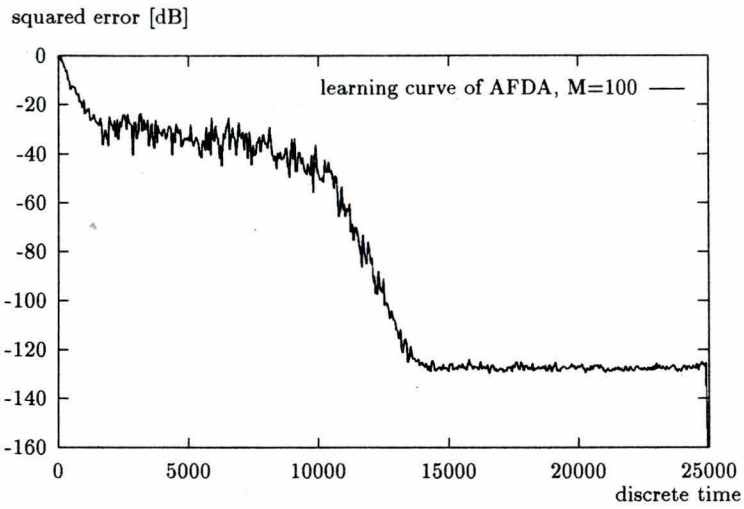


Figure 3 *Learning curve by AFDA*

ACKNOWLEDGEMENT

The research reported here was supported by the Hungarian National Science Foundation, project number T025964.

REFERENCES

[1] Widrow, B., Stearns, S.D.: *Adaptive signal processing*, Prentice Hall, 1985.

[2] Roberts , A.R., Mullis, C.T.: *Digital signal processing*, Addison Wesley, 1990

[3] Kuo, S.M., Morgan D.R.: *Active noise control systems, Algorithms and DSP implementations*, John Wiley, 1996

[4] Kailath, T.: *Linear systems*, Prentice Hall, 1990

[5] Rissanen, J,: "Modelling by shortest data description", *Automatica*, vol. 14., pp. 465-471

[6] Akaike, H,: "A new look at the statistical model identification", IEEE Trans. Automatic Control, vol. AC-19, pp. 716-723.

MULTIRATE BLOCK CODES

J.B.H. Peek

Computing Science Department
Nijmegen University

INTRODUCTION

Control of the transmitted signal spectrum is essential in digital transmission systems [1]. If the power spectral density (PSD)-function of the transmitted signal differs greatly from the frequency response of the channel, severe ISI and loss of signal power will result.

In [2, 3] it is shown that the transmitted signal spectrum can be shaped by using a linear discrete-time multirate digital filter. "Multirate" refers to the property that the output sampling rate can be larger or smaller than the input sampling rate. With these filters spectral nulls at DC or (and) at the Nyquist frequency can be created. The multirate filter operation is described by a $K \times L$ matrix that transforms a block of L bipolar input samples into the block of K output samples with two or more uniformly spaced levels. Thus the transformation is given by $\underline{y} = A\underline{x}$, where A is a $K \times L$ matrix and $\underline{x}$ and $\underline{y}$ are column vectors of length L and K respectively.

Note that a baseband signal $s(t) = \sum_{i=1}^{K} y(i)\, p(t - iT)$, where $p(t)$ is a pulse and T the time interval between samples, may have shaping contributed by both the block code and the pulse $p(t)$. Examples of matrices were given in [2, 3]. In order that such a matrix transformation is one-to-one, the matrix must be $\{-1, 0, 1\}$-nonsingular [2]. All matrices described hereafter have this property.

In [2] it was shown that if the bipolar input samples are equiprobable and statistically independent then the PSD-function of the output process $y(m)$ is

$$S_{yy}(e^{j\theta}) = \frac{1}{K} \cdot \|\underline{\mathbf{z}}\|^2 \tag{1}$$

where $\underline{\mathbf{z}}$ is a complex vector defined by

$$\underline{\mathbf{z}} = \sum_{p=1}^{K} \underline{a}_{p*}\, e^{-j\theta(p-1)} \tag{2}$$

and where $\underline{a}_{p*}$ denotes the p-th row of A and where $\theta = \omega T$ is the relative frequency.

MULTIRATE BLOCK CODES WITH SPECTRAL NULL AT DC OR AT THE NYQUIST FREQUENCY

For a multirate (MR)-code described by a matrix A and with a spectral null at DC, $\underline{z} = \underline{0}$ for $\theta = 0$. In that case

$$\underline{0} = \sum_{p=1}^{K} \underline{a}_{p*} \ . \tag{3}$$

In other words, the sum of the elements of each column of A is zero. An example of a matrix with a spectral null at DC is

$$C = \begin{pmatrix} 1 & 1 & & & & & & \\ 1 & -1 & & & & & & \\ -1 & & 1 & & & & & \\ -1 & & & 1 & & & & \\ & & -1 & & \cdot & & & \\ & & & \cdot & & \cdot & & \\ & & & & \cdot & & \cdot & \\ & & & & & -1 & & 1 \\ & & & & & & -1 & -1 \end{pmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} K \ , \qquad K \geq 6$$

$$L = K - 1$$

in which for reasons of clarity the zero entries are not indicated. The sample output rate is $R = \frac{K}{K-1}$ times the sample input rate. If $\underline{x} \in \{-1, 1\}^L$ then $\underline{y} \in \{-2, 0, 2\}^K$. That is the output codewords have $M = 3$ levels. The property that all columns of C give a zero sum is not lost by permutation of the rows. A permutation can, however, result in a different shape of the PSD-function $S_{yy}(e^{j\theta})$. This gives an extra degree of freedom when selecting a code for a specific application. A DC-free MR-code has the additional property that the block digital sum, i.e. the sum of all samples in an output block, is zero. This can be seen from

$$\begin{aligned} S_1 \ &\triangleq \ y_1 + y_2 + ... + y_K = \underline{a}_{1*}^T \cdot \underline{x} + ... + \underline{a}_{K*}^T \cdot \underline{x} \ = \\ &= \ \underline{x}^T \cdot (\underline{a}_{1*} + ... + \underline{a}_{K*}) = \underline{x}^T \cdot \underline{0} = 0 \ . \end{aligned} \tag{4}$$

This property will be used below in decoding (error correction). Coding specialists may note that the DSV (digital sum variation) [1] for this C-code is 4 and the minimum Hamming and Euclidean distance are $d_H = 2$ and $d_E = 2\sqrt{2}$ respectively. To demodulate (input data recovery) an MR-code with a spectral null at DC, note that because of (3) the last row of C can be deleted, and similarly, because of (4), the last component of the vector $\underline{y}$ can be deleted. To use these properties in demodulation, define the reduced vector

$$\underline{y}_1 \ \triangleq \ (y_1, ..., y_{K-1})^T$$

and let C_1 be the square $(K-1)$ by $(K-1)$ matrix that is obtained by deleting the last row of C. Clearly, C_1 is invertible, and so the vector $\underline{x}$ can be recovered by

$$C_1^{-1}\,\underline{y}_1 = C_1^{-1}\,C_1\,\underline{x} = \underline{x}\ . \tag{5}$$

This inverse matrix operation can be implemented with an MR-filter. By changing the signs of the entries of the odd numbered rows of C, another matrix is obtained such that

$$\sum_{p=1}^{K} (-1)^p\,\underline{c}_{p*} = \underline{0}\ , \tag{6}$$

which implies, from (2), that instead of a DC null, a Nyquist null will result ($\underline{z} = \underline{0}$ for $\theta = \pm\pi$). If an MR-code, with matrix A, has a Nyquist null it follows that

$$\begin{aligned} S_2 \ &\triangleq\ -y_1 + y_2 - y_3 + \ldots + (-1)^K\,y_K \ = \\ &=\ -\underline{a}_{1*}^T \cdot \underline{x} + \underline{a}_{2*}^T \cdot \underline{x} - \ldots + (-1)^K \cdot \underline{a}_{K*} \cdot \underline{x} \ = \\ &=\ \underline{x}^T \cdot \{-\underline{a}_{1*} + \underline{a}_{2*} - \ldots + (-1)^K\,\underline{a}_{K*}\} = \underline{x}^T \cdot \underline{0} = 0\ , \end{aligned} \tag{7}$$

i.e. the output block sum with alternating signs, is zero.

SPECTRAL NULLS AT BOTH DC AND THE NYQUIST FREQUENCY

An MR-code that has a spectral null at DC and at the Nyquist frequency is given by

$$B = \begin{pmatrix} -1 & & & & & & & -1 \\ 1 & -1 & & & & & & \\ 1 & 1 & & & & & & \\ -1 & & 1 & & & & & \\ & -1 & & \cdot & & & & \\ & & \cdot & & \cdot & & & \\ & & & \cdot & & \cdot & & \\ & & & & \cdot & & 1 & \\ & & & & & -1 & & 1 \\ & & 1 & & & & -1 & \end{pmatrix} K\ , \qquad \text{where } K \geq 8 \text{ and even}\ .$$

$$L = K - 2$$

The output rate of the MR-filter for this code B is $R = \frac{K}{K-2}$ with $M = 3$. According to (4) and (7), $S_1 = 0$ and $S_2 = 0$ for this code. These two constraints can also be expressed in a different way. If S_0 denotes the sum of all odd numbered samples in an output block $\underline{y}$ and S_e denotes the sum of the even numbered samples in a block then (4) and (7) can be rewritten as

$$\begin{aligned} S_1 \ &=\ S_0 + S_e = 0 \\ S_2 \ &=\ -S_0 + S_e = 0 \ \text{ or} \\ S_0 \ &=\ S_e = 0\ . \end{aligned} \tag{8}$$

These constraints can be used in decoding (error correction) this B-code. Demodulation (input data recovery) is similar to that for the C-code. Instead of one row of the matrix and one component of $\underline{y}$, now two rows of B and two components of $\underline{y}$ can be deleted. Let $\underline{y}_2 \triangleq (y_1, ..., y_{K-2})^T$ and let B_1 be the square $(K-2)$ by $(K-2)$ matrix that is obtained by deleting the last two rows of B. Clearly, B_1 is invertible, and so the vector $\underline{x}$ can be recovered by

$$B_1^{-1}\,\underline{y}_2 = B_1^{-1}\,B_1\,\underline{x} = \underline{x}\ . \tag{9}$$

HIGHER-ORDER SPECTRAL NULLS

Sometimes a faster reduction of the transmitted signal power near DC or near the Nyquist frequency is required. This can be achieved with MR-codes with higher order spectral nulls. Karabed and Siegel [4] described partial response codes with higher order spectral nulls. Before describing MR-codes with a higher order spectral null at either DC or the Nyquist frequency it is necessary to extend the theory of MR-codes.

According to (2) the i-th component of $\underline{\mathbf{z}}$ is

$$\sum_{p=1}^{K} a_{pi}\, e^{-j\theta(p-1)}\ , \qquad 1 \le i \le L\ . \tag{10}$$

With z-transform notation ($z = e^{j\theta}$), (10) can be denoted as $H_i(z)$ and expressed as

$$H_i(z) \triangleq a_{1i} + a_{2i}z^{-1} + ... + a_{Ki}z^{-(K-1)}\ , \qquad 1 \le i \le L\ . \tag{11}$$

Therefore (1) can be rewritten as

$$S_{yy}(z) = \frac{1}{K} \cdot \sum_{i=1}^{L} H_i(z)\, H_i(z^{-1})\ , \quad \text{or} \tag{12}$$

$$S_{yy}(e^{j\theta}) = \frac{1}{K} \cdot \sum_{i=1}^{L} H_i(e^{j\theta})\, H_i(e^{-j\theta})\ . \tag{13}$$

This PSD-function has an order-N spectral null at $\theta = \theta_0$ if [4]

$$\frac{d^n}{d\theta^n}\, S_{yy}(e^{j\theta})\Big|_{\theta=\theta_0} = 0\ , \qquad n = 0, 1, ..., 2N-1\ . \tag{14}$$

For example if

$$H_i(e^{j\theta_0}) = H_i'(e^{j\theta_0}) = 0\ , \qquad \forall\, i\ , \tag{15}$$

then successive differentiations of (13) show that $S_{yy}(e^{j\theta})$ will have an order-2 spectral null for $\theta = \theta_0$. The Fourier transform of a block of K output samples is

$$Y(e^{j\theta}) = y_1 + y_2\, e^{-j\theta} + ... + y_K\, e^{-j(K-1)\theta}\ . \tag{16}$$

Equation (16) can also be written as

$$Y(e^{j\theta}) = \underline{x}^T \cdot (H_1(e^{j\theta}), ..., H_L(e^{j\theta}))^T \ .$$

If there is an order-2 spectral null at $\theta = \theta_0$ then, using (15),

$$Y'(e^{j\theta_0}) = \underline{x}^T \cdot \underline{0} = 0 \ .$$

For example an order-2 spectral null at DC implies that

$$S \triangleq Y'(e^{j0}) = y_2 + 2y_3 + ... + (K-1)\, y_K = 0 \ . \tag{17}$$

This constraint on the output samples in every block will be used later when decoding of an MR-code with an order-2 spectral null at DC or the Nyquist frequency is described.

We now describe an MR-code which has an order-2 spectral null at DC and an order-1 Nyquist null. It is given by the 8×4 matrix, generating a code with rate $R = 2$ and $M = 3$ levels,

$$D = \begin{pmatrix} 1 & & 1 & \\ & 1 & -1 & \\ -1 & & & 1 \\ & -1 & & -1 \\ -1 & & & -1 \\ & -1 & & 1 \\ 1 & & -1 & \\ & 1 & 1 & \end{pmatrix} \ . \tag{18}$$

The four polynomials associated with the four columns are

$$\begin{aligned} H_1(z^{-1}) &= 1 - z^2 - z^4 + z^6 = (1-z)^2(1+z)^2(1+z^2) \\ H_2(z^{-1}) &= z\, H_1(z^{-1}) = z(1-z)^2(1+z)^2(1+z^2) \\ H_3(z^{-1}) &= 1 - z - z^6 + z^7 = (1-z)(1-z^3)(1+z^3) \\ H_4(z^{-1}) &= z^2 - z^3 - z^4 + z^5 = z^2(1-z)^2(1+z) \ . \end{aligned}$$

Since these polynomials have a double zero at DC ($z = 1$) and a single zero at the Nyquist frequency ($z = -1$) in common, it follows that

$$H_i(z^{-1})\Big|_{z=1} = 0 \,, \quad H_i'(z^{-1})\Big|_{z=1} = 0 \ \text{ and } \ H_i(z^{-1})\Big|_{z=-1} = 0 \,, \qquad i = 1,2,3,4 \ . \tag{19}$$

Therefore $S_{yy}(z)$ has an order-2 spectral null at DC and an order -1 Nyquist null. From (8) and (17) it follows that for this D-code the following three constraints are valid

$$S_0 = 0 \,, \quad S_e = 0 \ \text{ and}$$

$$S \triangleq y_2 + 2y_3 + 3y_4 + 4y_5 + 5y_6 + 6y_7 + 7y_8 = 0 \ . \tag{20}$$

It can be shown that the minimum Hamming and minimum Euclidean distance of this D-code are $d_H = 4$ and $d_E = 4$ respectively. After decoding, the last three rows of D and the last 3 components of $\underline{y}$ can be deleted when demodulation is required. If $\underline{y}_3 \triangleq (y_1, y_2, y_3, y_4, y_5)^T$ and if D_1 is a left inverse (4 by 5) of the reduced (5-row) matrix D' then

$$D_1 \, \underline{y}_3 = D_1 \, D' \, \underline{x} = \underline{x} \; .$$

As a second example, a matrix that gives an order-2 Nyquist and an order -1 DC null is obtained by substituting for z the value $-z$ in the four polynomials mentioned before with the corresponding matrix

$$D'' = \begin{pmatrix} 1 & & 1 & \\ & -1 & 1 & \\ -1 & & & 1 \\ & 1 & & 1 \\ -1 & & & -1 \\ & 1 & & -1 \\ 1 & & -1 & \\ & -1 & -1 & \end{pmatrix} .$$

HIGHER NUMBER OF LEVELS TO CONSERVE BANDWIDTH

The matrices B, C, D and D'', which generate ternary ($M = 3$) codewords can be used to construct extended matrices that will give code words with $M = 7, 15, 31, ..., 2^N - 1$ levels. Consider for example the matrix

$$A = \begin{pmatrix} 1 & & 1 \\ & 1 & -1 \\ -1 & & 1 \\ & -1 & -1 \end{pmatrix} , \qquad R_1 = \frac{4}{3} , \quad M = 3 ,$$

which gives an order-1 DC null. We define the notation $(A2A)$ as the matrix A concatenated with a matrix in which all entries of A are multiplied by a factor two [2]. Thus

$$(A2A) = \begin{pmatrix} 1 & & 1 & 2 & & 2 \\ & 1 & -1 & & 2 & -2 \\ -1 & & 1 & -2 & & 2 \\ & -1 & -1 & & -2 & -2 \end{pmatrix} .$$

This matrix corresponds to a code with $M = 7$ levels, with $R_2 = \frac{4}{6}$ which is half the rate of $R_1 = \frac{4}{3}$. In this way bandwidth was reduced by a factor of 2 in exchange for more levels. It can be shown that matrices $(A2A)$, $(A2A4A)$, etc. are $\{-1, 0, 1\}$-non-singular and yield the same PSD-function as A, provided A is a matrix of the type B, C, D or D''. Hence multilevel MR-codes with the same spectral shaping properties as those of the B, C, D or D'' codes can be constructed with various bandwidth-number of levels tradeoffs. The samples of the $\underline{y}$ vector (block) generated by a B, C, D, or D'' code or one of their extensions, are transmitted over a channel as a pulse train in which each sample value $y(i)$

multiplies a pulse $p(t - iT)$. If $p(t)$ has the Fourier transform $P(f)$ then the PSD-function of the transmitted output process [5] is (recalling that $\theta = 2\pi fT$)

$$S(f) = \frac{1}{T} |P(f)|^2 \cdot S_{yy}(e^{j2\pi fT}) \ . \tag{21}$$

DECODING (ERROR CORRECTION) AND DEMODULATION (INPUT DATA RECOVERY)

Modeling the transmission channel as an AWGN (additive white Gaussian noise) channel and assuming that the autocorrelation function of the pulse $p(t)$ satisfies Nyquist's first criterion one obtains, after a filter matched to the pulse $p(t)$, a received vector

$$\underline{r}' = \underline{y} + \underline{n} \ , \tag{22}$$

where $\underline{y}$ is the transmitted codeword of length K and $\underline{n}$ is a noise vector of length K with statistically independent noise components each with a variance σ^2. As described earlier $\underline{y}$ consists of components with $M = 3, 7, 15$ or in general $2^N - 1$ levels. The components of $\underline{r}'$ are rounded to the nearest of the M levels, yielding

$$\underline{r} \ \triangleq \ (\underline{r}')_{\text{round-off}} \ . \tag{23}$$

We now describe a decoding algorithm for code C ($d_H = 2$, $d_E = 2\sqrt{2}$):

if	$S_0 + S_e = 0$
then	$\underline{r}$ correct
else if	$S_0 + S_e = \pm 2$
then	change that sample that has the largest rounding error $\lvert e_i \rvert_{\max}$ ($e_i = r_i' - r_i$) in such a way that the sum $S_0 + S_e$ becomes zero (this is Wagner decoding [6,7])
else if	$\lvert S_0 \rvert \geq 2$ and $\lvert S_e \rvert \geq 2$
then	detect error.

A decoding algorithm for the code B ($d_H = 2$, $d_E = 2\sqrt{2}$) looks similar. This decoding of the code B or C is not correct if $d_H(\underline{y}, \underline{r}) \geq 2$ or if $d_H(\underline{y}, \underline{r}) = 1$ and the determined error position (as based on $\lvert e_i \rvert_{\max}$) is in fact correct, or if $d_H(\underline{y}, \underline{r}) = 1$ and $\lvert e_i \rvert_{\max} > 2$. If $P_1(\text{error})$ denotes the probability of a decoding error then it can be shown that for small values of σ

$$P_1(\text{error}) \approx c_1 \, Q\Big(\frac{d_E}{2\sigma}\Big) \qquad \text{where } d_E = 2\sqrt{2} \text{ and } c_1 \text{ a constant} \ .$$

This result implies that for high SNR's the MLS (maximum likelihood sequence) detection bound is reached. A decoding algorithm for the D-code ($d_H = 4$, $d_E = 4$) is as follows: Let

$$S_0 \ \triangleq \ r_1 + r_3 + r_5 + r_7$$

$$
\begin{aligned}
S_e &\triangleq r_2 + r_4 + r_6 + r_8 \\
S &\triangleq r_2 + 2r_3 + 3r_4 + 4r_5 + 5r_6 + 6r_7 + 7r_8 \\
T &\triangleq S_0 + S_e + S = \sum_{i=1}^{8} i\, r_i
\end{aligned}
$$

if	$S_0 = S_e = S = 0$
then	$\underline{r}$ correct
	(first step: single error detection and correction)
else if	$S_0 = \pm 2$ and $S_e = 0$ and $T \neq 0$
then	error size is S_0 and error position is T/S_0, correct this sample by ∓ 2
else if	$S_0 = 0$ and $S_e = \pm 2$ and $T \neq 0$
then	error size is S_e and error position is T/S_e, correct this sample by ∓ 2
	(second step: double error detection and correction)
else if	$S_0 = \pm 4$ and $S_e = 0$ and $T \neq 0$,
	or $S_e = \pm 4$ and $S_0 = 0$ and $T \neq 0$,
	or $S_0 = \pm 2$ and $S_e = \pm 2$ and $T \neq 0$,
then	change that sample, either in an even or odd position, that has the largest rounding error such that the absolute value of S_0 ($S_0 \neq 0$) or S_e ($S_e \neq 0$) is reduced by 2
	(what is left is a single error which can be corrected using single error correction)
else if	$S_e = S_0 = 0$ and $T \neq 0$
then	determine that sample with the largest rounding error $\lvert e_i \rvert_{\max}$
	if $\lvert e_i \rvert_{\max} > 0$
	then correct that sample by adding 2
	if $\lvert e_i \rvert_{\max} < 0$
	then correct that sample by adding -2
	(what is left is a single error which can be corrected using single error correction)
else if	$\lvert S_e \rvert > 4$ or $\lvert S_0 \rvert > 4$
then	detect error.

Suppose $d_H(\underline{y}, \underline{r}) = 2$ and that the error positions are i and j in case a double error is detected. Then decoding is not correct if $\lvert e_i \rvert_{\max}$ does not occur at positions i or j or if $\lvert e_i \rvert_{\max} > 2$. If $P_2(\text{error})$ denotes the probability of a decoding error then it can be shown that for small values of σ

$P_2(\text{error}) \approx c_2\, Q(\frac{d_E}{2\sigma})$ where $d_E = 4$ and c_2 a constant. Thus for high SNR's the MLS detection bound is reached.

Demodulation of the MR-codes B, C and D has already been described. The first step in the demodulation of their extensions is similar to that for the B, C and

D codes. For example, the first step in the demodulation of the code $(B2B4B)$ with $R = \frac{K}{3(K-2)}$ and $M = 15$ is as follows. Assume that $\underline{r}$ is correct so that $\underline{r} = \underline{y}$ and

$$\underline{r} = \underline{y} = (B2B4B)\,\underline{x} \; . \tag{24}$$

Let $\underline{x} = \begin{pmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \underline{x}_3 \end{pmatrix}$ where $\underline{x}_1$, $\underline{x}_2$ and $\underline{x}_3$ are column vectors of length $(K-2)$. Equation (24) can then be rewritten as

$$\underline{y} = (B2B4B)\,\underline{x} = B(\underline{x}_1 + 2\underline{x}_2 + 4\underline{x}_3) \; .$$

The first demodulation step gives (9)

$$B_1^{-1}\,\underline{y}_2 = B_1^{-1}\,B_1(\underline{x}_1 + 2\underline{x}_2 + 4\underline{x}_3) = \underline{x}_1 + 2\underline{x}_2 + 4\underline{x}_3 \; .$$

This is an 8-ary PAM-vector of length $(K-2)$. The second and last step in demodulation of this PAM-vector to the original vector $\underline{x}$ is a standard procedure.

Acknowledgement. I want to thank Dr. L.M.G.M. Tolhuizen for the many fruitful discussions and for his valuable observations and comments and Dr. S.B. Weinstein and Dr. J.W.M. Bergmans for reading the manuscript and their helpful comments.

REFERENCES

[1] E.A. Lee, D.G. Messerschmitt, "Digital Communication", Kluwer Academic Publishers, Second Edition 1994, Chapter 12 (Spectrum Control).

[2] J.B.H. Peek, L.F.P.M. Lakeman, "Generating Block Line Codes with Spectrum Nulls Using Multirate Digital Filters", Proc. IEEE ICC 1991, Denver, June, 1991, pp. 1098–1102.

[3] R.D. Gitlin, J.F. Hayes and S.B. Weinstein, "Data Communications Principles", Plenum, 1992, Section 4.7, "Block codes: a multirate digital filtering approach", pp. 277–278.

[4] R. Karabed, P. Siegel, "Matched Spectral-Null Codes for Partial-Response Channels", IEEE Trans. on Information Theory", Vol. 37, No. 3, May 1991, pp. 818–855.

[5] R.D. Gitlin, J.F. Hayes and S.B. Weinstein, "Data Communications Principles", Plenum, 1992, Appendix 4A (Power density function of a correlated line signal).

[6] R.A. Silverman and M. Balser, "Coding for Constant Data Rate Systems", IRE Trans. Information Theory, Vol. 4, September 1954, pp. 50–63.

[7] S.G. Wilson, "Digital Modulation and Coding", Prentice Hall, 1996, p. 518.

THE BINARY MULTIPLYING CHANNEL WITHOUT FEEDBACK: NEW RATE PAIRS IN THE ZERO-ERROR CAPACITY REGION

Ludo Tolhuizen

Philips Research Laboratories, WY 61
Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands
tolhuizn@natlab.research.philips.com

Two terminals, T1 and T2, wish to communicate over the binary multiplying channel (BMC). To this end, they choose sets X and Y, respectively, of (input) vectors in $\{0,1\}^n$. If $\mathbf{x}\in X$ and $\mathbf{y}\in Y$ are fed to the channel, it gives as output the vector $\mathbf{x}\cdot\mathbf{y}$, defined by

$$(\mathbf{x}\cdot\mathbf{y})_i = x_i y_i \text{ for all } i \in \{1,2,\ldots,n\}.$$

Each terminal should be able to determine *unambiguously* the vector transmitted by the other one, using its own transmitted vector and the observed channel output. We call a pair (X,Y) satisfying this requirement *uniquely decodable*, or UD for short. Moreover, we call a UD pair (X,Y) *symmetric* if $X = Y$. Note that unlike [1], we do *not* allow feedback, that is, encoding of a message does not depend on the output bits observed so far.

Example Take $X = Y = \{(1,1,0),(1,0,1),(0,1,1)\}$. Suppose that T1 transmits (1,1,0). If T2 transmits (1,1,0),(1,0,1) or (0,1,1), the channel outputs are (1,1,0),(1,0,0) and (0,1,0), respectively. As these channel outputs are distinct, T1 can determine which vector was transmitted by T2. The same holds for the other possible input vectors of T1. By symmetry, T2 can act like T1, and we conclude that (X,X) is a (symmetric) UD pair.
Note that (1,0,0) occurs as output vector twice, namely if T1 transmits (1,1,0) and T2 transmits (1,0,1), or *vice versa*. The knowledge of its own transmitted vector allows each terminal to choose the correct alternative.

The natural question is how many words the codes of a UD pair can have. In order to compare UD pairs of unequal length, we consider the rates of the codes of such pairs, i.e., if (X,Y) is a UD pair of length n, we consider the rate pair $(R(X),R(Y)) = (\frac{1}{n}\log|X|, \frac{1}{n}\log|Y|)$ As usual, all logarithms have base 2.

We will call a rate pair (x, y) achievable if for each $\varepsilon > 0$, there exists a UD pair (X, Y) such that

$$R(X) > x - \varepsilon \quad \text{and} \quad R(Y) > y - \varepsilon.$$

The set of achievable rate pairs will be called the zero-error capacity region of the BMC without feedback and it will be denoted by Z.

In [2], we provide a construction of UD pairs from cosets of binary linear codes with many information sets and obtain the following theorem. As usual, h denotes the binary entropy function:

$$h(p) = -p \log p - (1-p) \log(1-p), \quad 0 < p < 1.$$

Theorem 1 $\{(h(R_2) + R_1 - 1, h(R_1) + R_2 - 1) \mid \frac{1}{2} \leq R_1, R_2 < 1\} \subset Z$. *Moreover, for* $\frac{1}{2} \leq R < 1$, *the rate pair* $(h(R) + R - 1, h(R) + R - 1)$ *can be achieved with symmetric UD pairs.*

Specializing Theorem 1 to the case $R_1 = R_2 = 2/3$, we find

Corollary 1 *The rate pair* $(\log(3/2), \log(3/2)) \approx (0.58500, 0.58500)$ *can be achieved with symmetric UD pairs.*

The rate pair of Corollary 1 yields the largest known sum of the rates of pairs in Z. The largest known sum rate so far was obtained by Shearer [3] who constructed an UD pair of length 72 with rate pair (0.54797,0.54797). It follows from [4, Thm. 3] that the rate pair of Corollary 1 is the largest possible that can be achieved with symmetric UD pairs. Stated differently, asymptotically our construction yields cancellative families of sets [4] [5, Sec. VII] of largest possible rate.

The results are represented graphically in Figure 1. The new rate pairs from Theorem 1 lie on and below the curve N, labelled by "new rate pairs". The line segment "upper bound" represents the upper bound of [6], which states that for any rate pair (x, y) in Z, $x + y \leq 1.2181$. [1]

As $(\{1\},\{0,1\})$ is a UD pair, $(0,1) \in Z$; similarly, $(1,0) \in Z$. With a time sharing argument [1, Sec. 8], it can be shown that Z is convex. Consequently, all rate pairs on and below the tangents to N through (1,0) and (0,1) are in Z. The relevant segments of these tangents are drawn as well.

[1] As remarked by Erik Meeuwissen in [7, Stellling 2], combination of this upper bound with Shannon's lower bound [1, Sec. 13] shows that the zero-error capacity region of the BMC without feedback is strictly smaller than its ε-error capacity region.

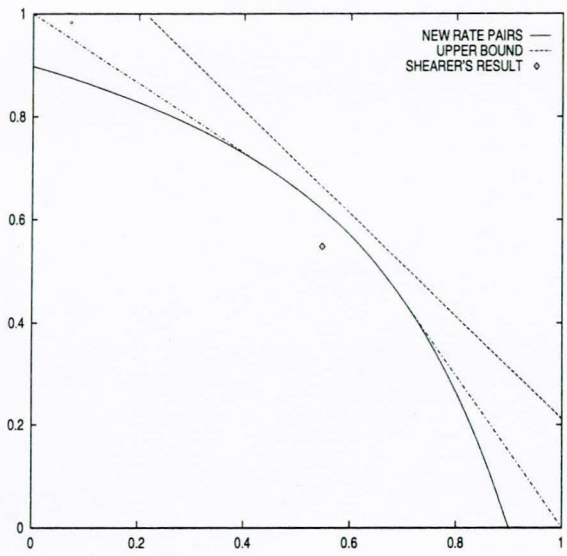


Figure 1: Graphical representation of the results. All points below the solid curve or its two tangents are in Z.

REFERENCES

[1] C.E. Shannon, "Two-way communication channels", Proc. 4th Berkeley Symp. Math. Stat. and Prob., 1961, pp. 611–644. Reprinted in *Key Papers in the Development of Information Theory*, D. Slepian(ed), IEEE Press, 1974, pp. 339–372.

[2] L.M.G.M. Tolhuizen, "New rate pairs in the zero-error capacity region of the binary multiplying channel without feedback", April 1999, submitted to *IEEE Trans. Inform. Th.*

[3] J.B. Shearer, "On Cancellative Families of Sets", *Electr. J. Combinatorics*, Vol. 1, No. 4, 1996.

[4] P. Frankl and Z. Füredi, "Union-free Hypergraphs and Probability Theory", *Europ. J. Combinatorics*, Vol. 5, 1984, pp. 127–131.

[5] J. Körner and A. Orlitsky, "Zero-Error Information Theory", *IEEE Trans. Inform. Th.*, **IT**-44, 6, Oct. 1998, pp. 2207–2229.

[6] R. Holzman and J. Körner, "Cancellative Pairs of Families of Sets", *Europ. J. Combinatorics*, Vol. 16, 1995, pp. 263–266.

[7] H.B. Meeuwissen, *Information Theoretical Aspects of Two-way Communication*, Ph.D. thesis, Eindhoven Univ. Techn., May 1998.

Error Correction Capabilities of Concatenated Coding Schemes with Single-Trial Bounded Distance Decoding and Optimized Erasing

Jos H. Weber*
Delft University of Technology, Fac. ITS
POB 5031, 2600 GA Delft, The Netherlands
J.H.Weber@ITS.TUDelft.NL

Khaled A.S. Abdel-Ghaffar†
University of California, Dept. ECE
Davis, CA 95616, USA
ghaffar@ece.ucdavis.edu

20TH SYMPOSIUM ON INFORMATION THEORY IN THE BENELUX,
HAASRODE, BELGIUM, MAY 27-28, 1999

For concatenated coding schemes with single-trial bounded distance decoding and optimized erasing, the maximum number of channel errors for which correction is guaranteed is determined.

I Introduction

We consider concatenated coding schemes using a Q-ary outer code of length N and Hamming distance D, and a q-ary inner code of length n, size Q, and Hamming distance d. Data is first encoded by the outer encoder to form a sequence of N Q-ary symbols. Using a one-to-one map from the alphabet of the outer code onto the inner code, the ith Q-ary symbol is mapped to a codeword $\mathbf{c}_i$ in the inner code. This results in a sequence $\mathbf{c}_1, \ldots, \mathbf{c}_N$ of Nn q-ary symbols, which we call the overall codeword.

The Nn q-ary symbols are then transmitted over a q-ary channel and may suffer from channel errors. The output of the channel is partitioned into N sequences $\mathbf{r}_1, \ldots, \mathbf{r}_N$, each having n q-ary symbols. Let e_i, where $i = 1, \ldots, N$, be the number of q-ary symbols that are erroneously received in the ith sequence $\mathbf{r}_i$.

Each sequence $\mathbf{r}_i$ is decoded according to the inner code using bounded distance decoding. The inner decoder either succeeds in decoding $\mathbf{r}_i$ into a codeword $\mathbf{c}_i'$ in the inner code that differs from the received sequence in $h_i \leq \lfloor (d-1)/2 \rfloor$ q-ary symbols, or it may fail in decoding the sequence, in which case we set $h_i = \lfloor (d+1)/2 \rfloor$. Let $\mathcal{I}'$, of size I', be the set of all indices i for which $h_i = \lfloor (d+1)/2 \rfloor$, i.e., for which the inner decoder fails to decode $\mathbf{r}_i$, and $\mathcal{I}''$, of size I'', be the set of all indices i for which $\mathbf{c}_i' \neq \mathbf{c}_i$, i.e., for which the inner decoder erroneously decodes $\mathbf{r}_i$. Clearly, the receiver knows the vector $\mathbf{h} = (h_1, \ldots, h_N)$ and consequently the set $\mathcal{I}' = \{i : h_i = \lfloor (d+1)/2 \rfloor\}$, but it does not know the set $\mathcal{I}''$ except that it is a subset of $\{1, \ldots, N\} \backslash \mathcal{I}'$. Based on this knowledge, the receiver chooses a set $\mathcal{L} \subseteq \{1, \ldots, N\}$ containing $\mathcal{I}'$. This set determines the input to the outer decoder which is a sequence of length N whose elements are either Q-ary symbols or erasures. If $i \notin \mathcal{L}$, the ith element of this sequence is a Q-ary symbol obtained by applying the inverse of the map used by the encoder to $\mathbf{c}_i'$. If $i \in \mathcal{L}$, the ith element in the input sequence to the outer decoder is an erasure. The outer decoder operates on the sequence of N Q-ary symbols/erasures using bounded

*The major part of this work was done while this author was a visiting researcher at the University of California in Davis, USA.

†This author was supported by NSF Grant CCR-96-12354.

distance decoding. We say that decoding is successful if the outer decoder succeeds in producing the Q-ary sequence of length N which is the output of the outer encoder. This is not the case if and only if

$$|\mathcal{L}| + 2|\mathcal{I}'' \backslash \mathcal{L}| \geq D. \tag{1}$$

Several erasing strategies have been proposed in literature (see, e.g., [2], [3]), varying from a simple rule as merely erasing a fixed number of the most unreliable symbols to more complex rules in which for each possible inner decoding result the erasure pattern is optimized according to some criterion. Another distinction which can be made is whether outer decoding is done only once (*single-trial*) or multiple times (*multi-trial*), where in the latter case the best among the various outer decoding results associated with different erasure patterns is chosen as the final decoding result, as in Forney's Generalized Minimum Distance (GMD) decoding algorithm [1].

In this paper, we focus on schemes with optimized erasing and single-trial decoding. Optimized erasing with a restricted number of trials has been considered by Kovalev [2] in the context of GMD decoding with reliability coefficients from the continuous interval [0,1]. The pattern of symbols to be erased was optimized with respect to minimizing the loss of realizable distance compared with the Forney algorithm. Here, the set of reliability coefficients is finite, and the pattern of symbols to be erased is chosen such that the number of q-ary errors in the concatenated codeword which is guaranteed to be corrected is maximum. Let $e(d, D)$ denote the minimum value of these maximized numbers, where the minimum is taken over all possible inner decoding results. Hence, $e(d, D)$ is the number of channel errors which is guaranteed to be corrected by the concatenated coding scheme. Section II provides an explicit expressions for $e(d, D)$.

II Results

Define $E(d, D, \mathbf{h}, \mathcal{L})$ to be the minimum number of channel errors $e_1 + \ldots + e_N$ that cause unsuccessful decoding for a given vector $\mathbf{h}$ and a given set $\mathcal{L}$. Next, define

$$E(d, D) = \min_{\mathbf{h}} \max_{\mathcal{L}} E(d, D, \mathbf{h}, \mathcal{L}), \tag{2}$$

where the minimum is over all vectors $\mathbf{h} = (h_1, \ldots, h_N)$ with $0 \leq h_1, \ldots, h_N \leq \lfloor (d+1)/2 \rfloor$. Note that $e(d, D) = E(d, D) - 1$. Clearly, in search for the minimum in (2), we can assume that $h_1 \geq \cdots \geq h_N$, which we will do in the rest of this paper. With this assumption, $\mathcal{I}' = \{i : 1 \leq i \leq I'\}$. Further, it is convenient to define for $0 \leq l \leq N$,

$$E(d, D, \mathbf{h}; l) = \sum_{i=l+1}^{\lceil (D+l)/2 \rceil} (d - 2h_i) + \sum_{i=1}^{N} h_i. \tag{3}$$

The next theorem, which is proved in the Appendix, provides a useful expression for $E(d, D)$.

Theorem 1 *For $d, D \geq 1$,*

$$E(d, D) = \min_{\mathbf{h}} \max_{\substack{0 \leq l \leq D-1 \\ l \equiv D-1 (2)}} E(d, D, \mathbf{h}; l) = \min_{\mathbf{h}} \max_{\substack{0 \leq l \leq D-1 \\ l \equiv D-1 (2)}} \sum_{i=l+1}^{(D+l+1)/2} (d - 2h_i) + \sum_{i=1}^{D} h_i,$$

Table I: Value of $E(d, D)$.

d, D	$E(d, D)$
$d \leq 2 \vee D \leq 2$	$\lceil Dd/2 \rceil$
$d = 3 \wedge D \geq 3$	$D + \lfloor D/3 \rfloor - \lfloor D/6 \rfloor + 1$
$d = 5 \wedge D \geq 3$	$3D + 1 - 2\lfloor D/2 \rfloor$
$d \geq 4 \wedge d \neq 5 \wedge D \equiv 3(4)$	$\lceil 3(D+1)d/8 \rceil$
$d \geq 4 \wedge d \neq 5 \wedge D = 4$	$\lceil 3d/2 \rceil + \lceil (d-1)/4 \rceil$
$d \geq 4 \wedge d \neq 5 \wedge D \equiv 0(4) \wedge D \geq 5$	$\lceil 3dD/8 \rceil + \lceil d/4 \rceil$
$d \equiv 0(2) \wedge d \geq 4 \wedge D \equiv 1(8) \wedge D \geq 5$	$(3D+5)d/8 - \lfloor d/3 \rfloor$
$d \equiv 1(2) \wedge d \geq 7 \wedge D \equiv 1(8) \wedge D \geq 5$	$(3D+5)d/8 - \lceil d/3 \rceil + 1$
$d \geq 4 \wedge d \neq 5 \wedge D \equiv 5(8) \wedge D \geq 5$	$\lceil (3D+5)d/8 \rceil - \lfloor d/3 \rfloor$
$d \geq 4 \wedge d \neq 5, 9 \wedge D \equiv 2(4) \wedge D \geq 5$	$\lceil (3D+2)d/8 \rceil$
$d = 9 \wedge D \equiv 2(4) \wedge D \geq 5$	$\lfloor 9(3D+2)/8 \rfloor + 1$

where the minimization is over all $\mathbf{h} = (h_1, \ldots, h_N)$ *such that* $d/2 \geq h_1 \geq \cdots \geq h_D \geq h_{D+1} = \cdots = h_N = 0$.

Based upon this result, we can now derive an explicit expression for $E(d, D)$, and thus for $e(d, D)$.

Theorem 2 *For* $d, D \geq 1$, *the value of* $E(d, D) = e(d, D) + 1$ *is as given in Table I.*

Proof. First, we will give the proof for the case $d, D \equiv 0(4)$. Let

$$\mathbf{h}^* = ((d/4)^{D/2+1}, 0^{N-D/2-1}),$$

where x^y denotes the vector $(x, x, \ldots, x)$ of length y. From (3), we have

$$E(d, D, \mathbf{h}^*; l) = \begin{cases} 3dD/8 + d/4 & \text{if } 1 \leq l \leq D/2+1,\ l \text{ odd}, \\ 3dD/8 + d/4 - (2l - D - 2)d/4 & \text{if } D/2+2 \leq l \leq D-1,\ l \text{ odd}. \end{cases}$$

In particular, it follows from Theorem 1 that

$$E(d, D) \leq 3dD/8 + d/4. \tag{4}$$

To show equality, let $\mathbf{h}$ be as given in Theorem 1 and suppose that

$$E(d, D, \mathbf{h}; l) \leq 3dD/8 + d/4 - 1 \tag{5}$$

for all l with $0 \leq l \leq D-1$ and l odd. Hence, it follows from (3) that

$$3dD/4 + d/2 - 2 \geq E(d, D, \mathbf{h}; 1) + E(d, D, \mathbf{h}; D/2+1) \geq 2h_1 + 3dD/4$$

which implies that $h_1 \leq d/4 - 1$. Since $h_1 \geq \cdots \geq h_D \geq 0$, we have from (3)

$$\begin{aligned} E(d, D, \mathbf{h}; 1) &= h_1 - \sum_{i=2}^{D/2+1} h_i + \sum_{i=D/2+2}^{D} h_i + dD/2 \geq (h_1 - h_2) - \sum_{i=3}^{D/2+1} h_i + dD/2 \\ &\geq -(D/2-1)(d/4-1) + dD/2 \geq 3dD/8 + d/4 \end{aligned}$$

Table II: Vector $\mathbf{h}^*$ such that $E(d, D, \mathbf{h}^*; l) \leq E(d, D)$ for all l. N.B. A vector $(x, x, \ldots, x)$ of length y is denoted as x^y; further, $a = 2\lfloor d/3 \rfloor - \lfloor d/2 \rfloor$, $b = 2\lfloor d/3 \rfloor - \lceil d/2 \rceil$, and $\mathbf{0}$ is an all-zero vector such that the length of $\mathbf{h}^*$ equals N.

d, D	$\mathbf{h}^*$
$d \leq 2 \vee D \leq 2$	$(\lfloor d/2 \rfloor^D, \mathbf{0})$
$d = 3 \wedge D \geq 3$	$(1^{\lfloor D/2 \rfloor + \lfloor D/3 \rfloor - \lfloor D/6 \rfloor + 1}, \mathbf{0})$
$4 \leq d \leq 5 \wedge D \geq 3$	$(1^{\lfloor D/2 \rfloor + 1}, \mathbf{0})$
$d \geq 6 \wedge D \equiv 1(4) \wedge D \geq 3$	$(\lfloor d/3 \rfloor^{(D+1)/2}, a^{\lceil (D-1)/8 \rceil}, b^{\lfloor (D-1)/8 \rfloor}, \mathbf{0})$
$d \geq 6 \wedge d \equiv 0(4) \wedge D \not\equiv 1(4) \wedge D \geq 3$	$((d/4)^{\lfloor D/2 \rfloor + 1}, \mathbf{0})$
$d \geq 6 \wedge d \equiv 1(4) \wedge D \not\equiv 1(4) \wedge D \geq 3 \wedge D \neq 4$	$(\lceil d/4 \rceil^{\lfloor D/2 \rfloor + 1}, 2^{\lfloor (D+5)/8 \rfloor}, 1^{\lfloor (D+1)/8 \rfloor}, \mathbf{0})$
$d \geq 6 \wedge d \equiv 1(4) \wedge D = 4$	$(((d-1)/4)^3, \mathbf{0})$
$d \geq 6 \wedge d \equiv 2(4) \wedge D \not\equiv 1(4) \wedge D \geq 3$	$(\lceil d/4 \rceil^{\lfloor D/2 \rfloor + 1}, 1^{\lfloor (D+1)/4 \rfloor}, \mathbf{0})$
$d \geq 6 \wedge d \equiv 3(4) \wedge D \not\equiv 1(4) \wedge D \geq 3$	$(\lceil d/4 \rceil^{\lfloor D/2 \rfloor + 1}, 1^{\lfloor (D+2)/8 \rfloor}, \mathbf{0})$

Table III: Set $\mathcal{S}$ containing for any $\mathbf{h}$ as given in Theorem 1 an l^* such that $E(d, D, \mathbf{h}; l^*) \geq E(d, D)$. N.B. $r = |\{i : h_i = 1\}|$ and $s = |\{i : h_i = 2\}|$.

d, D	$\mathcal{S}$
$d = 1, 2$	$\{r\}$
$d = 3$	$\{0, 1, r-2, r-1, r, r+1\}$
$d = 5$	$\{0, 1, s-1, s, s+1\}$
$d \neq 1, 2, 3, 5 \wedge D \equiv 0(4)$	$\{1, D/2+1\}$
$d \neq 1, 2, 3, 5 \wedge D \equiv 1(4)$	$\{0, 2, (D-1)/2, (D+3)/2\}$
$d \neq 1, 2, 3, 5, 9 \wedge D \equiv 2(4)$	$\{1, D/2\}$
$d = 9 \wedge D \equiv 2(4)$	$\{1, 3, D/2, D/2+2\}$
$d \neq 1, 2, 3, 5 \wedge D \equiv 3(4)$	$\{0, (D+1)/2\}$

contradicting (5). Hence, equality holds in (4), and so the theorem has been proved for the case $d, D \equiv 0(4)$.

Other cases can be proved similarly using the following guidelines. In Table II, we give for any $d, D \geq 1$ a vector $\mathbf{h}^*$ for which $E(d, D, \mathbf{h}^*; l)$ is at most equal to the indicated value of $E(d, D)$ for all l. In Table III, we give for any $d, D \geq 1$ a set $\mathcal{S}$ which contains for any $\mathbf{h}$ as given in Theorem 1 an element l^* such that $E(d, D, \mathbf{h}; l^*)$ is at least equal to the indicated value of $E(d, D)$. Combining the two results completes the proof. □

We can conclude that in this paper the number of errors, which is guaranteed to be corrected by a concatenated coding scheme with an outer code of distance D and an inner code of distance d, using single-trial bounded distance decoding and optimized erasing, has been fully determined. Note that for all $d, D \geq 1$ this number is roughly a fraction between $3/8$ and $1/2$ of the designed distance dD of the concatenated code, while a fixed erasing strategy as the single-trial version of the Zyablov method [3] typically realizes lower fractions, e.g., roughly $1/3$ for $D \geq 2$ and d large.

Appendix

In this appendix a proof of Theorem 1 is given.

Lemma 1 *For all* $d, D \geq 1$, $\mathbf{h}$, *and* $\mathcal{L}$,

$$E(d, D, \mathbf{h}, \mathcal{L}) = \sum_{i \in \hat{\mathcal{I}}''} (d - 2h_i) + \sum_{i=1}^{N} h_i,$$

where $\hat{\mathcal{I}}''$ *is the set composed of the smallest* $\max\{0, \lceil (D - |\mathcal{L}|)/2 \rceil\}$ *indices* i *that do not belong to* $\mathcal{L}$.

Proof. First, we make the following observations regarding the minimization of $e_1 + \cdots + e_N$ for given sets $\mathcal{I}'$ and $\mathcal{I}''$.

- If $i \in \mathcal{I}'$, then $e_i \geq h_i = \lfloor (d+1)/2 \rfloor$, where equality may hold.
- If $i \in \mathcal{I}''$, i.e., $\mathbf{c}_i \neq \mathbf{c}_i'$, then since the Hamming distance between $\mathbf{c}_i$ and $\mathbf{c}_i'$ is at least d, and the Hamming distances between $\mathbf{r}_i$ and $\mathbf{c}_i$ and between $\mathbf{r}_i$ and $\mathbf{c}_i'$ are e_i and $h_i \leq \lfloor (d-1)/2 \rfloor$, respectively, $e_i \geq d - h_i$, where equality may hold.
- If $i \notin \mathcal{I}' \cup \mathcal{I}''$, then $\mathbf{c}_i = \mathbf{c}_i'$ and $e_i = h_i \leq \lfloor (d-1)/2 \rfloor$.

Hence, for given $\mathbf{h}$, $\mathcal{I}''$, and $\mathcal{L}$ satisfying (1), the minimum number of errors $e_1 + \cdots + e_N$ that cause unsuccessful decoding is given by

$$\sum_{i \in \mathcal{I}''} (d - h_i) + \sum_{i \notin \mathcal{I}''} h_i = \sum_{i \in \mathcal{I}''} (d - 2h_i) + \sum_{i=1}^{N} h_i. \tag{6}$$

Since $d - 2h_i \geq 1$ for $i \in \mathcal{I}''$, the minimum of (6) over all choices of $\mathcal{I}''$ is achieved when $\mathcal{I}''$ is as small as possible subject to (1). In particular, $\mathcal{I}''$ should be disjoint from $\mathcal{L}$ and $I'' = \max\{0, \lceil (D - |\mathcal{L}|)/2 \rceil\}$. Since $h_1 \geq \cdots \geq h_N$, it follows that the minimum of (6) over all choices of $\mathcal{I}''$ is achieved when $\mathcal{I}''$ is equal to $\hat{\mathcal{I}}''$. □

Having found an expression for $E(d, D, \mathbf{h}, \mathcal{L})$, we proceed to maximize this expression over all choices of $\mathcal{L}$.

Lemma 2 *For all* $d, D \geq 1$ *and* $\mathbf{h}$,

a) $\max_{\mathcal{L}:|\mathcal{L}|=l} E(d, D, \mathbf{h}, \mathcal{L}) = E(d, D, \mathbf{h}; l)$ *for* $I' \leq l \leq N$,

b) $\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l) = \sum_{i=1}^{N} h_i$ *if* $I' \geq D$,

c) $\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l) = \max_{\substack{I' \leq l \leq D-1 \\ l \equiv D-1 (2)}} \sum_{i=l+1}^{(D+l+1)/2} (d - 2h_i) + \sum_{i=1}^{N} h_i$ *if* $I' \leq D - 1$.

Proof. Since $h_1 \geq \cdots \geq h_N$, the maximum of $E(d, D, \mathbf{h}, \mathcal{L})$ as given in Lemma 1, over all choices of $\mathcal{L}$ of fixed size l with $I' \leq l \leq N$, is achieved when the elements of the set $\hat{\mathcal{I}}''$ are as large as possible. This is accomplished by choosing $\mathcal{L} = \{i : 1 \leq i \leq l\}$. In this case, $\hat{\mathcal{I}}'' = \{i : l+1 \leq i \leq \lceil (D+l)/2 \rceil\}$, and thus

$$\max_{\mathcal{L}:|\mathcal{L}|=l} E(d, D, \mathbf{h}, \mathcal{L}) = \sum_{i=l+1}^{\lceil (D+l)/2 \rceil} (d - 2h_i) + \sum_{i=1}^{N} h_i, \tag{7}$$

which concludes the proof of part a). Since, for $l \geq I'$, the first sum in the right hand side of (7) is over $\max\{0, \lceil (D-l)/2 \rceil\}$ positive terms, this sum is maximized for some l such that $I' \leq l \leq \max\{I', D-1\}$. Thus, the first sum in the right hand side of (7) is maximized over all choices of $I' \leq l \leq N$ by $l = I'$ if $I' \geq D$ and by some l such that $I' \leq l \leq D-1$ if $I' \leq D-1$. In the latter case, we notice that if $l \equiv D(2)$, then increasing l by one does not decrease the first sum in the right hand side since $h_1 \geq \cdots \geq h_N$. Therefore, when maximizing the sum over l, we can assume that $l \equiv D-1(2)$. □

Next, we minimize the expression of $\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l)$ given in Lemma 2 b) and c) over all choices of $\mathbf{h}$ in order to obtain $E(d, D)$ in case d is odd or $D \geq 2$.

Lemma 3 *If d is odd or $D \geq 2$, then*

$$E(d, D) = \min_{\mathbf{h}} \max_{\substack{I' \leq l \leq D-1 \\ l \equiv D-1(2)}} \sum_{i=l+1}^{(D+l+1)/2} (d - 2h_i) + \sum_{i=1}^{D} h_i,$$

where the minimization is over all $\mathbf{h} = (h_1, \ldots, h_N)$ such that $d/2 \geq h_1 \geq \cdots \geq h_D \geq h_{D+1} = \cdots = h_N = 0$ and $h_D < d/2$.

Proof. First, suppose that $\mathbf{h} = (h_1, \ldots, h_N)$ is such that $I' = |\{i : h_i = \lfloor (d+1)/2 \rfloor\}| \geq D$ and define $\mathbf{g} = (g_1, \ldots, g_N)$ by

$$g_i = \begin{cases} h_i = \lfloor (d+1)/2 \rfloor & \text{for } 1 \leq i \leq D-2, \\ h_i - 1 = \lfloor (d-1)/2 \rfloor & \text{for } \max\{1, D-1\} \leq i \leq I', \\ h_i & \text{for } I'+1 \leq i \leq N. \end{cases} \tag{8}$$

Since $D \geq 2$ or d is odd, Lemma 2 gives

$$\begin{aligned} \max_{\max\{D-2,0\} \leq l \leq N} E(d, D, \mathbf{g}; l) &= d - 2\lfloor (d-1)/2 \rfloor + \sum_{i=1}^{N} g_i \\ &\leq \sum_{i=1}^{N} h_i = \max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l). \end{aligned}$$

This proves that the minimum value of $\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l)$ is attained by some $\mathbf{h}$ such that $I' \leq D-1$. For such $\mathbf{h}$, it follows from Lemma 2 that

$$\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l) = \max_{\substack{I' \leq l \leq D-1 \\ l \equiv D-1(2)}} \sum_{i=l+1}^{(D+l+1)/2} (d - 2h_i) + \sum_{i=1}^{N} h_i.$$

Note that for all l in the right hand side it holds that $(D+l+1)/2 \leq D$. Hence, to minimize this side over all vectors $\mathbf{h}$ with $0 \leq I' \leq D-1$, we should choose $h_i = 0$ for $D+1 \leq i \leq N$. This proves the lemma in case d is even. In case d is odd, which we assume now, we have to show that the minimum value of $\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l)$ is achieved by some $\mathbf{h}$ whose components are all less than $(d+1)/2$, i.e., $I' = 0$. Suppose that $\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l)$ attains its minimum value for some $\mathbf{h} = (h_1, \ldots, h_N)$ such that $1 \leq I' \leq D-1$. From the above, we can assume that $h_i = 0$ for $D+1 \leq i \leq N$. Define the vector $\mathbf{f} = (f_1, \ldots, f_N)$ by

$$f_i = \begin{cases} h_i - 1 = (d-1)/2 & \text{for } 1 \leq i \leq I', \\ h_i & \text{for } I'+1 \leq i \leq N. \end{cases}$$

Lemma 2 implies that $E(d, D, \mathbf{f}; l)$ is maximized by some l such that $0 \leq l \leq D-1$ and $l \equiv D-1(2)$. Let l_1 be such an l achieving the maximum. Define

$$l_2 = \begin{cases} l_1 & \text{if } l_1 \geq I', \\ I' & \text{if } l_1 < I' \text{ and } I' \equiv D-1(2), \\ I'+1 & \text{if } l_1 < I' \text{ and } I' \equiv D(2). \end{cases} \tag{9}$$

From Lemma 2 and (3) it follows that

$$\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l) \geq E(d, D, \mathbf{h}; l_2) = \sum_{i=l_2+1}^{(D+l_2+1)/2} (d - 2h_i) + \sum_{i=1}^{D} h_i \tag{10}$$

and

$$\begin{aligned} \max_{0 \leq l \leq N} E(d, D, \mathbf{f}; l) &= E(d, D, \mathbf{f}; l_1) = \sum_{i=l_1+1}^{(D+l_1+1)/2} (d - 2f_i) + \sum_{i=1}^{D} f_i \\ &= \sum_{i=l_1+1}^{\min\{I', (D+l_1+1)/2\}} (d - 2f_i) + \sum_{i=\max\{l_1, I'\}+1}^{(D+l_1+1)/2} (d - 2f_i) + \sum_{i=1}^{D} f_i \\ &= \max\{0, \min\{I', (D+l_1+1)/2\} - l_1\} \\ &\quad + \sum_{i=\max\{l_1, I'\}+1}^{(D+l_1+1)/2} (d - 2h_i) - I' + \sum_{i=1}^{D} h_i, \end{aligned}$$

which implies

$$\max_{0 \leq l \leq N} E(d, D, \mathbf{f}; l) \leq \sum_{i=\max\{l_1, I'\}+1}^{(D+l_1+1)/2} (d - 2h_i) + \sum_{i=1}^{D} h_i. \tag{11}$$

From (10), (11), and (9) we have

$$\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l) - \max_{0 \leq l \leq N} E(d, D, \mathbf{f}; l) \geq \sum_{i=l_2+1}^{(D+l_2+1)/2} (d-2h_i) - \sum_{i=\max\{l_1, I'\}+1}^{(D+l_1+1)/2} (d-2h_i) \geq 0.$$

Thus, we conclude that in minimizing $\max_{I' \leq l \leq N} E(d, D, \mathbf{h}; l)$ over all $\mathbf{h}$ we can assume that none of the components of $\mathbf{h}$ equal $(d+1)/2$. □

Next, the expression of $E(d, D)$ given in Lemma 3 is simplified. Furthermore, the case d is even and $D = 1$ is included.

Lemma 4 *For $d, D \geq 1$,*

$$E(d,D) = \min_{d/2 \geq h_1 \geq \cdots \geq h_D \geq 0} \max_{\substack{0 \leq l \leq D-1 \\ l \equiv D-1 (2)}} \sum_{i=l+1}^{(D+l+1)/2} (d - 2h_i) + \sum_{i=1}^{D} h_i.$$

Proof. In case d is odd, we know from Lemma 3 that $\max_{I' \leq l \leq N} E(d,D,\mathbf{h};l)$ is minimized by some $\mathbf{h} = (h_1, \ldots, h_N)$ such that $h_i \leq (d-1)/2$ for all i. Hence, $I' = |\{i : h_i = \lfloor (d+1)/2 \rfloor\}| = 0$ for such an $\mathbf{h}$ and the result follows immediately.

Next, we consider the case d is even and $D \geq 2$. It follows from Lemma 3 that

$$E(d,D) \leq \min_{\substack{d/2 \geq h_1 \geq \cdots \geq h_D \geq 0 \\ h_D < d/2}} \max_{\substack{0 \leq l \leq D-1 \\ l \equiv D-1 (2)}} \sum_{i=l+1}^{(D+l+1)/2} (d - 2h_i) + \sum_{i=1}^{D} h_i. \tag{12}$$

To show equality, we will show that for all $\mathbf{h}$ with $d/2 \geq h_1 \geq \cdots \geq h_D \geq h_{D+1} = \cdots = h_N = 0$ and $h_D < d/2$ and $0 \leq l \leq I' - 1$,

$$E(d,D,\mathbf{h};l) \leq E(d,D,\mathbf{h};l'), \tag{13}$$

where $I' \leq l' \leq D-1$ is such that $E(d,D,\mathbf{h};l') = \max_{I' \leq l \leq D-1} E(d,D,\mathbf{h};l)$. Clearly, for such an l,

$$\begin{aligned} \sum_{i=l+1}^{\lceil (D+l)/2 \rceil} (d - 2h_i) &= \sum_{i=l+1}^{\min\{\lceil (D+l)/2 \rceil, I'\}} (d - 2h_i) + \sum_{i=I'+1}^{\lceil (D+l)/2 \rceil} (d - 2h_i) \\ &= \sum_{i=I'+1}^{\lceil (D+l)/2 \rceil} (d - 2h_i) \leq \sum_{i=I'+1}^{\lceil (D+I')/2 \rceil} (d - 2h_i) \leq \sum_{i=l'+1}^{\lceil (D+l')/2 \rceil} (d - 2h_i), \end{aligned}$$

and (13) follows from (3). Hence, equality holds in (12). The result now follows by observing that the restriction $h_D < d/2$ in the minimization can be omitted, since for all $0 \leq l \leq D-1$ with $l \equiv D-1(2)$ and $\mathbf{h}$ with $d/2 = h_1 = \cdots = h_D > h_{D+1} = \cdots = h_N = 0$, we have $E(d,D,\mathbf{h};l) \geq E(d,D,\mathbf{g};l)$, where $\mathbf{g} = (g_1, \ldots, g_N)$ is as defined in (8).

Finally, in case d is even and $D = 1$, it follows from (2) and Lemma 2 that

$$\begin{aligned} E(d,1) &= \min_{\mathbf{h}} \max_{I' \leq l \leq N} E(d,1,\mathbf{h};l) = \min\{\min_{\mathbf{h}:I' \geq 1} \sum_{i=1}^{N} h_i, \min_{\mathbf{h}:I'=0} d - h_1 + \sum_{i=2}^{N} h_i\} \\ &= \min\{d/2, d/2+1\} = d/2 = \min_{d/2 \geq h_1 \geq 0} d - h_1, \end{aligned}$$

which completes the proof. □

From the above, we can conclude Theorem 1.

References

[1] G.D. Forney, Jr., *Concatenated Codes.* Cambridge, MA: M.I.T. Press, 1966.

[2] S.I. Kovalev, "Two classes of minimum generalized distance decoding algorithms," *Problemy Peredachi Inform.*, vol. 22, no. 3, pp. 35-42, 1986.

[3] V.V. Zyablov, "Optimization of concatenated decoding algorithms," *Problemy Peredachi Inform.*, vol. 9, no. 1, pp. 26-32, 1973.

CODING FOR RANDOM ACCESS COMMUNICATIONS

Presented at 1999 WIC meeting at the University of Leuven, Belgium

A.J. Han Vinck

University of Essen, Essen, Germany
Vinck@exp-math.uni-essen.de

SUMMARY

We discuss the transmission of information over a multi-user channel where the set of active users is small compared to the total amount of potential users. The channel inputs are integers and the channel output at each time instant is a symbol which identifies which subset of integers occurred as inputs to the channel, but not how many of each integer occurred. The so called T-user M-frequency noiseless multiple access channel without intensity information, as defined by Chang and Wolf [2] is an example of such a channel. As another practical example, in Pulse Positioning Modulation (PPM) format each integer is transmitted as a single pulse positioned in one of q disjoint subslots. The detector output after each slot is equal to the positions where a pulse is detected. Hence, for a q-ary input we have 2^q-1 possible outputs. This channel model is equivalent to the T-User M-Frequency Multi Access Channel. It is the purpose of this paper to describe a signaling method that allows m users to use the q-ary input channel simultaneously.

We first extend and modify the class of binary Superimposed Codes (SIC) introduced by Kautz-Singleton [1]. A binary Superimposed code SIC(q, N, 2, m) consists of N binary code words of length q, with the property that from the Boolean sum of any m-subset we are able to uniquely determine the individual code words from the m-subset see also [3]. We extend the definition of binary SICs to the situation where code words have q-ary symbols and the channel output is a symbol which identifies which subset of integers occurred as input to the channel (no intensity information).

Secondly we give a transmission method for random access using the "optical OR" channel with "random" signature sequences. Asymptotically, this method achieves the maximum obtainable throughput with vanishing error probability. We give new lower bounds for the capacity for $T \to \infty$ and discuss results of numerical calculations of the capacities for various values of T.

REFERENCES

[1] W.H. Kautz and R.C. Singleton, "Nonrandom Binary Superimposed Codes," IEEE Trans. on Inform. Theory, IT-10, October 1964, pp. 363-377.
[2] Shin-Chun Chang and J.K. Wolf, "On the T-User M-Frequency Noiseless Multiple Access Channel with and without Intensity Information," IEEE Trans. on Inform. Theory, Vol. IT-27, January 1981, pp. 41-48.
[3] A.G. Dyachkov and V.V. Rykov, "A Survey of Superimposed Code Theory," Problems of Control and Information Theory, Vol. 12, No. 4, pp. 1-13, English Translation.

List of Participants

V. Balakirsky
T.U. Eindhoven
Dept. EE
Tuinkersweg 12
5632 KS Eindhoven - Nederland
Tel. 31/40/247.56.81
Email v.b.balakirsky@ele.tue.nl

A. Barbé
K.U.Leuven
Elektrotechniek, SISTA
Kardinaal Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.10.53
Fax. 32/16/32.19.70
Email andre.barbe@esat.kuleuven.ac.be

M.S. Bargh
T.U. Eindhoven
EH 11.16
Posbus 513
5600 MB Eindhoven - Nederland
Tel. 31/40/247.36.52
Email mortaza@ei.ele.tue.nl

J. Biemond
Delft University of Technology
ICT-group
PO Box 5031
2600 GA Delft - Nederland
Tel. 31/152/78.46.95
Email j.biemond@et.tudelft.nl

I. Bruyland
RUG-TELIN
Gentsesteenweg 108
9300 Aalst - België
Tel. 32/9/264.34.10
Email ignace.bruyland@rug.ac.be

J. Cardinal
ULB
Campus Plaine, Bld. Du Triomphe, CP 212
1050 Brussel - België
Tel. 32/2/650.56.01
Email jcardin@ulb.ac.be

J. Claessens
K.U.Leuven
Elektrotechniek, COSIC
Kardinaal Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.18.53
Fax. 32/16/32.19.69
Email joris.claessens@esat.kuleuven.ac.be

P.H.N. de With
University Mannheim
B6.26
68131 Mannheim - Germany
Tel. 49/621/292.1650
Email dewith@ti.uni-mannheim.de

B. DeKnuydt
K.U.Leuven
Elektrotechniek, PSI
Kard. Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.18.80
Fax. 32/16/32.17.23
Email deknuydt@esat.kuleuven.ac.be

K. Demuynck
K.U.Leuven
Elektrotechniek, PSI
Kard. Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.18.60
Fax. 32/16/32.17.23
Email kris.demuynck@esat.kuleuven.ac.be

J. Doumen
T.U. Eindhoven
HG 9.87
Postbus 90159
5600 MB Eindhoven - Nederland
Tel. 31/40/247.50.16
Email doumen@win.tue.nl

A. Gerrits
Philips Research
Prof. Holstlaan 4
5656 AA Eindhoven - Nederland
Tel. 31/40/274.41.74
Email gerritsa@natlab.research.philips.com

Z. Goey
Eindhoven University of Technology
Batterijstraat 23b
6211 SE Maastricht - Nederland
Email zoe@win.tue.nl

G.H.L.M. Heideman
Universiteit Twente
TIOS
Postbus 217
7500 AE Enschede - Nederland
Email heideman@cs.utwente.nl

C.J.A. Jansen
Philips Crypto B.V.
De Witbogt 2, P.O. Box 218
5600 MD Eindhoven - Nederland
Tel. 31/40/2722.600
Email c.jansen@crypto.philips.com

D. Kleima
T.U. Twente
Fac. EL
Postbus 217
7500 AE Enschede - Nederland
Email dkleima@el.utwente.nl

F. Koeune
UCL
Crypto Group - DICE
Place du Levant 3
1348 Louvain-la-Neuve - België
Email fkoeune@dice.ucl.ac.be

A. Koppelaar
Philips Research
Prof. Holstlaan 4
5656 AA Eindhoven - Nederland
Tel. 31/40/274.22.26
Email koppelr@natlab.research.philips.com

L. Kovács
Technical University of Budapest
Stoczek u.2
1111 Budapest - Hunagary
Tel. 36/1/463.35.47
Email kovacsl@hit.bme.hu

E. Lerouge
K.U.Leuven
Elektrotechniek, SISTA
Kardinaal Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.18.84
Fax. 32/16/32.19.70
Email ellen.lerouge@esat.kuleuven.ac.be

J. Levendovszky
Technical University of Budapest
Stoczek u.2
1111 Budapest - Hunagary
Tel. 36/1/463.35.47
Email levendov@hit.bme.hu

H. Massias
UCL
Crypto Group - DICE
Place du Levant 3
1348 Louvain-la-Neuve - België
Email massias@dice.ucl.ac.be

S. Mietens
University Mannheim
B6.26
68131 Mannheim - Germany
Tel. 49/621/292.5643
Email mietens@ti.uni-mannheim.de

F. Mindru
K.U.Leuven
Elektrotechniek, PSI
Kard. Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.17.18
Email florica.mindru@esat.kuleuven.ac.be

R. Moddemeijer
University of Groningen
P.O. Box 800
9700 AV Groningen - Nederland
Tel. 31/50/363.39.40
Email rudy@cs.rug.nl

T. Moons
K.U.Leuven
Elektrotechiek, PSI
Kard. Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.17.02
Fax. 32/16/32.17.23
Email theo.moons@esat.kuleuven.ac.be

J. Nakahara
K.U.Leuven
Elektrotechniek, COSIC
Kard. Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.10.73
Fax. 32/16/32.19.69
Email jorge.nakahara@esat.kuleuven.ac.be

J.B.H. Peek
Univ. Nijmegen
Computing Science Dept.
Toernooiveld 1
6525 ED Nijmegen - Nederland
Tel. 31/40/283.22.49
Email hpeek@cs.kun.nl

V.V. Prelov
Inst. for Inform. Trans. Probl. of the RAS
Moscow - Russia
Email prelov@iitp.ru

J.P.M. Schalkwijk
T.U. Eindhoven
Dept. EE
5600 MB Eindhoven - Nederland
Tel. 31/40/247.35.15
Email piet@ei.ele.tue.nl

C.H. Slump
University of Twente
Electrical Engineering
PO Box 217
7500 AE Enschede - Nederland
Email c.h.slump@el.utwente.nl

D. Therssen
Philips ITCL C.E.
Interleuvenlaan 74-76
3000 Leuven - België
Tel. 32/16/39.06.56
Email dieter.therssen@leu.ce.philips.com

L. Tolhuizen
Philips Nat. Lab.
Prof. Holstlaan 4
5656 AA Eindhoven - Nederland
Tel. 31/40/274.20.69
Email tolhuizen@natlab.research.philips.com

E.C. van der Meulen
K.U.Leuven
Toegepaste Wiskunde
Celijnenlaaan 200B
3001 Heverlee - België
Tel. 32/16/32.70.47
Fax. 32/16/32.70.15
Email ecvdm@gauss.wis.kuleuven.ac.be

M. van Dijk
Philips Research
Prof. Holstlaan 4
5656 AA Eindhoven - Nederland
Tel. 31/40/274.37.96
Email mvandijk@natlab.research.philips.com

C.L.M. van Pul
Philips Crypto B.V.
De Witbogt 2, P.O. Box 218
5600 MD Eindhoven - Nederland
Tel. 31/40/2722.600
Email c.v.pul@crypto.philips.com

B. Van Rompay
K.U.Leuven
Elektrotechniek, COSIC
Kardinaal Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.18.62
Fax. 32/16/32.19.69
Email bart.vanrompay@esat.kuleuven.ac.be

J. Vandewalle
K.U.Leuven
Elektrotechniek, SISTA
Kardinaal Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.10.52
Fax. 32/16/32.19.70
Email joos.vandewalle@esat.kuleuven.ac.be

S. Vangheluwe
K.U.Leuven
Dept. Wiskunde
Celestijnenlaan 200B
3001 Heverlee - België
Tel. 32/16/32.70.53
Fax. 32/16/32.79.98
Email saskia.vangheluwe@wis.kuleuven.ac.be

P. Vanroose
K.U.Leuven
Elektrotechniek, PSI
Kardinaal Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.17.07
Fax. 32/16/32.19.86
Email peter.vanroose@esat.kuleuven.ac.be

A.J.H. Vinck
University of Essen
Ellernstr. 29
45326 Essen - Germany
Email vinck@exp-math.uni-essen.de

P. Volf
T.U. Eindhoven
EH 11.16
Postbus 513
5600 MB Eindhoven - Nederland
Tel. 31/40/247.35.52
Email paul@ei.ele.tue.nl

F. Von Haeseler
K.U.Leuven
Elektrotechniek, SISTA
Kardinaal Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.18.17
Fax. 32/16/32.19.70
Email fvanhaes@esat.kuleuven.ac.be

P. Wambacq
K.U.Leuven
Elektrotechniek, PSI
Kard. Mercierlaan 94
3001 Heverlee - België
Tel. 32/16/32.10.57
Email patrick.wambacq@esat.kuleuven.ac.be

J.H. Weber
T.U. Delft
Fac. ITS
Postbus 5031
2600 GA Delft - Nederland
Email j.h.weber@its.tudelft.nl

F. Willems
T.U. Eindhoven
Dept. EE
Postbus 513
5600 MB Eindhoven - Nederland
Tel. 31/40/247.35.39
Email f.m.j.willems@ele.tue.nl

S. Xu
T.U. Eindhoven
HG 9.56
Postbus 513
5600 MB Eindhoven - Nederland
Tel. 31/40/247.43.46
Email sbxu@win.tue.nl