COMPARATIVE STUDY OF TURBO EQUALISERS USING CONVOLUTIONAL CODES AND BLOCK-BASED TURBO CODES FOR GMSK MODULATION

B.L. Yeap, T.H. Liew, J. Hámorský, L. Hanzo

Dept. of Electr. and Comp. Sc., Univ. of Southampton, SO17 1BJ, UK.
Tel: +44-1703-593 125, Fax: +44-1703-594 508
Email: lh@ecs.soton.ac.uk
http://www-mobile.ecs.soton.ac.uk

ABSTRACT

In contrast to previously proposed turbo equalisers, where typically non-iterative channel decoders were used, this paper compares the performance of partial-response GMSK turbo equalisers using two different encoders, namely block BCH turbo codes and convolutional codes. The BER performance is assessed over non-dispersive Gaussian channels, and dispersive Rayleigh fading channels.

1. INTRODUCTION

Turbo equalisation [1] was first introduced by C.Douillard, A.Picart, M.Jézéquel, P.Didier, C.Berrou and A.Glavieux in 1995 for a serially concatenated rate $R = 0.5$ convolutional coded BPSK system. Specifically, Douillard *et al* demonstrated that the turbo equaliser was capable of mitigating the effects of Inter-Symbol Interference (ISI), provided that the channel impulse response is known. Instead of performing the equalisation and error correction decoding independently, better performance can be achieved by considering the channel's memory, when performing joint equalisation and decoding iteratively. Gertsman and Lodge [5] then showed that the iterative process of turbo equalisers can compensate for the degradations due to imperfect channel estimation. A turbo equalisation scheme for the Global System of Mobile Communications, - known as GSM - was also proposed by Bauch and Franz [6], where different approaches were investigated to overcome the dispersion of the so-called *a priori* information due to the interburst interleaving scheme used in GSM. Research into combined turbo coding using convolutional constituent codes and turbo equalisation have also been investigated by Raphaeli and Zarai [7].

In this paper, we compare the performance of the GMSK turbo equaliser using convolutional codes and block-based turbo codes. Specifically, the Bose-Chaudhuri-Hocquengham (BCH) codes [2, 3] are used as the component codes of the block-based turbo codec, which will be denoted as BTC. Since BCH codes may be constructed with parameters n and k, which represent the number of coded bits and data bits, respectively, we will use the notation BCH (n,k).

2. PRINCIPLE OF TURBO EQUALISATION USING TURBO CODES

With reference to Figure 1, in this section we describe the turbo equalisation principle for a baseband receiver consisting of an equaliser and N_d component decoders. Typically, for turbo codes there are $N_d = 2$ component decoders, whereas for non-iterative convolutional codes we have $N_d = 1$. A vital rule for such turbo equalisers is that the input information to a particular block in the current iteration must not include the information contributed by that particular block from the previous iteration, since then the information used in consecutive iterations would be dependent on each other [5]. We will elaborate on this later on.

Each of the blocks in Figure 1 employ a Soft-In/Soft-Out (SISO) algorithm, such as the optimal Maximum *A Posteriori* (MAP) algorithm [10] or the Log-MAP algorithm [9], which yields the *a posteriori* information. The *a posteriori* information concerning a bit is the information that the SISO block generates taking into account all available sources of information about the bit. When using the MAP algorithm or the Log-MAP algorithm, we express the *a posteriori* information in terms of its so-called Log Likelihood Ratio [9] (LLR).

At the equaliser, - which is denoted by Stage 0 in Figure 1, - the output *a posteriori* LLR consists of the LLR of the *a priori* information and the LLR of the combined channel and extrinsic information. For clarity, we employ the approach used by Gertsman and Lodge [5] and express the LLR of the equaliser using vector notations, giving:

$$L_p^c(0) = L^i + L_p^a(0), \qquad (1)$$

where $L_p^c(0)$ is the composite *a posteriori* LLR of Stage 0 in Figure 1 at the pth iteration, while L^i is the LLR of the combined channel and extrinsic information, and $L_p^a(0)$ is the Stage 0 *a priori* LLR at the pth iteration. We are unable to separate the channel information and extrinsic information at the output of the equaliser

0-7803-5435-4/99/$10.00 © 1999 IEEE

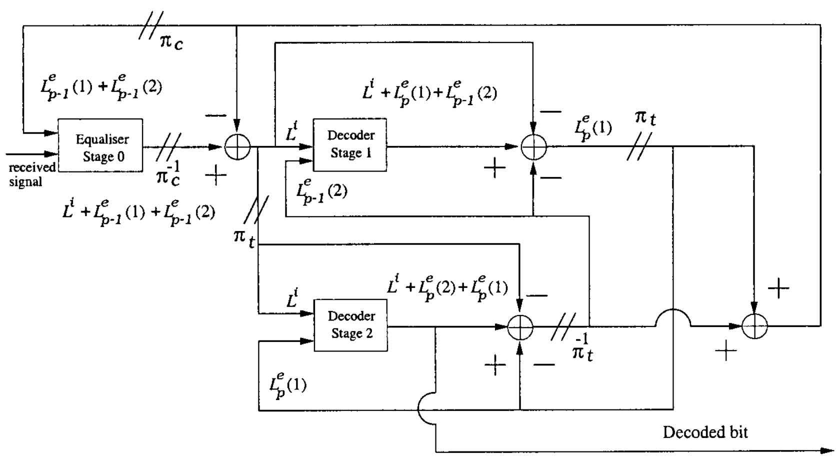

Figure 1: Structure of the turbo equaliser using $N_d = 2$ component decoders. For conceptual simplicity, we have omitted the interleavers and only marked the interleaver positions, where π_c and π_t represent the channel interleaver and turbo interleaver, respectively. The superscript '−1' is used to denote a deinterleaver.

since the channel impulse response can be viewed as that of a non-systematic code [6]. Here, the *a priori* information is obtained from the previously determined extrinsic information of the other N_d decoder stages, - namely Stages 1 and 2 in Figure 1, - which can be expressed as:

$$L^a_p(0) = \sum_{j=1}^{N_d} L^e_{p-1}(j), \tag{2}$$

where $L^e_{p-1}(j)$ is the extrinsic LLR of the jth decoder stage from the previous iteration, namely from $p-1$. Substituting Equation 2 into Equation 1, the composite *a posteriori* LLR at the output of the equaliser in Figure 1 becomes:

$$L^c_p(0) = L^i + \sum_{j=1}^{N_d} L^e_{p-1}(j). \tag{3}$$

At the dth systematic decoder of the pth iteration - i.e. at the inputs of Stages 1 and 2 in Figure 1, - we receive the sum of the augmented *a priori* information $L^a_p(d)$, as well as the combined channel and extrinsic LLR L^i from the equaliser. The augmented *a priori* information at the pth iteration can be expressed as:

$$L^a_p(d) = \sum_{j=1}^{d-1} L^e_p(j) + \sum_{j=d+1}^{N_d} L^e_{p-1}(j). \tag{4}$$

From Equation 4, we observe that the *a priori* information $L^a_p(d)$ consists of the extrinsic information from the decoders of the **previous stages**, - namely for $j < d$ in the **current iteration** - and from the decoders of the **later stages**, - namely for $j > d$ from the **previous iteration**, - but does not include any extrinsic information from Stage d.

For the dth stage systematic decoder, - i.e. at the inputs of the Stages 1 and 2 in Figure 1, - the *a posteriori* LLR $L^c_p(d)$ can be expressed as the sum of the augmented *a priori* information $L^a_p(d)$, the extrinsic information $L^e_p(d)$ produced by the decoder stage and the combined channel LLR and extrinsic LLR L^i from the equaliser:

$$L^c_p(d) = L^a_p(d) + L^e_p(d) + L^i. \tag{5}$$

By substituting Equation 4 into Equation 5 for our specific scheme, we obtain at the output of decoder Stages 1 and 2 in Figure 1:

$$\begin{aligned} L^c_p(d) &= L^a_p(d) + L^e_p(d) + L^i \\ &= \sum_{j=1}^{d-1} L^e_p(j) + \sum_{j=d+1}^{N_d} L^e_{p-1}(j) + L^e_p(d) + L^i \\ &= \sum_{j=1}^{d} L^e_p(j) + \sum_{j=d+1}^{N_d} L^e_{p-1}(j) + L^i. \end{aligned} \tag{6}$$

Explicitly, only two component encoders are used in practical turbo encoders. Therefore, by substituting $N_d = 2$ into Equations 3 and 6, we can determine the *a posteriori* LLR of the equaliser and decoders, whereas Equations 2 and 4 can be used to determine the corresponding *a priori* inputs into the equaliser and decoders. Again, Figure 1 illustrates the structure of the turbo equaliser, which employs the principles discussed previously. For conceptual simplicity, we have omitted the channel interleaver and turbo interleavers. We have only marked their positions, where π_c and π_t represent the channel and turbo interleaver, respectively. The superscript '−1' is used to represent a deinterleaver.

In the following Section, we present the results of the different turbo equalisation systems, using the principles described in this Section.

Simulation Parameters	
Modulation	GMSK: $B_n = 0.3$, transmission frequency=900 MHz, bit rate = 270.833 Kbit/s
BCH (15,11) turbo encoder	Rate=0.58 BCH (15,11) turbo code No puncturing, random turbo interleaver with a depth of 12100 bits Random channel interleaver with a depth of 20900 bits
BCH (31,26) turbo encoder	Rate=0.722 BCH (31,26) turbo code No puncturing, random turbo interleaver with a depth of 14300 bits Random channel interleaver with a depth of 19800 bits
Convolutional encoder Rate = 0.5	$G0 = 35$ $G1 = 23$ octal, no puncturing Random channel interleaver with a depth of 20000 bits
Convolutional encoder Rate = 0.75	$G0 = 35$ $G1 = 23$ octal.$R = 0.5$ convolutional code punctured using the pattern of Table 2. Random channel interleaver with a depth of 20000 bits
BCH decoders Convolutional decoder GMSK equaliser	Log-MAP algorithm [9]
Channel	a. Non-dispersive Gaussian channel b. Equally-weighted 5-path Rayleigh fading channel:Doppler frequency=40.3 Hz

Table 1: Simulation parameters for the turbo equaliser using the BCH turbo decoders and convolutional decoder.

3. SYSTEM PERFORMANCE

We commence by comparing the decoding performance of the two partial-response GMSK systems, where one of them uses the convolutional encoder/decoder (DEC-CONV), while the other employs the BCH turbo encoder/decoder (DEC-BTC), which performs 8 **decoding iterations**. The code rates investigated are $R \approx 0.5$ and $R \approx 0.75$. In this paper, we have assumed that we had perfect knowledge of the channel impulse response. Note that the term **decoding** refers here to the scenario, where the equaliser passes soft outputs to the decoder and there is no iterative processing between the equaliser and decoder. When using turbo decoding, there will be **decoding iterations**, where information is passed iteratively between the component decoders, but not between the decoders and the equaliser. Information is only passed iteratively between the equaliser and decoder, when **turbo equalisation** is employed.

The parameters of the simulation are summarised in Table 1. Both systems are evaluated over the non-dispersive Gaussian channel and an equally-weighted, symbol spaced five-path Rayleigh fading channel. We then compared the turbo equalisation performance of the GMSK system employing convolutional codes (TEQ-CONV) with that using BCH turbo codes (TEQ-BTC) for code rates $R \approx 0.5$ and $R \approx 0.75$ after 8 **turbo equalisation iterations** and setting the delay imposed by the depth of the channel interleaver to approximately 20000 bits. The $R = 0.75$ convolutional code was obtained by puncturing the $R = 0.5$ convolutional code according to the Digital Video Broadcast (DVB) standard's puncturing pattern [11] specified in Table 2, where $G0$ and $G1$ are the generator polynomials.

Over the non-dispersive Gaussian channel, we still have to perform equalisation for the partial-response GMSK modulation scheme since controlled ISI is deliberately introduced in the modulator [4]. The decoding performance of the $R = 0.58$ DEC-BTC system using the BCH (15,11) code after 8 **decoding iterations** is

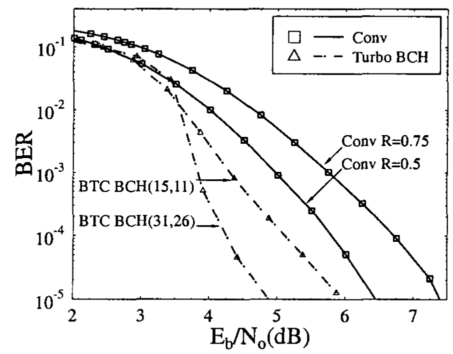

Figure 2: Decoding performance of the $R = 0.5$ and $R = 0.75$ convolutional decoder and the $R = 0.58$ BTC decoder using the BCH(15,11) code and the $R = 0.722$ BTC decoder employing the BCH(31,26) code after 8 **decoding iterations**, over the non-dispersive Gaussian channel.

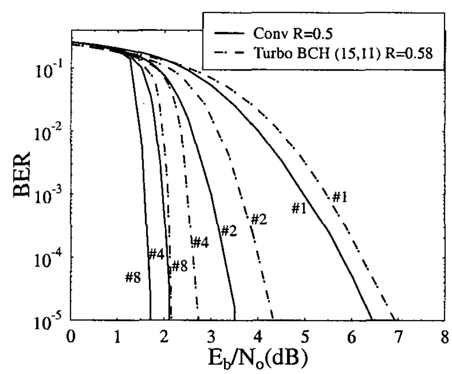

Figure 3: Comparing the turbo equalisation performance of the $R = 0.5$ convolutional decoder and the $R = 0.58$ BTC decoder using the BCH(15,11) code for 1, 2, 4 and 8 **turbo equalisation iterations**, over the non-dispersive Gaussian channel.

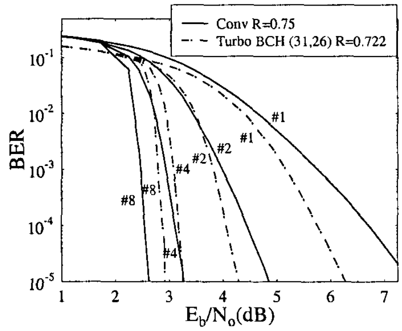

Figure 4: Comparing the turbo equalisation performance of the $R = 0.75$ convolutional decoder and the $R = 0.722$ BTC decoder using the BCH(31,26) code for 1, 2, 4 and 8 **turbo equalisation iterations**, over the non-dispersive Gaussian channel.

approximately 0.75 dB better than that of the $R = 0.5$ DEC-CONV system at BER= 10^{-4}. By contrast, an improvement of 2.5 dB is achieved by the $R = 0.722$ DEC-BTC system using the BCH (31,26) code after 8 **decoding iterations** compared to the $R = 0.75$ DEC-CONV at BER= 10^{-4}, as shown in Figure 2.

However, when turbo equalisation was employed, we observed in Figure 3 that the BER performance of the $R = 0.5$ TEQ-CONV system was better by approximately 0.4 dB compared to the $R = 0.58$ TEQ-BTC system after 8 **turbo equalisation iterations**. Figure 4 shows that the $R = 0.75$ TEQ-CONV system also has a gain of 0.4 dB over the $R = 0.722$ TEQ-BTC system after 8 **turbo equalisation iterations**. From Table 3, we can see that the iteration gains of the $R = 0.5$ TEQ-CONV and $R = 0.58$ TEQ-BTC systems are similar, whereas the $R = 0.75$ TEQ-CONV system has higher iteration gains compared to the $R = 0.722$ TEQ-BTC system.

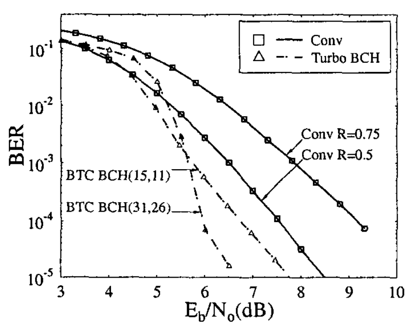

Figure 5: Decoding performance of the $R = 0.5$ and $R = 0.75$ convolutional decoder and the $R = 0.58$ BTC decoder using the BCH(15,11) code, as well as that of the $R = 0.722$ BTC decoder using the BCH(31,26) code after 8 **decoding iterations** over the 5-path fading channel of Table 1.

The decoding performance of the DEC-CONV and DEC-BTC systems over the equally-weighted and symbol spaced 5-path Rayleigh fading channel also showed the same performance trend, as observed over the non-dispersive Gaussian channel scenario. The $R = 0.58$ DEC-BTC scheme requires 0.75 dB lower E_b/N_o than the $R = 0.5$ DEC-CONV system at BER= 10^{-4}, while the $R = 0.722$ DEC-BTC system has a 3 dB gain over the $R = 0.75$ DEC-CONV system at BER= 10^{-4} as shown in Figure 5. For turbo equalisation over the equally-weighted 5-path fading channel, the required E_b/N_o for the system employing $R = 0.5$ TEQ-CONV is approximately 0.4 dB lower, than that of the $R = 0.58$ TEQ-BTC at BER= 10^{-4} after 8 **turbo equalisation iterations**. By contrast, the $R = 0.75$ TEQ-CONV system required an approximately 0.75 dB lower E_b/N_o value than the $R = 0.722$ TEQ-BTC system at

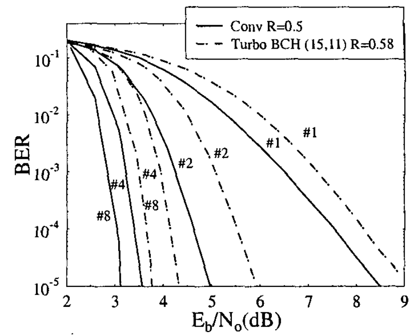

Figure 6: Comparing the turbo equalisation performance of the $R = 0.5$ convolutional decoder and the $R = 0.58$ BTC decoder using the BCH(15,11) code for 1, 2, 4 and 8 **turbo equalisation iterations**, over the 5-path fading channel of Table 1.

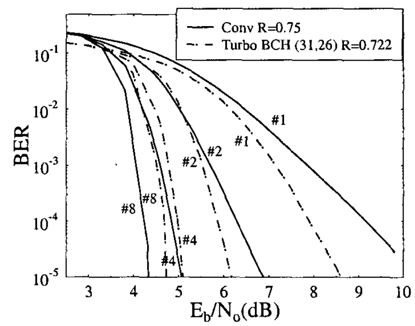

Figure 7: Comparing the turbo equalisation performance of the $R = 0.75$ convolutional decoder and the $R = 0.722$ BTC decoder using the BCH(31,26) code for 1, 2, 4 and 8 **turbo equalisation iterations**, over the 5-path fading channel of Table 1.

Code Rate $R = 0.75$
G0: 1 0 1
G1: 1 1 0
1 = transmitted bit
0 = non transmitted bit

Table 2: DVB puncturing pattern [11] applied to the coded bits of the $R = 0.5$ convolutional code in order to obtain a rate $R = 0.75$ convolutional code.

	$R = 0.5$ CONV	$R = 0.58$ BTC
Itr = 2	2.5 dB	2.3 dB
Itr = 4	3.7 dB	3.5 dB
Itr = 8	4.5 dB	4.2 dB
	$R = 0.75$ CONV	$R = 0.722$ BTC
Itr = 2	2.0 dB	1.8 dB
Itr = 4	3.4 dB	2.6 dB
Itr = 8	4.0 dB	3.0 dB

Table 3: The iteration (Itr) gains relative to the first iteration for the $R = 0.5$ and $R = 0.75$ TEQ-CONV and $R = 0.58$ and $R = 0.722$ TEQ-BTC systems over the non-dispersive Gaussian channel.

	$R = 0.5$ CONV	$R = 0.58$ BTC
Itr = 2	2.8 dB	2.2 dB
Itr = 4	4.1 dB	3.9 dB
Itr = 8	4.4 dB	4.4 dB
	$R = 0.75$ CONV	$R = 0.722$ BTC
Itr = 2	2.7 dB	2.1 dB
Itr = 4	4.3 dB	3.0 dB
Itr = 8	4.8 dB	3.3 dB

Table 4: The iteration (Itr) gains relative to the first iteration for the $R = 0.5$ and $R = 0.75$ TEQ-CONV and the $R = 0.58$ and $R = 0.722$ TEQ-BTC systems over the equally-weighted and symbol spaced 5 path fading channel.

BER= 10^{-4} after 8 **turbo equalisation iterations**, as shown in Figures 6 and 7, respectively. As in the non-dispersive Gaussian channel scenario, we observed from Table 4, that the iteration gains of the $R = 0.5$ TEQ-CONV and $R = 0.58$ TEQ-BTC systems over the equally-weighted 5-path fading channel were similar whereas the $R = 0.75$ TEQ-CONV system achieved higher iteration gains compared to the $R = 0.722$ TEQ-BTC system.

In summary, we observed that in the context of turbo equalisation the system employing convolutional codes (TEQ-CONV) has a better performance in terms of iteration gain and BER, than that using BCH turbo codes (TEQ-BTC). Conversely, when only decoding is performed, the system using convolutional codes (DEC-CONV) is poorer than that employing BCH turbo codes (DEC-BTC). Despite the investment in additional complexity in the form of the BTC decoder in the TEQ-BTC system, we observed that the system using the lower complexity convolutional codes has a better performance in terms of mitigating the effects of ISI introduced by the modulator - as in the case of partial-response GMSK modulation - and by the dispersive channel.

4. CONCLUSION

We have compared the decoding performance of the partial-response GMSK system, which employs BCH turbo codes (DEC-BTC) to that using convolutional codes (DEC-CONV). The GMSK turbo equalisation system using BCH turbo codes (TEQ-BTC) was also compared to the non-iterative convolutional coded GMSK turbo equalisation system (TEQ-CONV). We observed that for non-iterative equalisation followed by independent channel decoding, the lower complexity DEC-CONV system performs more poorly than the DEC-BTC system. However, when employing turbo equalisation, the lower complexity TEQ-CONV system performs better than the TEQ-BTC system in mitigating the effects of ISI due to the partial-response GMSK modulator and the dispersive channel.

Acknowledgements

The financial support of the Commission of European Community; Motorola ECID, Swindon, UK; EPSRC, UK is gratefully acknowledged.

5. REFERENCES

[1] C.Douillard, A.Picart, M.Jézéquel, et al, "Iterative correction of intersymbol interference: Turbo-equalization," *European Transactions on Communications*, vol. 6, pp. 507–511, September-October 1995.

[2] A. Hocquenghem "Codes correcteurs d'erreurs" *Chiffres (Paris)*, vol. 2, pp. 147–156, September 1959.

[3] R.C.Bose and D.K.Ray-Chaudhuri "On a class of error-correcting binary group codes" *Information and Control*, vol. 3, pp. 68–79, March 1960.

[4] K.Hirade and K.Murota "GMSK modulation for digital mobile radio telephony" *IEEE Transactions On Communications*, vol. 29, pp. 1044-1050, July 1981.

[5] Michael J.Gertsman and John H.Lodge, "Symbol-by-symbol MAP demodulation of CPM and PSK signals on Rayleigh flat-fading channels," *IEEE Transactions on Communications*, vol. 45, pp. 788–799, July 1997.

[6] G. Bauch and V. Franz, "Iterative Equalisation and Decoding for the GSM-System," in *IEEE 48th Vehicular Technology Conference*, pp. 2262–2266, 1998.

[7] D.Raphaeli and Y.Zarai, "Combined turbo equalization and turbo decoding," *IEEE Communications Letter*, vol. 2, pp. 107–109, April 1998.

[8] Ramesh Pyndiah, "Iterative Decoding of Product Codes: Block Turbo Codes," *Proceedings of the International Symposium on Turbo Codes & Related Topics*, pp. 71–79, Sept 1997. Brest, France.

[9] Patrick Robertson, Emmanuelle Villebrun and Peter Hoeher, "A Comparison of Optimal and Sub-Optimal MAP Decoding Algorithms Operating in the Log Domain," *Proceedings of the International Conference on Communications*, pp. 1009–1013, June 1995.

[10] L.R. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal Decoding of Linear Codes for Minimising Symbol Error Rate," *IEEE Transactions on Information Theory*, pp. 284–287, March 1974.

[11] ETSI, "Digital Video Broadcast (DVB); Framing structure, channel coding and modulation for 11/12 GHz Satellite Services," *EN 300 421 V1.1.2*, pp. 12–14, August 1997.