Reducing bit width of extrinsic memory in turbo decoder realisations

J. Vogt, J. Ertel and A. Finger

Extrinsic memory consumes a substantial part of turbo decoder resources. The authors present a new method by which the necessary bit width of the extrinsic symbols is reduced by applying a nonlinear quantisation. Simulations demonstrate that the bit width of the extrinsic symbols can be reduced to three bits without significant loss in performance.

Introduction: Turbo coding [1] has attracted great interest due to large coding gains. It has been chosen for the third mobile communications standard (IMT-2000) and it is planned to use turbo codes in future DVB systems. For systems with higher data rates an ASIC design is often used. Especially in mobile applications, minimising the necessary chip area and power consumption is important. A very large part of the overall turbo decoder chip area is consumed by the memory [2]. For the 3GPP standard with $K = 4$, $R = 1/3$ and a maximum block length of 5114 bits, a complexity estimation shows memory consumption > 66%, most of this being used for channel and extrinsic symbols. Fig. 1 shows the relative area consumption of a designed turbo decoder with a soft-in-soft-out (SISO) unit. The complexity estimation clearly indicates that it is necessary to shrink the overall memory.

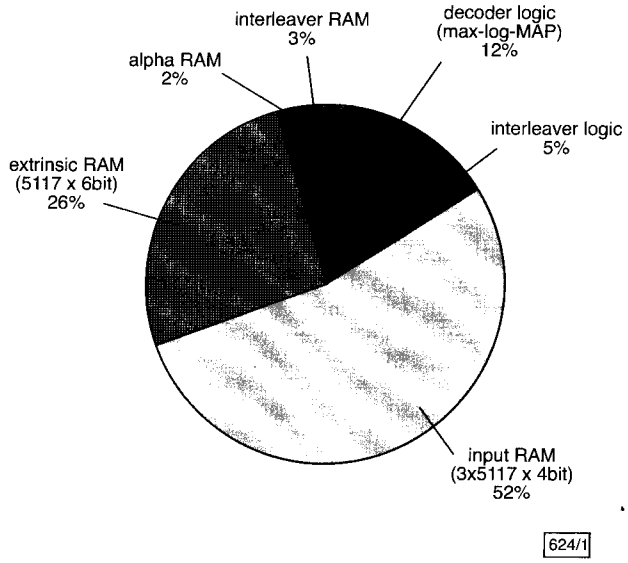

Fig. 1 *3GPP turbo decoder chip area consumption*
$K = 4$, $R = 1/3$, block length: 40 – 5114

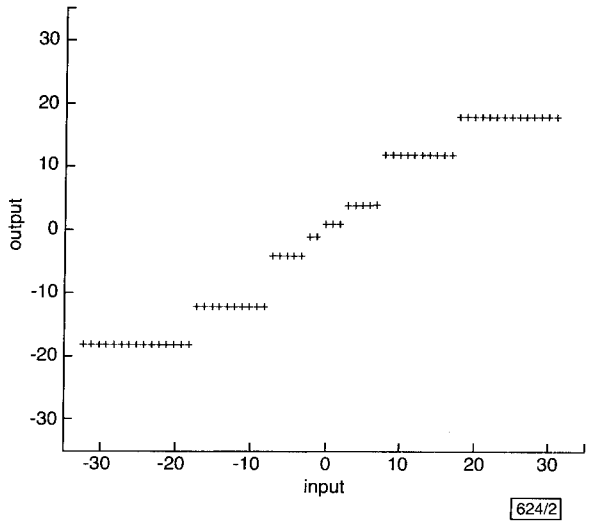

Fig. 2 *Quantisation characteristic*

In this Letter we present a method to reduce the necessary bit width of the extrinsic symbols to three bits without significant performance loss. Simulation results are given for the max-log-MAP algorithm.

Turbo decoder: The turbo decoding process is carried out in an iterative manner. SISO decoding of the convolutional subcodes is carried out with the use of *a priori* information from previous decoding steps. Here, only relevant formulas of the max-log-MAP algorithm are given. For further details of turbo codes and turbo decoders, see [1, 3 – 5]. In contrast to the MAP algorithm, the max-log-MAP uses log likelihood ratios (LLR) [5] and decoding is independent of the channel signal to noise ratio (SNR). Branch metrics are calculated as follows:

$$\gamma_k(s, s') = x_k^s \left(L_{in}^e + y_k^s\right) + x_k^p \cdot y_k^p \qquad (1)$$

where x^s and x^p are the systematic and the redundancy outputs of the turbo encoder for the state transition s' to s and y^s and y^p are the received channel symbols. L^e is an *a priori* information for u_k calculated in the previous iteration.

Forward and backward state metrics α, β are calculated as follows:

$$\alpha_k(s) = \max_{s' \in S} \left(\alpha_{k-1}(s') + \gamma_k(s', s)\right) \qquad (2)$$

$$\beta_k(s) = \max_{s' \in S} \left(\beta_{k-1}(s') + \gamma_k(s', s)\right) \qquad (3)$$

LLR for a corresponding u_k is defined as:

$$L(u_k) = \max_{S^1} \left(\alpha_{k-1}(s') + \gamma_k(s', s) + \beta_k(s)\right) - \max_{S^0} \left(\alpha_{k-1}(s') + \gamma_k(s', s) + \beta_k(s)\right) \qquad (4)$$

From $L(u_k)$ the extrinsic information can be derived, which is used as the *a priori* information in the next decoder iteration:

$$L_{out}^e(u_k) = \frac{L(u_k)}{2} - \left(y_k^s + L_{in}^e(u_k)\right) \qquad (5)$$

For the entire block it is necessary to save $L_{out}^e(u_k)$ in a temporary memory to be available in the next iteration. This memory is called the extrinsic memory and the size depends on the block length and the number of bits per symbol.

Fig. 3 *Transformation of extrinsic symbols*

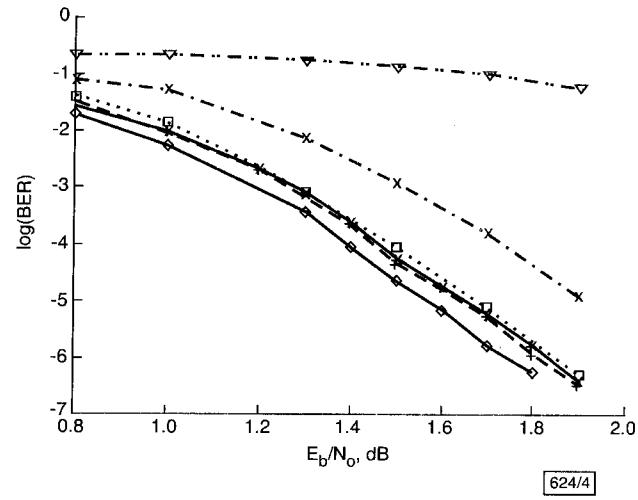

Fig. 4 *BER simulation with max-log-MAP, 656 bit 3GPP-interleaver and eight iterations, AWGN*

—◇— floating point operation
– –+– – quantisation operation and six-bit extrinsic
····□···· quantisation operation and five-bit extrinsic
· –×· – quantisation operation and four-bit extrinsic
– ··▽– ·· quantisation operation and three-bit extrinsic
—✳— quantisation with new transformation and three-bit extrinsic

Transformation of extrinsic symbols: Reducing the bit width of the extrinsic symbols is achieved by applying a quantisation scheme after calculation of $L^e(u_k)$. A linear quantisation scheme has shown reasonable results, but better results can be reached with a heuristically determined nonlinear quantisation scheme, as shown in Fig. 2. This means the amount of different numbers is restricted to the eight indicated intervals. These intervals can be encoded using a three-bit number and saved for the next decoder operation. Before using the saved number as an *a priori* value in the next process the number again needs to be decoded to the intervals defined in Fig. 2. Fig. 3 shows the additional operations, contrasted with traditional processing. Limiting the allowed intervals and the encoding is done in the T-box, and the decoding is done in the T^{-1}-box.

Simulation results: Simulations were carried out with an ideal (floating point) and with different quantised decoders. The quantised decoders use four-bit channel symbols and six-bit state metrics. For quantised decoders clipping of the least significant bits (LSBs) of the extrinsic values was contrasted with our proposed nonlinear quantisation scheme. It can be seen in Fig. 4 that for clipping at least five bits, and in case of our proposed quantisation only three bits, are necessary for only a small performance loss.

Investigations with different block lengths have shown a performance loss of 0.1 to 0.3dB, in contrast to the ideal floating point operation.

Conclusion: A new method for reducing the necessary bit width of the turbo decoder extrinsic memory has been shown. Simulations show that only three bits are necessary to reach near ideal performance. Reducing the extrinsic bit width from six to three bits can reduce the overall turbo decoder complexity by 13% for the 3GPP standard. The introduced transformation also works well for serial concatenated codes.

Acknowledgment: This research was supported by Asahi Chemical Industrie Co., Ltd.

7 August 2000
Electronics Letters Online No: 20001177
DOI: 10.1049/el:20001177

J. Vogt, J. Ertel and A. Finger (*Department of Electrical Engineering, Dresden University of Technology, 01062 Dresden, Germany*)

E-mail: vogt@ifn.et.tu-dresden.de

References

1 BERROU, C., GLAVIEUX, A., and THITIMAJSHIMA, P.: 'Near Shannon limit error-correcting coding and decoding: turbo-codes (1)'. Proc. ICC 1993, Geneva, Switzerland, May 1993, pp. 1064–1070

2 VOGT, J., KOORA, K., FINGER, A., and FETTWEIS, G.: 'Comparison of different turbo decoder realizations for IMT-2000'. Proc. GLOBECOM '99, Rio de Janeiro, Brazil, Dec. 1999, pp. 2704–2708

3 BAHL, L.R., COCKE, J., JELINEK, F., and RAVIV, J.: 'Optimal decoding of linear codes for minimizing symbol error rate', *IEEE Trans. Inf. Theory,* 1974, **20**, pp. 284–287

4 ROBERTSON, P., VILLEBRUN, E., and HOEHER, P.: 'A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain'. Proc. ICC, Seattle, Washington, USA, June 1995, pp. 1009–1013

5 HAGENAUER, J., ROBERTSON, P., and PAPKE, L.: 'Iterative (turbo) decoding of systematic convolutional codes with the MAP and SOVA algorithms'. Proc. ITG Conf., Munich, Germany, Oct. 1994