ELECTRONICS LETTERS

AN INTERNATIONAL PUBLICATION

CONTENTS

pages 1 - 96

7th January 1999 Vol. 35 No. 1

(continued on back cover)

CONTENTS

(continued from front cover)

(continued on inside back cover)

ELECTRONICS LETTERS

THE INSTITUTION OF ELECTRICAL ENGINEERS

7 JANUARY 1999
ELLEAK 35 (1)

VOLUME 35
1 – 96

NUMBER 1
ISSN 0013-5194

ELECTRONICS LETTERS (ISSN 0013-5194) is published every other week except for one issue in December (total of 25 issues). 1999 annual subscription price £630.00. Single copy £26.00. Available also in a concurrent microfiche edition at the same subscription rate as for the printed edition. Air mail service direct to subscribers an additional £62.00.

All subscription inquiries and orders should be sent to IEE Publication Sales, PO Box 96, Stevenage SG1 2SD, United Kingdom.

Contributions should be in accordance with the Guide to Authors, which is usually to be found on the inside back cover or the last page of each issue, and should be addressed as indicated in the Guide.

All other correspondence, including advertisements, should be sent to the IEE Publishing Department.

The advantages of rapid publication given by *Electronics Letters* are nullified if subscribers are subject to excessive delays in the receipt of the journal. The Executive Editor would be grateful to learn of any difficulties subscribers may experience.

Electronics Letters Online

Electronics Letters is available online through the IEE Online Journals service on the IEE's World-Wide Web server. Articles are stored in PDF format and may be viewed and printed using the Adobe® Acrobat® Reader. Features include: browsable table of contents pages for each journal issue, cross-journal searching of bibliographic records (inc. authors, titles, abstracts), and the ability to print articles locally. The issues are added in advance of print publication. For further information, please look at http://ioj.iee.org.uk/ or contact the Marketing Department at Stevenage.

The Institution of Electrical Engineers
Savoy Place
London WC2R 0BL
United Kingdom

Telephone +44 (0)1 71 240 1871
Facsimile +44 (0)1 71 240 7735
WWW http://www.iee.org.uk

Publishing Department:
IEE
Michael Faraday House
Six Hills Way
Stevenage SG1 2AY
United Kingdom
Telephone +44 (0) 1438 313311
Telex 825578 IEESTV G
Facsimile:
Sales +44 (0) 1438 742792
Editorial +44 (0) 1438 742849
Advertising +44 (0) 1438 318361
Marketing +44 (0) 1438 742840
E-mail:

Secretary of the IEE:
J. C. Williams, OBE, PhD, FEng, FIEE

Editorial staff:
Gill Wheeler (Managing Editor)
Joanne Davies
Rachel Smyly

Typeset in England by
Pindar plc
Ryedale Building
First Floor
60 Piccadilly
York YO1 9WU

HONORARY EDITORS
Sir Eric Ash, CBE, FEng, FRS
Prof. Peter Clarricoats, CBE, FEng, FRS
Prof. Chris Toumazou

Printed in UK by
A. McLay & Co. Limited
Longwood Drive
Forest Farm
Cardiff CF4 7ZB

Security analysis: Some possible attacks against the proposed scheme are presented below.

Attack 1: Although the group authority has the knowledge of (r_i, s_i, k_i), the group signature cannot be forged without the secret key x_i of U_i. It is impossible to obtain x_i from y_i without being able to solve the discrete logarithm problem. Moreover, because the generator $\alpha_i = g^{a \cdot k_i} \bmod p$, $k_i \in Z_q^*$ and $a \in Z_q^*$, both α_i and g have the same order q. Therefore, forging (R, S) is as difficult as breaking ElGamal's scheme [3]. Since the group authority cannot forge the group signature, forgery by an adversary is even more difficult. Thus, the impersonation attack can not be successful.

Attack 2: The signer U_i can be identified if we can obtain y_i from the signature $\{R, S, h(m), A, B, C, D, E\}$. Since the receiver does not know the (r_i, s_i, k_i) of the group authority, he cannot check the equation $D^B \cdot y_I^C \cdot E \equiv D^{k_I} \bmod p$. Obtaining ($r_i$, s_i, k_i) from given $\{A, B, C, D, E\}$ depends on the discrete logarithm.

Attack 3: The group authority may publish the information (r_T, s_T, y_i) for the message m's signature to enable a verifier to check the identity of U_i. This does not damage the anonymity of U_i's previous group signatures because the information (r_T, s_T, y_i) is only provided for the specific group signature $\{R, S, h(m), A, B, C, D, E\}$. For different messages, U_i will have chosen different random integers a and b to generate group signatures. If an adversary wants to obtain a, b and (r_i, s_i) from given $\{A, B, C, D, E\}$,this is as difficult as solving the discrete logarithm.

Discussion: The improved scheme preserves the main merits inherent in most of the Lee-Chang scheme. In the case of a later dispute, the group authority may publish the information (r_T, s_T, y_i) to enable a verifier to check the identity of the signer, although this does not damage the anonymity of the other previous signatures of the signer. Meanwhile, the group authority need not renew any key of the signer. The reason is that the information (r_T, s_T, y_i) is only provided for the specific group signature $\{R, S, h(m), A, B, C, D, E\}$. Compared to the original scheme, the improved scheme requires some additional cost in terms of computational time and the size of the group signature. For generating a group signature, the signer U_i may precompute several different $\{\alpha_i, A, B, C, D, E\}$ using (r_i, s_i) to reduce the real-time computational time.

Conclusions: We have proposed an improved group signature scheme based on the discrete logarithm. In our improvement, a group signature can be opened to reveal the identity of the signer, the anonymity of the other previous signatures signed by this group member are not damaged. Meanwhile, the group authority also need not renew the keys of the signer. We have demonstrated some possible attacks against the proposed scheme. Under the difficulty of computing the discrete logarithm problem, we have shown that the improved scheme is secure against these attacks.

28 October 1998
Electronics Letters Online No: 19990071

Yuh-Min Tseng and Jinn-Ke Jan (*Institute of Applied Mathematics, National Chung Hsing University, Taichung, Taiwan 402, Republic of China*)

Jinn-Ke Jan: corresponding author

E-mail: jkjan@amath.nchu.edu.tw

References

1 CHAUM, D., and HEYST, F.: 'Group signatures', *Proc. EUROCRYPT'91*, 1992, pp. 257–265

2 CHEN, L., and PEDERSEN, T.P.: 'New group signature schemes', *Proc. EUROCRYPT'94*, 1995, pp. 171–181

3 ELGAMAL, T.: 'A public key crypto system and a signature scheme based on discrete logarithms', *IEEE Trans. Inf. Theory*, 1985, **31**, (4), pp. 469–472

4 LEE, W.B., and CHANG, C.C.: 'Efficient group signature scheme based on the discrete logarithm', *IEE Proc. Comput. Digit. Tech.*, 1998, **145**, (1), pp. 15–18

5 NYBERG, K., and RUEPPEL, R.A.: 'Message recovery for signature schemes based on the discrete logarithm problem', *Designs, Codes*

Low density parity check codes with semi-random parity check matrix

Li Ping, W.K. Leung and Nam Phamdo

A semi-random approach to low density parity check code design is shown to achieve essentially the same performance as an existing method, but with considerably reduced complexity.

Introduction: Recently, there has been revived interest in the low density parity check (LDPC) codes originally introduced in 1962 by Gallager [1]. It has been shown that such codes can achieve very good performances (within 1.5dB of theoretical limits) with modest decoding complexity [2].

An LDPC code is defined from a randomly generated parity check matrix $\mathbf{H}$ [2]. For the purpose of encoding, it is necessary to transfer $\mathbf{H}$ into $\mathbf{H}_{sys}$, the equivalent systematic form of $\mathbf{H}$, which can be accomplished by Gaussian elimination. For a rate $R = k/n$ (k = information length, n = coded length), the size of $\mathbf{H}$ is $(n–k) \times n$. When n is large, Gaussian elimination can be costly in terms of both memory and the operations involved. Besides, a considerable amount of memory is required to store $\mathbf{H}_{sys}$ in the encoder, which is not necessarily sparse even though $\mathbf{H}$ is usually designed so.

In this Letter, we report a modified approach to LDPC code design. We adopt a semi-random technique, i.e. only part of $\mathbf{H}$ is generated randomly, and the remaining part is deterministic. The new method can achieve essentially the same performance as the standard LDPC encoding method with significantly reduced complexity.

Proposed approach: For simplicity we will only consider binary codes. Decompose the codeword $\mathbf{c}$ as $\mathbf{c} = [\mathbf{p}, \mathbf{d}]^t$, where $\mathbf{p}$ and $\mathbf{d}$ contain the parity and information bits, respectively. Accordingly, we decompose H into $\mathbf{H} = [\mathbf{H}^p, \mathbf{H}^d]$. Then

$$(\mathbf{H}^p, \mathbf{H}^d)\begin{pmatrix}\mathbf{p}\\ \mathbf{d}\end{pmatrix} = \mathbf{0} \tag{1}$$

In the proposed method, $\mathbf{H}^p$ is constructed in some deterministic form. Empirically, we found the following a good choice (recall that $\mathbf{H}^p$ must be a square matrix [3]):

$$\mathbf{H}^p = \begin{pmatrix} 1 & & & 0 \\ 1 & 1 & & \\ & \ddots & \ddots & \\ 0 & & 1 & 1 \end{pmatrix} \tag{2}$$

We adopt the following rules to create $\mathbf{H}^d$. Let t be a preset integer constrained by (*i*) t divides n-k and (ii) n-k divides kt. Partition $\mathbf{H}^d$ (which has $n–k$ rows) into t equal sub-blocks as

$$\mathbf{H}^d = \begin{pmatrix} \mathbf{H}^{d1} \\ \vdots \\ \mathbf{H}^{dt} \end{pmatrix} \tag{3}$$

In each sub-block $\mathbf{H}^{di}$, $i = 1, 2 \cdots t$, we randomly create exactly one element 1 per column and $kt/(n$-$k)$ 1s per row. The partition in eqn. 3 is to best increase the recurrence distance of each bit in the encoding chain (see below) and, intuitively, reduces the correlation during the decoding process. The resultant $\mathbf{H}^d$ has a column weight of t and a row weight of $kt/(n–k)$ (the weight of a vector is the number of 1s among its elements).

Based on eqns. 1 and 2, $\mathbf{p} = \{p_i\}$ can easily be calculated from a given $\mathbf{d} = \{d_i\}$ as

$$p_1 = \sum_j h_{1,j}^d d_j \quad \text{and} \quad p_i = p_{i-1} + \sum_j h_{i,j}^d d_j \pmod 2 \tag{4}$$

Compared with the standard LDPC code design [2], the above method has several advantages. First, the encoding process in eqn. 4 is much simpler than a full Gaussian elimination. Secondly, a random $\mathbf{H}^d$ can be singular, which causes additional programming complexity in realising a specified rate. On the other hand, $\mathbf{H}^p$ in eqn. 2 is always non-singular so the new method can realise any given rate directly and precisely. Thirdly, it requires very little memory to store $\mathbf{H}^d$ in the encoder if $\mathbf{H}^d$ is sparse (this can be

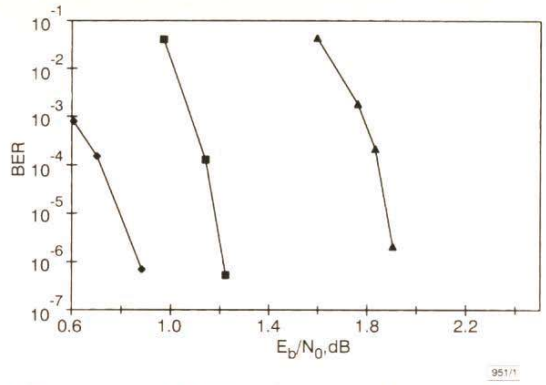

Fig. 1 *Performances of LDPC codes generated by semi-random parity check matrixes with k = 30000*

◆ $R = 1/3$
■ $R = 1/2$
▲ $R = 2/3$

Simulation study: Fig. 1 contains the simulated performances of the proposed encoding method for various rates (1/3, 1/2, 2/3) using $t = 4$. The decoding algorithm follows that in [2]. The results are essentially the same as those obtained using fully random **H**.

Conclusion: It has been shown that a semi-random approach to LDPC code design can achieve essentially the same performance as the existing method with considerably reduced complexity.

23 November 1998
Electronics Letters Online No: 19990065

Li Ping and W.K. Leung (*Department of Electronic Engineering, City University of Hong Kong, Hong Kong*)

E-mail: eeliping@cityu.edu.hk

Nam Phamdo (*Department of Electrical and Computer Engineering, State University of New York at Stony Brook, Stony Brook, NY 11794-2350, USA*)

References

1 GALLAGER, R.G.: 'Low density parity check codes', *IRE Trans. Inf. Theory*, 1962, **IT-8**, pp. 21–28
2 MacKAY, D.J.C., and NEAL, R.M.: 'Near Shannon limit performance of low density parity check codes', *Electron. Lett.*, 1997, **33**, (6), pp. 457–458
3 PROAKIS, J.G.: 'Digital communications' (McGraw-Hill, 1995)
4 PETERSON, W.W., and WELDON, E.J., Jr.: 'Error-correcting codes' (MIT Press, Cambridge, Massachusetts, 1972) 2nd edn.

Non-binary convolutional codes for turbo coding

C. Berrou and M. Jézéquel

The authors consider the use of non-binary convolutional codes in turbo coding. It is shown that quaternary codes can be advantageous, both from performance and complexity standpoints, but that higher-order codes may not bring further improvement.

Introduction: Turbo codes are error correcting codes with at least two dimensions (i.e. each datum is encoded at least twice). The decoding of turbo codes is based on an iterative procedure using the concept of extrinsic information. Fig. 1 gives an example of a two-dimensional turbo code built from a parallel concatenation of two identical recursive systematic convolutional (RSC) codes with generators 15, 13 (octal notation). The global (non-iterative) decoding of such a code is too complex to be envisaged because of the very large number of states induced by the interleaver. An iterative procedure is therefore used, the two codes being decoded decoders passing the result of their work to each other, at each iterative step.

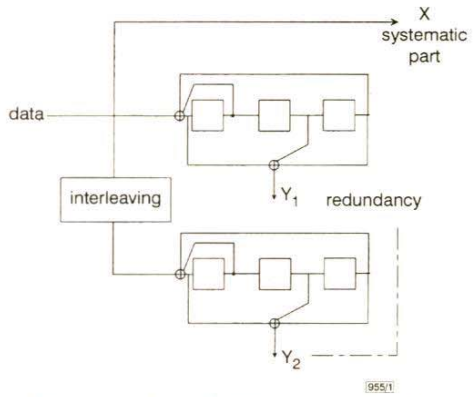

Fig. 1 *Two-dimension turbo code with generators 15, 13*

Binary codes versus quaternary codes: Fig. 2*a* represents a block of size k encoded by the code of Fig. 1. This block is seen as a two-dimensional $\sqrt{k} \times \sqrt{k}$ block and for simplicity we consider that the interleaver is a regular one: the sequence is encoded first by C_1, following the horizontal or linewise dimension, and secondly by C_2, following the vertical or columnwise dimension. The dashes on both dimensions symbolise the path error packets at the output of the two decoders, at a particular step of the iterative process. These packets do not contain only erroneous decisions but they indicate where a wrong path has been chosen either by the decoder of C_1 or by the decoder of C_2. This corresponds to a certain path error density per dimension, which is the same in both dimensions if the component codes are identical.

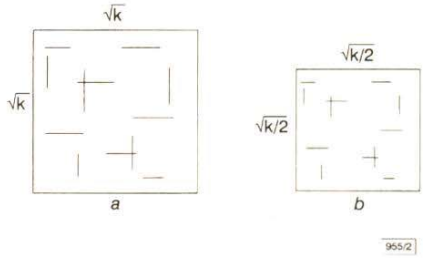

Fig. 2 *Path error packets in turbo decoding*

a Binary codes
b Quaternary codes

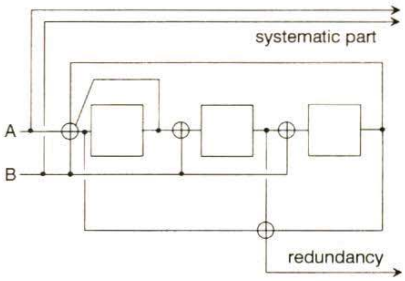

Fig. 3 *8-state quaternary recursive systematic convolutional (RSC) code with generators 15, 13*

The performance of turbo decoding is strongly dependent on the path error density per dimension. Obviously, the more numerous and the longer the horizontal and vertical dashes in the square box are, the harder the convergence to the correct codeword is. Now for each component code, replace the binary code of Fig. 1 by the quaternary code of Fig. 3. The data are thus encoded and interleaved in couples. The size of the block is $k/2$ couples and the square box now has the dimensions $\sqrt{k/2} \times \sqrt{k/2}$ (Fig. 2*b*). When a decoder selects a path in the decoding trellis, the same amount of information is used in the cases of both binary and quaternary