$$v_3(\alpha) = \frac{\sin\alpha}{\sqrt{\sin^2\alpha - \sin^2\delta}}\left\{w(\alpha) - \frac{1}{w(\alpha)}\right\}$$
$$v_4(\alpha) = w(\alpha) + \frac{1}{w(\alpha)} \tag{15}$$

These are both even functions of α, free of poles and zeros in $|\text{Re.}\alpha| < \pi$, and $O\,\{\exp[\frac{1}{2}|\text{Im.}\alpha|]\}$ as $|\text{Im.}\alpha| \to \infty$. In any given diffraction problem, the constraints which are imposed include the order of the allowed solution and specify the particular solution required.

The key point to be observed is that to construct solutions of eqn. 2, it is not necessary to regularise (or make uniform) the function $w(\alpha)$; to do so would involve the use of elliptic function integrals and the introduction of extraneous pole-zero pairs [5] on a two-sheeted Riemann surface to eliminate the contributions of loop integrals around the branch cuts of $\sqrt{[\sin^2\alpha - \sin^2\delta]}$. This significantly complicates the analysis as well as the final expression for $v(\alpha)$.

19 June 1997
Electronics Letters Online No: 19971064

T.B.A. Senior and S.R. Legault (*Radiation Laboratory, Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI 48109-2122, USA*)

References

1 SENIOR, T.B.A.: 'Some problems involving imperfect half planes' *in* USLENGHI, P.L.E. (Ed.): 'Electromagnetic scattering' (Academic Press, New York, 1978), pp. 185–219
2 DEMETRESCU, C.: 'A general solution of the second-order difference equation occurring in diffraction theory', *Radio Sci.*, 1996, **31**, pp. 461–467
3 MALIUZHINETS, G.D.: 'Excitation, reflection and emission of surface waves from a wedge with given face impedances', *Sov. Phys.-Dokl.*, 1958, **3**, pp. 752–755
4 LEVY, H., and LESSMAN, F.: 'Finite difference equations' (Dover Pub. Inc., New York, 1992), pp. 181–182
5 APPELL, P., and GOURSAT, E.: 'Théorie des fonctions algébriques et de leurs intégrales, Vol. 1' (Gauthier-Villars, Paris, 1929), Chap. 8

Combined source-channel coding using Turbo-codes

J. Bakus and A.K. Khandani

Indexing terms: Source coding, Channel coding, Turbo codes

The authors present a new approach to combined source-channel coding based on using the reliability information available from the Turbo-code channel decoder. Numerical results are presented for the transmission of the DCT coefficients of still images showing a noticeable improvement with respect to an ordinary, as well as a channel optimised quantiser.

Introduction: This Letter is concerned with improving the performance of a quantisation scheme by taking advantage of the reliability information available from a Turbo-code channel decoder (or, in general, any decoder with a soft output). We combine a Lloyd-Max type scalar quantiser [1] with the Turbo-code structure to produce a combined source-channel coding method in which the reconstruction values are computed as a sum of the reconstruction levels (centroids) weighted by the corresponding likelihood values. We use the name Turbo coded quantiser (TCQ) for the proposed method.

The performance of the method is compared with two other schemes. The first scheme is based on a standard Lloyd-Max quantiser which is designed neglecting the effect of channel noise. In the second scheme, the quantiser is designed by modelling the Turbo-code and the channel as a binary symmetric channel. The quantiser design in this case is based on the method given in [2]. We refer to these alternatives as the LMQ and BSQ, respectively. In all three cases, the mean square error is used as the distortion measure. The three methods are tested on the quantisation of the DCT coefficients of still images at a number of different channel noise levels. In all cases, the proposed method performs better than the other two alternatives from both subjective and objective points of view.

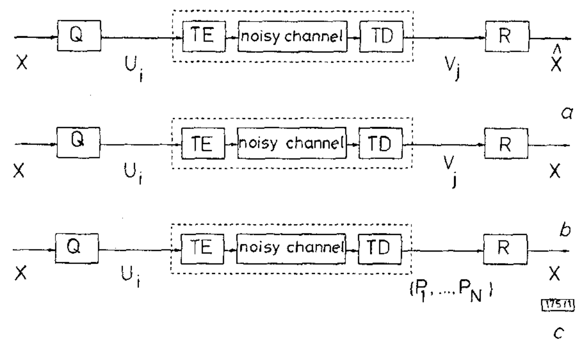

Fig. 1 *Block diagrams of the three quantisation schemes*

a [LMQ] Lloyd-Max quantiser
b [BSQ] binary symmetric channel quantiser
c [TCQ] Turbo coded quantiser

Basic structure: The block diagrams of the three methods investigated are shown in Fig. 1, where the quantiser is represented by *Q*, the reconstructor by *R*, and the Turbo-code encoder and decoder by *TE* and *TD*, respectively. The encoder inputs a sample x, quantises it using *Q*, and then maps the result to a symbol selected from an alphabet set $\{u_1, ..., u_N\}$. The corresponding symbol is represented by a code word of $\log_2 N$ bits. The code words are concatenated, encoded by the Turbo-code encoder *TE*, and sent through the channel. The Turbo-decoder *TD* calculates the LLR (log-likelihood ratio), Λ, of each bit, according to:

$$\Lambda = \ln\frac{\Pr(\text{bit 1 sent}|\text{channel output})}{\Pr(\text{bit 0 sent}|\text{channel output})} \tag{1}$$

In the LMQ, and BSQ methods, the decoder makes a hard decision on the LLR value to estimate the corresponding bit. The received bits are then concatenated to form a received codeword, which is subsequently mapped to a received symbol from an alphabet set $\{v_1, ..., v_N\}$. This symbol is mapped by the reconstructor *R* to the output value $\hat{X}$.

The proposed method differs from the LMQ and BSQ by the fact that no hard decision is made at the decoder. Instead, the LLR value for each bit is used to calculate the *bit a-posteriori probability* according to:

$$P_b(0) = \Pr(\text{bit 0 sent}) = \frac{1}{1+\exp(\Lambda)}$$
$$P_b(1) = \Pr(\text{bit 1 sent}) = \frac{\exp(\Lambda)}{1+\exp(\Lambda)} \tag{2}$$

The bit a-posteriori probabilities within one codeword are used to calculate a set of *symbol a-posteriori probabilities* $\{P_1, ..., P_N\}$, where P_i is the probability that u_i was sent. Using these a-posteriori probabilities, the reconstructor estimates the transmitted symbol as a weighted average of the reconstruction levels (centroids), as given by

$$\hat{X} = \sum_{i=1}^{N} P_i R_i \qquad R_i = E[x|u_i] \tag{3}$$

To formulate the encoding procedure, the Turbo-coded channel is described using two quantities $A_i(k)$ and $B_i(m, n)$ defined as

$$A_i(k) = E[P_k|u_i]$$
$$B_i(m,n) = E[P_m P_n|u_i] \tag{4}$$

These quantities are measured by passing a test bit pattern through the Turbo-coding scheme. In this case, the input sample x is quantised to the quantisation level i which minimises the distortion given by

$$D_i = E[(x-\hat{X})^2|u_i]$$
$$= x^2 - 2x\sum_{k=1}^{N} R_k A_i(k) + \sum_{m=1}^{N}\sum_{n=1}^{N} R_m R_n B_i(m,n) \tag{5}$$

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.

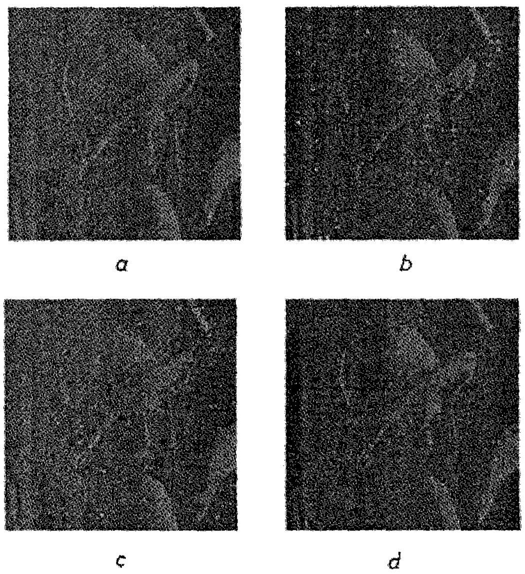

Fig. 2 *Test images quantised using the three different methods*

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	Equivalent ϵ	LMQ	BSQ	TCQ
dB				
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 = 22.63dB

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 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 $B_i(m, n)$ values
    Loop until distortion is minimal
    Loop for all training data
        Read one data point x
        find all distortions $D_i = 2x \Sigma_{k=1}^{N} R_k A_i(k) - \Sigma_{m=1}^{N} \Sigma_{k=1}^{N} R_m R_n B_i(m, n)$
        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

 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 ~1.2dB 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 $E_b/N_0 \simeq 3.5$dB can be achieved, only ~1.2dB 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 $c = \{c_k : k = 1, 2, ..., N\}$. 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]

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

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]:

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