

Appendix 1 contains an iterative algorithm which is used to design the encoder and decoder structures.

Numerical results: The proposed method is tested in conjunction with an image coding scheme. The encoding is achieved by performing a discrete cosine transform (DCT) on a block of 8×8 pixels and quantising the corresponding transform coefficients by a bank of scalar quantisers. The bank of quantisers is optimised by allocating a fixed number of bits in an optimal way among them using a method based on dynamic programming [3]. The partitions of the scalar quantisers are labelled using a natural binary code, and the transmitted codeword for the block is formed by concatenating these labels.

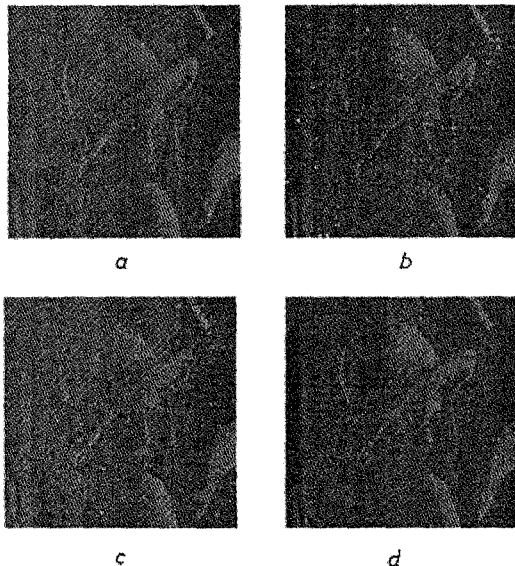


Fig. 2 Test images quantised using the three different methods

- a LMQ (noiseless channel)
- b BSQ ($E_b/N_0 = 0.5$)
- c LMQ ($E_b/N_0 = 0.5$)
- d TCQ ($E_b/N_0 = 0.5$)

Table 1: Summary of quantisation SNR [dB]

E_b/N_0 dB	Equivalent ϵ	LMQ	BSQ	TCQ
0.9	0.5×10^{-2}	20.93	20.43	21.26
0.7	1.0×10^{-2}	18.02	17.91	20.55
0.5	2.0×10^{-2}	15.80	16.29	19.24
0.3	4.0×10^{-2}	13.53	13.84	17.24

Maximum possible SNR = 22.63 dB

The system is tested on a 512×512 Lena image for different channel signal-to-noise ratios. The Turbo-code encoder has a rate of 1/3 with the underlying RCCs of constraint length 3 with the generator polynomial (3, 5). The code block length is equal to 380 and the number of iterations is 10. The results for the three systems, LMQ, BSQ, TCQ are given in Table 1, and the corresponding images are shown in Fig. 2. It can be seen that the TCQ system results in an improvement in the signal-to-noise ratio (SNR) performance, as well as in the image quality.

Acknowledgment: This work was supported by the Information Technology Research Centre (ITRC) of Canada.

Appendix 1:

TCQ design algorithm:

```

Initialise the Reconstruction Levels  $R_i$ 
Calculate all the  $A_i(k)$  and  $B(m, n)$  values
  Loop until distortion is minimal
    Loop for all training data
      Read one data point  $x$ 
      find all distortions  $D_i = 2x \sum_{k=1}^N R_k A_i(k) - \sum_{m=1}^N \sum_{n=1}^N R_m R_n B(m, n)$ 

```

Find z such that $D_z < D_i, \forall i \neq z$

$tally[z] = tally[z] + 1$

$sum[z] = sum[z] + R_z$

Continue

For all levels i , set

$E[\hat{X}|u_i] = sum[i]/tally[i]$

$R_i = sum[i]/tally[i]$

Continue

Continue

© IEE 1997

20 June 1997

Electronics Letters Online No: 19971082

J. Bakus and A.K. Khandani (Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, N2L 3G1, Canada)

References

- 1 MAX, J.: 'Quantizing for minimum distortion', *IEEE Trans. Inf. Theory*, **60**, pp. 7-12
- 2 FARVARDIN, N., and VAISHAMPAYAN, V.: 'Optimal quantizer design for noisy channels: An approach to combined source-channel coding', *IEEE Trans. Inf. Theory*, **87**, pp. 827-838
- 3 BAKUS, J., and KHANDANI, A.K.: 'Combined source-channel coding using Turbo-codes'. 1997 Conf. Inf. Sci. Syst.,

Efficient soft-in-soft-out sub-optimal decoding rule for single parity check codes

Li Ping, S. Chan and Kwan L. Yeung

Indexing terms: Decoding, Codes

The authors present an efficient, sub-optimal, soft-in-soft-out decoding rule for single parity check (SPC) codes, which requires only three addition-equivalent-operations per information bit. Its application is demonstrated by the simulation results of a rate 5/6 four-dimensional concatenated SPC code, for which a performance of $BER = 10^{-5}$ at $E_b/N_0 = 3.5$ dB is observed, which is only ~ 1.2 dB from the theoretical limit.

Introduction: This Letter presents an efficient sub-optimal, symbol-by-symbol, soft-in-soft-out decoding rule for the single-parity-check (SPC) codes [1] and examines its application in decoding concatenated SPC codes. The new method requires only three addition-equivalent-operations (AEO) per information bit, compared with the alternative trellis approach [2, 3] which costs about 18 AEO per information bit. The simulation results of a rate 5/6 concatenated SPC code employing the iterative decoding technique [4, 5] show that a performance of $BER = 10^{-5}$ at $E_b/N_0 \approx 3.5$ dB can be achieved, only ~ 1.2 dB from the theoretical limit for rate 5/6 binary codes.

MAP and log-MAP rule: Consider a code C of length N with values in $\{-1, 1\}$. A codeword in C is denoted by $c = \{c_k : k = 1, 2, \dots, N\}$. The distorted vector is denoted by $x = \{x_k = c_k + n_k\}$ where $\{n_k\}$ are independent random Gaussian variables with zero mean and variance σ^2 . We will assume that all the codewords in C have equal probability of occurrence and so do the bits 1 and -1 . Thus, the output of a maximum a posteriori (MAP) soft-in-soft-output decoding can be described by [3 - 5]

$$L_k = \log \frac{\sum_{c_k=+1} \exp\left(\frac{\langle c, x \rangle}{\sigma^2}\right)}{\sum_{c_k=-1} \exp\left(\frac{\langle c, x \rangle}{\sigma^2}\right)} \quad \text{for } k = 1, 2, \dots, N \quad (1)$$

where $\langle c, x \rangle$ denotes the inner product of c and x . To reduce the computational costs, we can approximate the summations in eqn. 1 by the dominant terms, leading to the so-called log-MAP (or MAX-log-MAP) rule [5 - 7]:

$$L_k \approx \frac{1}{\sigma^2} \left(\max_{c_k=+1} \langle c, x \rangle - \max_{c_k=-1} \langle c, x \rangle \right) \quad (2)$$

The maximisations are over all the codewords with $c_k = +1$ and -1 , respectively. The scaling factor $1/\sigma^2$ in eqn. 2 has no impact on the final results of the iterative decoding based on the log-MAP rule, hence it can be ignored. This avoids the necessity for estimating σ^2 . Clearly, eqn. 2 is computationally more efficient than eqn. 1. The trellis approach [2, 6] can be used to evaluate eqn. 2 for the SPC codes which costs ~ 18 AEO per bit. In the following, we will introduce a method to reduce it to only three AEO per bit.

Main results: Every codeword in an SPC code considered below has an even number of -1 bits [1]. We define a vector s by $s_k = \text{sgn}(x_k)$, $k = 1, 2, \dots, N$, where $\text{sgn}(\cdot)$ is the signum function. Clearly, $\langle s, x \rangle = \sum_{k=1}^N |x_k|$. We denote k_m and k_s such that $|x_{k_m}| = \min\{|x_k| : k = 1, 2, \dots, N\}$ and $|x_{k_s}| = \min\{|x_k| : k = 1, 2, \dots, N, k \neq k_m\}$, i.e. x_{k_m} and x_{k_s} are the entries of x with the minimum and the second minimum amplitudes, respectively. Let p be the number of -1 bits in s . It is easy to verify that for even p :

$$\max_{c_k = s_k} \langle c, x \rangle = \langle s, x \rangle \quad \text{for all } k \quad (3.1)$$

$$\max_{c_k = -s_k} \langle c, x \rangle = \begin{cases} \langle s, x \rangle - 2(|x_k| + |x_{k_m}|) & \text{if } k \neq k_m \\ \langle s, x \rangle - 2(|x_{k_m}| + |x_{k_s}|) & \text{if } k = k_m \end{cases} \quad (3.2)$$

and for p odd:

$$\max_{c_k = s_k} \langle c, x \rangle = \begin{cases} \langle s, x \rangle - 2|x_{k_m}| & \text{if } k \neq k_m \\ \langle s, x \rangle - 2|x_{k_s}| & \text{if } k = k_m \end{cases} \quad (3.3)$$

$$\max_{c_k = -s_k} \langle c, x \rangle = \langle s, x \rangle - 2|x_k| \quad \text{for all } k \quad (3.4)$$

From eqn. 3 we have the following rule for evaluating eqn. 2 (omitting $1/\sigma^2$),

$$\max_{c_k = +1} \langle c, x \rangle - \max_{c_k = -1} \langle c, x \rangle = \begin{cases} 2s_k(|x_k| + |x_{k_m}|) & \text{if } p \text{ even, } k \neq k_m \\ 2s_{k_m}(|x_{k_m}| + |x_{k_s}|) & \text{if } p \text{ even, } k = k_m \\ 2s_{k_m}(|x_k| - |x_{k_m}|) & \text{if } p \text{ odd, } k \neq k_m \\ 2s_{k_m}(|x_{k_m}| - |x_{k_s}|) & \text{if } p \text{ odd, } k = k_m \end{cases} \quad (4)$$

Ignoring the operations such as taking the absolute value, negation and multiply-by-2, the complexity of the above decoding method is $3N-3$ AEO: $2N-3$ comparisons for finding k_s and k_m , and N additions for evaluating eqn. 4. This is only three AEO per information bit.

Eqn. 4 can be compared with the Wagner rule [8] for finding \hat{c} that maximises $\langle c, x \rangle$ in an SPC code, that is, $\hat{c} = s$ for p even and $\hat{c} = s'$ for p odd, where $s'_k = s_k$ except $s'_{k_m} = -s_{k_m}$. The Wagner rule is the most efficient method for soft-in-hard-out decoding of the SPC codes. It costs 1 AEO per bit while a trellis based algorithm [2] costs 6 AEO per bit.

Application example: High rate, high gain codes can be constructed using multidimensional concatenated SPC codes. We adopted the following concatenation scheme. Let $\{D_m = D_m[i, k] : m = 1, 2, \dots, M\}$ be M arrays obtained from interleaving an original information array D , where D, D_1, D_2, \dots, D_M all have the same size $I \times K$. The interleaver length is denoted by $N_d = I \times K$. Apply a length $N = K + 1$ SPC code row-wise to every D_m , $m = 1, 2, \dots, M$, producing M parity check vectors $\{P_m : m = 1, 2, \dots, M\}$, Fig. 1a. The overall codeword is formed by D and P_1, P_2, \dots, P_M , Fig. 1b. We call $D_m P_m$ the m th dimension. The overall coding rate is $R = K/(K + M)$. Similar to the turbo codes [3], the above coding scheme is parallel since parity bits are not re-encoded.

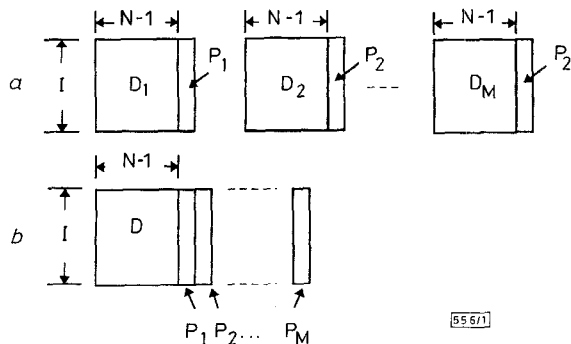


Fig. 1 Multi-dimensional concatenation scheme

a Encoding process
b Overall codeword

The iterative decoding strategy of [5] is employed in our simulation, with the generalisation to multidimensional cases. The product codes (without checks on checks) are special cases of the above scheme, e.g. when $M = 2$, $I = K$, $D_1[i, k] = D[i, k]$ and $D_2[i, k] = D[k, i]$, the above scheme is equivalent to the 2D product code method used in [5]. However, the scheme of Fig. 1 has the advantage of a more flexible interleaver length. The parameters used in our simulation are $N = 21$, $M = 4$, $I = 500$. Thus, $K = 20$, coding rate = $5/6$ and $N_d = 6000$. In this case, an equivalent full size product code would have in a huge information size of $20^4 = 160000$ bits, which is undesirable in many practical systems.

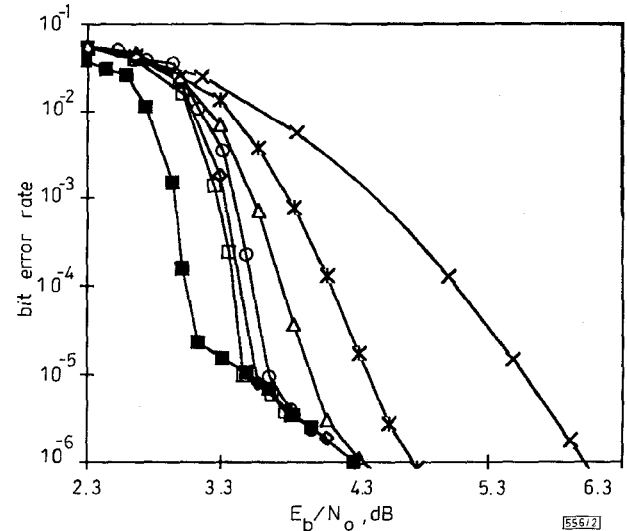


Fig. 2 Simulation results for four-dimensional parallel concatenated $[21, 20, 2]$ SPC coding scheme

$N = 21$, $M = 4$, $I = 500$ and rate = $5/6$

- x— log-MAP, 1 iteration
- *— log-MAP, 2 iteration
- △— log-MAP, 3 iteration
- log-MAP, 5 iteration
- ◇— log-MAP, 10 iteration
- log-MAP, 20 iteration
- MAP, 20 iteration

The interleavers used in our simulations are described by $D_m[i, k] = D[\pi(i, k, m), k]$ with $\pi(i, k, m) = (i + k \times I_m) \bmod 500$. We have chosen $I_m = \{0, 1, 25, 127\}$ empirically. The results are shown in Fig. 2, where the performance of MAP decoding with 20 iterations is also included for comparison. With 20 iterations, the performance of the MAP and log-MAP methods are almost identical for $E_b/N_0 > 3.5$ dB and there is < 0.4 dB difference between the two for $E_b/N_0 < 3.5$ dB. At $E_b/N_0 = 3.5$ dB, the proposed method can reach $BER = 10^{-5}$ with 20 iterations. The theoretical limit of rate $5/6$ binary codes is $\sim E_b/N_0 = 2.3$ dB. The log-MAP curves appear to converge to the MAP curve at high E_b/N_0 . It is interesting to note the corner points of the curves at around $BER \approx 10^{-5}$ and $E_b/N_0 \approx 3.3$ dB. Such a phenomenon is very similar to that reported for the turbo codes [9].

Conclusion: A very low cost, sub-optimal, soft-in-soft-out decoding rule for the single parity check (SPC) codes has been presented. Its application in decoding rate $5/6$, concatenated SPC codes has been examined and a performance within 1.2 dB from the theoretical limit has been observed in the range $BER = 10^{-5}$.

© IEE 1997

21 July 1997

Electronics Letters Online No. 19971092

Li Ping, S. Chan and Kwan L. Yeung (Department of Electronics Engineering, City University of Hong Kong, Hong Kong)

E-mail: eeliping@cityu.edu.hk

References

- 1 SKLAR, B.: 'Digital communications' (Prentice-Hall, 1988)
- 2 WOLF, J.: 'Efficient maximum likelihood decoding of linear block codes using a trellis', *IEEE Trans. Inf. Theory*, 1978, **IT-24**, pp. 76-81

- 3 BAHL, L., COCKE, J., JELINEK, F., and RAVIV, J.: 'Optimal decoding of linear codes for minimizing symbol error rate', *IEEE Trans. Inf. Theory*, 1974, **IT-20**, pp. 284-287
- 4 BERROU, C., GLAVIEUX, A., and THEITIMAJSHIMA, P.: 'Near Shannon limit error-correcting coding and decoding: Turbo-codes', Proc. IEEE ICC-93, 1993, pp. 1064-1070
- 5 HAGENAUER, J., OFFER, E., and PAPKE, L.: 'Iterative decoding of binary block and convolutional codes', *IEEE Trans. Inf. Theory*, 1996, **IT-42**, pp. 429-445
- 6 ROBERTSON, P., VILLEBRUN, E., and HOEHER, P.: 'A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain'. Proc. IEEE ICC-95, 1995, pp. 1009-1013
- 7 PING, L., CHAN, C., and YEUNG, K.L.: 'Max-log-MAP filtering algorithm for decoding product F_{24} code'. Proc. IEEE ICC-97, 1997, pp. 1366-1370
- 8 SILVERMAN, R.A., and BALSER, M.: 'Coding for a constant data rate source', *IRE Trans. Inf. Theory*, 1954, **IT-4**, pp. 50-63
- 9 BENEDETTO, S., and MONTORSI, G.: 'Unveiling turbo codes: some results on parallel concatenated decoding schemes', *IEEE Trans. Inf. Theory*, 1996, **IT-42**, pp. 409-428

ID-based group signature

Sangjoon Park, Seungjoo Kim and Dongho Won

Indexing terms: Identification, Cryptography

The authors present an ID-based group signature which is based on ordinary ID-based signature schemes such as Ohta-Okamoto's scheme and Guillou-Quisquater's scheme. Thus, the group signature is verified from the identities of group members. A signer proves that, by verifiable encryption of his ordinary signature, a group authority can identify him and, by Schoenmaker's method, he proves that he knows a signature of a group member.

Introduction: Group signatures were first proposed by Chaum and Heijst [1]. Their schemes were improved by Chen and Pedersen [2] who first used a Schoenmaker's method in application of a group signature. Petersen suggested a general method for converting any digital signature scheme into a group signature scheme [3]. However, since all previously proposed schemes are based on the discrete logarithm method, identity information is not unique to the signer's public key. In this Letter, we propose an ID-based group signature which is based on ID-based signature schemes such as Ohta-Okamoto's scheme [4] and Guillou-Quisquater's scheme [5]. The scheme consists of four kinds of participant: a trusted centre for generating the secret keys of all users, a group authority for identifying a signer, a signer (a group member) for issuing group signatures and a receiver for verifying them. In the scheme, each signer has only one secret key for both a group signature and an ordinary signature.

Key generation: The trusted centre (TC) generates secret keys for users. First, he chooses a modulus $n = p_1 \cdot p_2$, where p_1, p_2 and q are primes with $q|p_1 - 1$ and $q|p_2 - 1$. For security, the bit size of q is ~ 160 . He selects $g \in Z_n$ with the order of q , where Z_n is the integer ring. Let e be a large (160 bit) integer with $\gcd(e, \phi(n)) = 1$. He publishes n, q, g and e . Let $id_i \in Z_n$ be the identity information of a user. He computes a secret key s_i with $id_i = s_i^e \pmod n$. Now, a group authority generates his key pair (x, y) with $y = g^x \pmod n$ and publishes the public key y . Since it is difficult to obtain x from y , TC cannot know the secret key x .

Signing phase: A signer uses Stadler's verifiable encryption of the e th root (his ordinary signature) [6] to prove that a group authority can identify him and, by Schoenmaker's method [2], he proves that he knows the signature of a group member. Let $G = \{id_1, \dots, id_k\}$ be a set of identities of group members and $h(\cdot)$ be a secure hash function. A member id_i generates a group signature as follows:

- (i) $R = r^e \pmod n$, $c = s_i^{h(m,R)} \cdot r \pmod n$, where r is random in Z_n
- (ii) $A = g^\alpha \pmod n$, $B = c \cdot y^\alpha \pmod n$, where α is random in Z_n
- (iii) $C_i = id_i^{h(m,R)} \cdot R \pmod n$ ($i = 1, \dots, k$)

(iv) w_1, \dots, w_k and d_2, \dots, d_k are randomly chosen in Z_n and compute t_{g_i} s and t_{y_i} s, where $t_{g_i} = g^{w_i} \pmod n$, $t_{y_i} = y^{w_i} \pmod n$ and $t_{g_i} = g^{w_i} \cdot A^{d_i} \pmod n$, $t_{y_i} = y^{w_i} \cdot (B/C_i)^{d_i} \pmod n$ ($2 \leq i \leq k$)

(v) $d_1 = d_0 - \sum_{i=2}^k d_i \pmod q$, where

$$d_0 = h(t_{g_1}, \dots, t_{g_k}, t_{y_1}, \dots, t_{y_k}, A, B, h(m, R)) \quad (1)$$

(vi) $r_i = w_i - d_i \cdot \alpha \pmod q$, and $r_i = w_i$ ($2 \leq i \leq k$)

(vii) the signer sends a group signature $(m, R, A, B, d_0, d_1, \dots, d_k, r_1, \dots, r_k)$.

Verification phase: We suppose that a receiver knows the identities id_i s of a group G . The group signature is verified by G as follows:

(i) compute $C_i = id_i^{h(m,R)} \cdot R \pmod n$

(ii) confirm the equation $d_0 \stackrel{?}{=} \sum_{i=1}^k d_i \pmod q$

(iii) Compute t_{g_i} and t_{y_i} ($i = 1, \dots, k$), where $t_{g_i} = g^{r_i} \cdot A^{d_i}$, $t_{y_i} = y^{r_i} \cdot (B/C_i)^{d_i}$

(iv) if eqn. 1 holds, he accepts the signature.

Identification phase: A group authority can identify the signer without any assistance from group members as follows:

(i) obtain the ordinary signature $c = B/A^x \pmod n$ by deciphering (A, B)

(ii) Find id_i satisfying the equation $c^e \stackrel{?}{=} id_i^{h(m,R)} \cdot R = C_i \pmod n$ ($i = 1, \dots, k$)

and since $c = s_i^{h(m,R)} \cdot r \pmod n$ $c^e = C_i \pmod n$ and $c^e \neq C_i \pmod n$ ($2 \leq i \leq k$), so he can determine the signer id_i .

Security: First, the signer id_i cannot forge another member's signature. To forge a signature of the member id_j , the e th root of $C_j = id_j^{h(m,R)} \cdot R \pmod n$ should be computed. But, since he does not know the factor p_1 and p_2 , he can not compute the e th root of C_j . Second, a receiver and a group authority, who have no secret keys s_i 's of the group G , cannot generate a group signature.

If $d_0 = h(t_{g_1}, \dots, t_{g_k}, t_{y_1}, \dots, t_{y_k}, A, B, h(m, R))$ is given, then it is hard to get another A, B, t_{g_i} 's and t_{y_i} 's satisfying eqn. 1 because $h(\cdot)$ is a secure hash function. Obtaining r_i and d_i from given A, B, t_{g_i} and t_{y_i} depends on the discrete logarithm problem. Third, TC and a receiver cannot determine a signer of the group signature. Let $S = \{z|z = g^e \pmod n\}$. Then, t_{g_i} 's and t_{y_i} are randomly distributed in S . On the other hand, the possibility of $t_{y_i} \notin S$ ($2 \leq i \leq k$) is very high. However, since the size of the set S is sufficiently large, they cannot determine whether t_{y_i} is an element of S or not. Moreover, since they do not know x , the ordinary signature c cannot be derived from (A, B) .

Conclusion: We have proposed an ID-based group signature which is based on ordinary ID-based signature schemes. The group signature is verified by identities of group members which a receiver already knows. In the scheme, we combined Stadler's verifiable encryption of the e th root and Schoenmaker's method. The secret key for a group signature can also be used to issue an ordinary ID-based signature. The security of the proposed scheme depends on both the discrete logarithm problem and the e th root problem.

© IEE 1997

13 June 1997

Electronics Letters Online No:19971065

Sangjoon Park (#0710, Electronics and Telecommunications Research Institute, Yusong PO Box 106, Taejeon, 305-600 Korea)

Seungjoo Kim and Dongho Won (Department of Information Engineering, Sung Kyun Kwan University, 300 Chunchun-dong, Suwon, Kyunggi-do, 440-746, Korea)

E-mail: sjpark@dingo.etri.re.kr

References

- 1 CHAUM, D., and HEIJST, E.: 'Group signatures', Paper LNCS 547, Advances in Cryptology - EUROCRYPT'91, (Springer, 1992), pp. 257-265
- 2 CHIEN, L., and PEDERSEN, T.: 'New group signature schemes'. Advances in Cryptology - EUROCRYPT'94, Paper LNCS 950, (Springer, 1995), pp. 163-173
- 3 PETERSEN, H.: 'How to convert any digital signature scheme into a group signature scheme'. Proc. Security Protocols Workshop, 1997, (in press)
- 4 OHTA, K., and OKAMOTO, T.: 'Practical extension of Fiat-Shamir scheme', *Electron. Lett.*, 1988, **24**, (15), pp. 955-956