COMBINED TRELLIS–CODED MODULATION AND BLIND TURBO EQUALISATION

S. Vlahoyiannatos, S. X. Ng, L. Hanzo

Dept. of Electronics and Computer Science,
University of Southampton, SO17 1BJ, UK.
Tel: +44-23-8059 3125, Fax: +44-23-8059 4508
Emails: {sv97r, sxn99r, lh}@ecs.soton.ac.uk
http://www-mobile.ecs.soton.ac.uk

ABSTRACT

In this contribution, the Per-Survivor Processing (PSP) based blind turbo equalization is combined with various coded modulation schemes, which employ combined modulation and channel coding techniques. The proposed method exploits the enhanced data protection offered by either Trellis–Coded Modulation (TCM), Bit–Interleaved Coded Modulation (BICM) or Turbo–TCM (TTCM) and exhibits good performance in terms of its output Bit Error Rate (BER). Explicitly, a BER comparable to that of a trained turbo equaliser is attained at the cost of a modest complexity increase.

1. INTRODUCTION

Blind equalization has attracted significant research interests during recent years. The blind equalizer proposed in this contribution belongs to the class of sequence estimation techniques. It incorporates a Per-Survivor Processing (PSP) based equalizer [1], modified for producing soft outputs, as in [2, 3] and it involves channel coding not only to protect the transmitted data from the channel's effects, but also to assist the PSP equalizer during its convergence by utilizing a feedback loop.

The paper is organised as follows. The communications system is described in Section 2, while in Section 3 the proposed coded modulation assisted turbo-PSP equaliser is described. Specifically, in Section 4 the encoder and decoder used in the coded modulation schemes are detailed. Finally, in Section 5 performance results are provided for both static and fading dispersive channels.

2. SYSTEM DESCRIPTION

The communications system under consideration is shown in Figure 1. The information bits are mapped to QAM symbols by a channel encoder, which is a trellis coded modulation encoder [4], a turbo trellis coded modulation encoder [5] or a convolutional encoder in the case of BICM [6]. The QAM symbols are then interleaved, in order to disperse the channel's bursty errors as well as to enhance the turbo–equaliser's performance, generating the transmitted QAM symbols $a(n)$. These symbols are convolved with the Channel's Impulse Response (CIR) h_i and then the channel noise
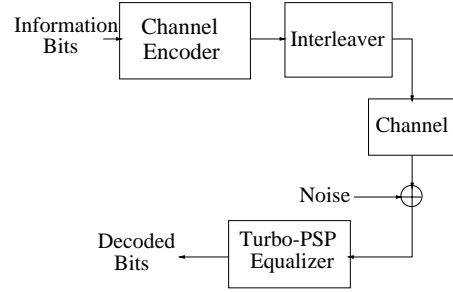

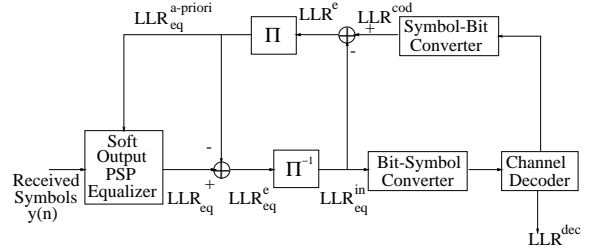Figure 1: Equalised communications system

Figure 2: The turbo–PSP equaliser

$e(n)$ is added, yielding the received symbols $y(n)$ as:

$$y(n) = \sum_{i=-L_1}^{L_2} h_i \cdot a(n-i) + e(n). \qquad (1)$$

The restoration of the original information bits is performed by the turbo–PSP equaliser. In the next section we provide further details concerning the operation of the turbo–PSP equaliser using TCM, TTCM and BICM.

3. TURBO–PSP EQUALISER DESCRIPTION

The turbo–PSP equaliser performs joint channel equalisation and coded modulation decoding using the schematic shown in Figure 2. The operation of this system is the same as that of the turbo–PSP equaliser using convolutional coding [7], where the *LogLikelihood Ratio* (LLR) values are defined as:

$$LLR(Bit) = ln\left[\frac{Prob(Bit=1)}{Prob(Bit=0)}\right]. \qquad (2)$$

The sole difference is the replacement of the convolutional decoder by a TCM decoder. Hence, a bit–to–symbol converter is placed before the TCM decoder on order to convert the LLR values to symbol probabilities, which are necessary for facilitating TCM decoding. Similarly, a symbol–to–bit converter is employed at the output of the decoder. The calculation of the symbol probabilities from the probabilities of the input bits is based on the following formula:

$$Prob(Symbol = A_i) = \Pi_j Prob(Bit_j(Symbol) = Bit_j(A_i)),$$
$$i = 0, \cdots, Q, \quad j = 0, \cdots, K, \qquad (3)$$

where Q is the number of symbols in the modulation constellation used, K is the number of bits per symbol, A_i represents the constellation symbols and the function $Bit_j(A_i)$ returns the value of the j–th bit of symbol A_i. The assumption implicitly stipulated here is that the bits of a symbol are independent of each other. This, however, is not a valid assumption, since channel coding has deliberately imposed a certain amount of correlation or interdependence on the bits, in order to exploit this redundancy for correcting transmission errors. Hence the symbol probability calculation is somewhat inaccurate. Fortunately the effects of this inaccuracy are mitigated by the iterative turbo equalizer. Similarly to the bit–to–symbol conversion of Equation 3, the symbol–to–bit conversion is based on the following formula:

$$Prob(Bit_j = Bit) = \sum_i Prob(Symbol_i; Bit_j(Symbol_i) = Bit).$$
$$(4)$$

This formula is accurate in the sense that it does not require any assumptions concerning the correlation of the input symbols. However, the coding-imposed correlation amongst the bits of a symbol would have already been lost at the bit–to–symbol converter of Figure 2 and hence the input of the symbol–to–bit converter is uncorrelated.

Having described the basic functions of the component modules of the turbo–PSP equaliser in Figure 2, we will now proceed to highlighting the operation of these components. The PSP equaliser is the same module as the one used in [7]. Therefore, we will not elaborate on it further in this paper. In the next section we describe the operation of the coded modulation encoder and decoder.

4. CODED MODULATION SCHEMES

Trellis Coded Modulation (TCM) [4] was proposed originally for Gaussian channels and it was later further developed for applications in mobile communications [8]. Turbo Trellis Coded Modulation (TTCM) [5] is a more recent joint coding and modulation scheme that has a structure similar to that of the family of power efficient binary turbo codes [2], but employs TCM schemes as component codes. Both TCM and TTCM use symbol–based interleavers and Set–Partitioning (SP) based signal labeling. Another coded modulation scheme distinguishing itself by utilising bit–based interleaving in conjunction with Gray signal constellation labeling is referred to as Bit–Interleaved Coded Modulation (BICM) [6]. The number of parallel bit–interleavers equals the number of coded bits in a symbol for the BICM scheme proposed in [6].

4.1. TCM Turbo–PSP

The TCM decoder operates similarly to the corresponding module of a trained turbo equaliser [3]. A non–binary MAP decoder is invoked for the channel decoder module. Given the LLR value $LLR(Bit)$ of a binary bit, we can calculate the probability of $Bit = +1$ or $Bit = -1$ as follows. Remembering that $Prob(Bit = -1) = 1 - Prob(Bit = +1)$, and taking the exponent of both sides in Equation 2 we can write:

$$e^{LLR(Bit)} = \frac{Prob(Bit = +1)}{1 - Prob(Bit = +1)}. \qquad (5)$$

Hence we have:

$$Prob(Bit = 1) = \frac{e^{LLR(Bit)}}{1 + e^{LLR(Bit)}}, \qquad (6)$$

and,

$$Prob(Bit = -1) = \frac{1}{1 + e^{LLR(Bit)}}. \qquad (7)$$

Then, assuming that the bits of a symbol are independent of each other, the probability of a $Symbol$ which is represented by the bits $Bit^1, \ldots, Bit^n$, can be calculated as in Equation 3, where we have $Symbol \in (0, \ldots, 2^n - 1)$ for the 2^n–array modulation scheme used. The probability of the transition from states s' to state s, commonly defined as $\gamma(s', s)$, used in the non–binary MAP decoder [2], can be calculated as:

$$\gamma(s', s) = \eta_{Symbol} \cdot \prod_{i=1}^{i=n} Prob(Bit^i), \qquad (8)$$

where $Symbol$ is the trellis transition branch label from state s' to state s and η_{Symbol} is the "a priori" information of the $Symbol$. Then the α and β values can be obtained from:

$$\alpha_k(s) = \sum_{s'} \gamma_k(s', s) \cdot \alpha_{k-1}(s') \qquad (9)$$

$$\beta_{k-1}(s') = \sum_s \gamma_k(s', s) \cdot \beta_k(s). \qquad (10)$$

The number of transitions emerging from a specific trellis state is equal to 2^k, where k is the number of information bits per n–bit modulation symbol. The coding rate used is $R = \frac{k}{n}$, where $k = n - 1$. Therefore the log–MAP decoder used is non–binary, when $k > 1$. By contrast, if $k = 1$, then the number of transitions emerging from a trellis state is equal to $2^1 = 2$, i.e. a binary MAP decoder is used. The a–posteriori probability (APP) of the information symbol u_t, $u_t \in (0, \ldots, 2^k - 1)$ at time instant t can be computed from [2] as:

$$APP(u_t) = \sum_{s' \to s, u_t} \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s). \qquad (11)$$

The final decoded information symbol at instant t is the hard decision based symbol generated from these APP values. However, we have to feed back the LLR values of all the n coded bits, rather than just those of the k information bits, to the PSP equaliser, after improving their reliability by the channel decoder. The APP of the coded symbol x_t, $x_t \in (0, \ldots, 2^n - 1)$ at time instant t can be computed from:

$$APP(x_t) = \sum_{s' \to s, x_t} \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s), \qquad (12)$$
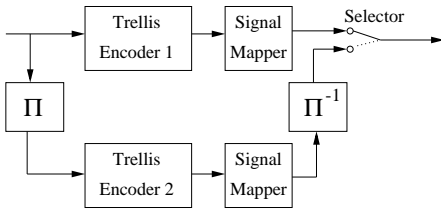
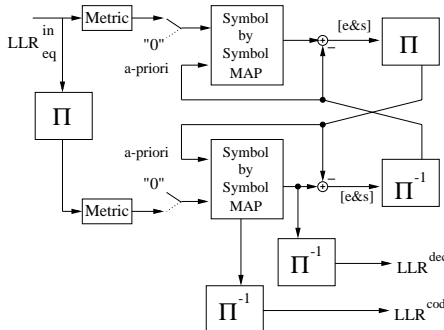Figure 3: Turbo trellis–coded modulation encoder.

Figure 4: TTCM decoder.

while Equation (11) formulated the APP of the original encoded information symbol. The probability of bit i to be 1 in a coded symbol x is calculated from:

$$Prob(Bit^i = 1) = \sum_{x=0}^{x=2^n-1} APP(x^i = 1), \qquad (13)$$

where x^i denotes the binary value at bit position i of the symbol x, $x^i \in (0,1)$ and in verbal terms the probability of $Bit^i = 1$ is given by the sum of the probabilities of all symbols from the set of $2^n - 1$ number of phasors, which have a binary 1 at bit position i. A similar procedure is invoked for determining $Prob(Bit^i = 0)$ and finally the LLR of the bits can be computed from Equation 2.

4.2. TTCM Turbo–PSP

An extension of the turbo–PSP equaliser employing Turbo Trellis–Coded Modulation (TTCM) has also been considered. TTCM was proposed in [5], where the information bits are sent only once, while the parity bits are provided alternatively by the two constituent TCM encoders. The TTCM encoder is shown in Figure 3, which comprises two identical TCM encoders linked by the symbol interleaver Π. The TTCM decoder structure shown in Figure 4 is similar to that of binary turbo codes, except for the difference in the nature of the information passed from one decoder to the other. Each decoder alternately processes its corresponding encoder's channel impaired output symbol, and then the other encoder's channel impaired output symbol. The information bits, i.e. the systematic bits, are constituted by the corresponding systematic TCM encoder's output bits received over the channel in both cases. The systematic information and the parity information are transmitted together in the same symbol. Hence, the systematic information component cannot be separated from the extrinsic information, since the noise that affects the parity component of a TTCM symbol also affects the systematic

information component. The output of each symbol–based MAP decoder of Figure 4 can be split into two components [2]:

1. the *a priori* component, and

2. the amalgamated (extrinsic and systematic) [e&s] component.

Each decoder of Figure 4 has to pass only the [e&s] component to the other decoder, which is written in parentheses in order to emphasize the inseparability of the extrinsic and systematic components. The reason the a–priori component is subtracted from the output of the symbol–based MAP decoder of Figure 4 is that this information component was generated by the other decoder and hence it must not be fed back to it. Otherwise the probability estimates of the two decoders become dependent on each other, preventing the iterative enhancement of the decoder's decision reliability. The LLR_{eq}^{in} output of the equaliser is forwarded to the "metric" calculation block of Figure 4, in order to generate a set of 2^n symbol reliabilities. The selectors, in front of the Symbol by Symbol MAP decoder of Figure 4, select the current symbol's reliabilities from the "metric" calculation block, if the current received symbol corresponds to that component decoder, otherwise depuncturing will be applied, where the reliabilities of the symbols are set to a "0" corresponding to no a–priori information. The "metric" calculation block provides the decoder with the parity and systematic [p&s] information, and the second input to the symbol by symbol MAP decoder of Figure 4 is the *a priori* information acquired from the other decoder. The MAP decoder then provides the *aposteriori* information constituted by the (*a priori*+[e&s]) components as the output. Then the *a priori* information is subtracted from the *a posteriori* information, again so that information is not used more than once in the other decoder. The resulting [e&s] information is interleaved (or de–interleaved) in order to create the *a priori* input of the other decoder. This decoding process will continue iteratively, in order to generate an improved version of the set of symbol reliabilities for the other decoder. One iteration comprises the decoding of the received symbols by both of the component decoders. Finally, the *aposteriori* information of the lower component decoder of Figure 4 will be de–interleaved, in order to extract $(n-1)$ decoded information bits per symbol. On the other hand, the *aposteriori* information of the n coded bits is de–interleaved, in order to convey the LLR^{cod} signal of Figure 2 to the equaliser's input.

4.3. BICM Turbo–PSP

The major difference between BICM [6] in comparison to TCM [4] is that bit-based interleaving is used and non–systematic convolutional codes are utilised for protecting the information. The decoding of BICM is similar to that of TCM, as it was highlighted in Section 4.1.

These two coded modulation schemes exhibit a similar coding rate, depending on the modulation mode used. Specifically, a rate–1/2 code is used for QPSK and a rate–2/3 code is employed for 8–PSK. In our investigations, Gray signal constellation labelling is invoked, since we found that the performance of the combined system was better, when using Gray-coded constellation labelling instead of set-partitioning.

Having described the most important components of the proposed coded modulation based turbo–PSP equaliser, we
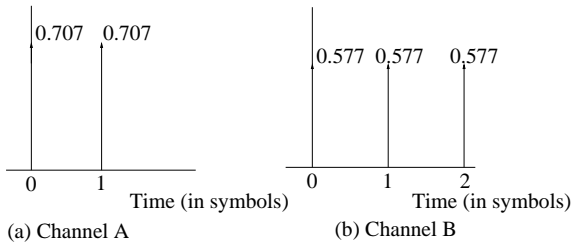
Figure 5: The CIRs used in our simulations

Frame length (bits)	174
Interleaver block length	5x174
Carrier frequency	1900MHz
Symbol rate	2.6MBaud
Doppler speed	48Km/h
Equaliser step–size	$5x10^{-3}$
Convergence control	D2 [7]
TCM memory	6
TTCM memory	3
BICM memory	6
TTCM number of iterations	4

Table 1: The turbo–PSP equaliser parameters used in the simulations. Observe in the table that for the shake of a fair comparison between TCM, TTCM and BICM, the code memory was 6, 3 and 6 respectively, since TTCM used four iterations, which resulted in a similar complexity for all three schemes.

will now provide performance results characterising the various schemes, which were obtained by means of computer simulations.

5. PERFORMANCE RESULTS

In this section we present performance results for the turbo PSP–equaliser described in Section 3. The results are based on computer simulations. The channels assumed are static or fading, having the CIRs given in Figure 5. The convergence control is assumed to be D2, as it was defined in [7] where the variance of the difference between the previous and current equalizer output LLRs is measured and compared against a fixed threshold. The general turbo–PSP equaliser parameters are given in Table 1. In Figure 6 we have plotted the coding gain of the various techniques at a BER of $5 \cdot 10^{-5}$ for QPSK for transmissions over the static 2–path channel of Figure 5 versus their relative complexity. It has been assumed that the complexity of the PSP equaliser is approximately the same for all the algorithms and therefore the only factor that affects the total complexity is the complexity of the channel decoder. In Figures 7 and 8 the BER versus Bit SNR curves are given for QPSK and 8–PSK, over the fading channels of Figure 5. The Bit SNR is defined as:

$$\text{Bit SNR} = \frac{E_a}{\text{IBPS} \cdot E_e}, \qquad (14)$$

where E_a is the average QAM signal power, E_e is the average noise power and IBPS is the number of information bits per symbol. This definition assists in the realistic assessment of the benefits of using channel coding for protecting the data.

We observe from Figure 6 that the coding gain is the highest for BICM at a lower decoding complexity for static dispersive channels. When the complexity is increased, then the coding gain of TTCM becomes marginally better. This happens at a coding gain of approximetaly 5.7dB. In the fading channels, BICM exhibits the worst performance. The performance of BICM is only marginally lower than that of TCM or TTCM in conjunction with QPSK. However, BICM performs significantly worse, than TCM and TTCM, when 8–PSK modulation is employed. TCM is found to perform the best for the QPSK modulation and marginally worse than TTCM, when 8–PSK is employed over the dispersive fading channels having the CIR of Figure 5.

The important trend we inferred from our simulations is the superiority of BICM for transmissions over static channels, but this trend is reversed for fading channels. Essentially BICM is a binary version of TCM, which is related to the turbo–PSP algorithm of [7]. While TCM uses set–partitioning for maximising the Euclidean distance of the constellation points represented by the uncoded bits of a symbol, BICM is concerned only with bits, not with symbols. However, the PSP equaliser exhibits a degraded performance, when using no Gray coding, since then the neighbouring symbols of the constellation can have more than one different bits. With the aid of simulations we found that the PSP equaliser's performance degradation encountered due to using no Gray coding was higher, than the performance degradation incurred when using Gray–coded rather than SP-based TCM. Thus in our simulations Gray–coded rather than SP-based modulation was used in conjunction with TCM, TTCM and BICM. This is one of the reasons for which TCM appears to perform worse than BICM in static channel scenarios. A further reason of the performance degradation of TCM in comparison with BICM over static, but dispersive, channels is due to using bit LLRs in the turbo–PSP equaliser. In fact, a TCM turbo–PSP equaliser would perform better, if the bit probabilities represented by the LLRs were replaced by symbol probabilities. However, this would dramatically increase the associated complexity, since it would mean that at each symbol instant the signal in the turbo–PSP device would be represented by an increased number of values, which is equal to the number of symbols in the QAM constellation, rather than the number of bits per symbol. Therefore, if K is the number of bits per symbol, we would have a factor of $2^K/K$ more LLR values to process. In order to avoid this complexity problem, we have used bit LLR values, just as in the case of the turbo–PSP of [7]. As we have mentioned before, this is not optimum in conjunction with symbol–based coding techniques, such as TCM and TTCM, because we assumed that the probability of a symbol is given by the product of the probabilities of its constituent bits, which ignores the correlation amongst these bits, as it was highlighted in Section 2.

6. SUMMARY

In this paper a novel blind equaliser was proposed, extending the turbo–PSP algorithm of [7] by replacing the separate convolutional channel coding scheme with coded modulation techniques. The idea originated from the fact that by combining modulation and channel coding the turbo–PSP
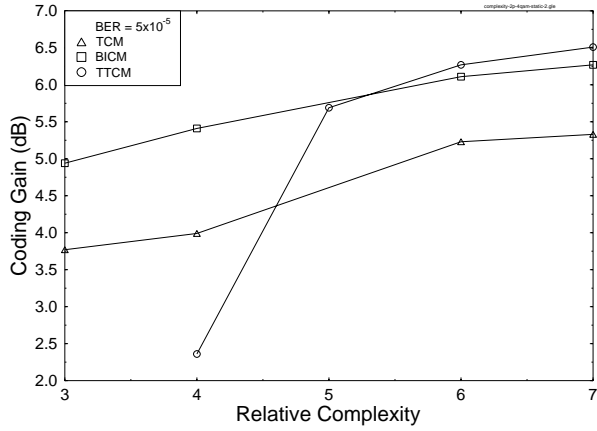
Figure 6: The coding gain of TCM, TTCM and BICM at a BER of $5 \cdot 10^{-5}$ versus complexity for transmissions over the **static** 2–path channel of Figure 5, employing **QPSK**.
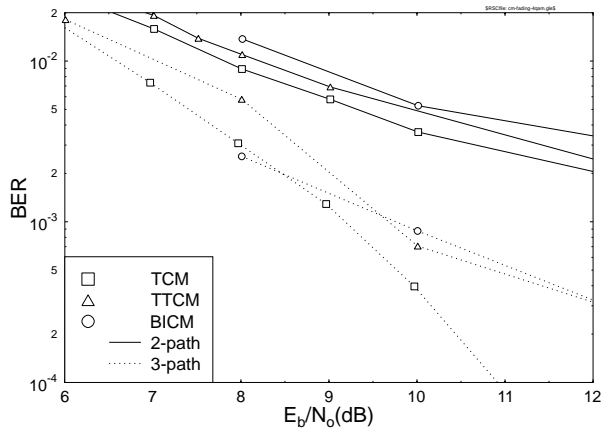

Figure 7: BER versus Bit SNR curves over **fading channels**, exhibiting the impulse response shown in Figure 5 and using the turbo–PSP equaliser parameters given in Table 1 for **QPSK** modulation.
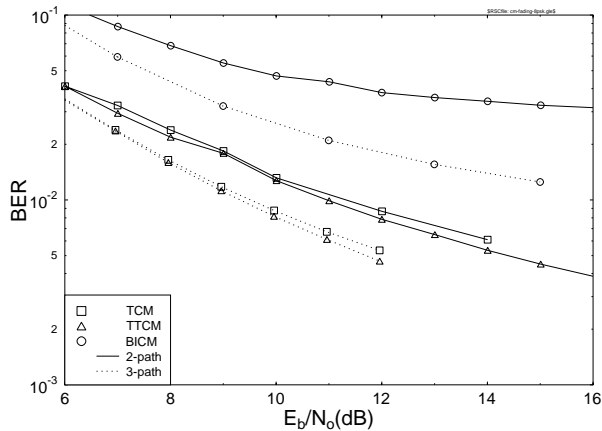

Figure 8: BER versus Bit SNR curves over **fading channels**, exhibiting the impulse response shown in Figure 5 and using the turbo–PSP equaliser parameters given in Table 1 for **8–PSK** modulation.

equaliser would become more efficient, since it is a symbol–based detection technique. However, three main calamities exist in the context of this technique. On the one hand, when the symbol probabilities replace the bit LLR values, then the complexity and storage requirements increase dramatically, especially in the context of higher–order QAM schemes. Thus the turbo equaliser was modified for using bit–based LLR values for communicating amongst the detection blocks of Figure 2. This gave rise to the second main problem. Specifically, the conversion of the bit LLR values to symbol probabilities was based on the assumption that the bits of a symbol are independent even after channel encoding. However, this assumption imposes a performance degradation on the TCM and TTCM decoders, since the symbol probabilities became somewhat inaccurate, although this was partially compensated by the iterative TTCM scheme. Finally, the employment of Gray–coded modulation, which was preferable in terms of the overall system's performance, gave rise to a performance degradation of the TCM and TTCM decoders, since these schemes were optimised for set–partitioning based mapping of the bits to the symbols.

Nevertheless, in Figure 6 we found that the TTCM scheme slightly outperformed the BICM arrangement, which was similar to the convolutional coding based scheme used in [7] in the context of QPSK and static channels. By contrast, in fading environments TCM emerged as the best performer from the set of schemes studied.

7. REFERENCES

[1] N. Seshadri, "Joint data and channel estimation using blind trellis search techniques," *IEEE Transactions on Communications*, vol. COM-42, pp. 1000–1011, February-April 1994.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and encoding: Turbo-codes (1)," in *IEEE International Conference on Communications*, (Geneva, Switzerland), pp. 1064–1070, 23–26 May 1993.

[3] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo-equalization," *European Transactions on Telecommunications and Related Technologies*, vol. 6, pp. 507–511, September–October 1995.

[4] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Transactions on Information Theory*, vol. IT-28, pp. 55–67, January 1982.

[5] P. Robertson and T. Worz, "Bandwidth-Efficient Turbo Trellis-Coded Modulation Using Punctured Component Codes," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 206–218, February 1998.

[6] E. Zehavi, "8-PSK trellis codes for a Rayleigh fading channel," *IEEE Transactions on Communications*, vol. 40, pp. 873–883, May 1992.

[7] S. Vlahoyiannatos and L. Hanzo, "Blind psp–based turbo equalization," *Proceedings of the IEEE Vehicular Technology Conference 2001*, May 2001.

[8] D. Divsalar and M. K. Simon, "The design of trellis coded MPSK for fading channel: Performance criteria," *IEEE Transactions on Communications*, vol. 36, pp. 1004–1012, September 1988.