BLIND PSP-BASED TURBO EQUALIZATION

S. Vlahoyiannatos, L. Hanzo

Dept. of Electr. and Comp. Sc., Univ. of Southampton, SO17 1BJ, UK.
Tel: +44 23 80 593 125; Fax: +44 23 80 594508
Email: lh@ecs.soton.ac.uk; http://www-mobile.ecs.soton.ac.uk

ABSTRACT

A plethora of approaches have been proposed for blind equalization to date. In this contribution a new blind equalizer is proposed, extending the concept of trained turbo equalizers to the blind scenario by using Per-Survivor Processing (PSP). This equalizer exploits the enhanced data protection offered by channel coding and exhibits good performance in terms of its output Bit Error Rate (BER). Explicitly, a BER comparable to that of a trained turbo equalizer is attained at the cost of a modest complexity increase.

1. INTRODUCTION

Blind equalization has attracted significant research interests in 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 to produce soft outputs as in [2, 3, 4] and it involves channel coding not only to protect the transmitted data from the additive noise inflicted by the channel, but also to assist the PSP equalizer to converge by utilizing a feedback loop.

The paper is organized as follows. The communications system is described in Section 2, while in Section 3 the proposed turbo-PSP equalizer is highlighted. In Section 4 the associated phase ambiguity problem is addressed. 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 encoded by a convolutional encoder and then punctured in order to produce the necessary coding rate. The coded bits are then interleaved in order to disperse the channel's bursty errors as well as to enhance the turbo-equalizer's performance and then mapped to the QAM constellation symbols $a(n)$. These symbols are convolved with

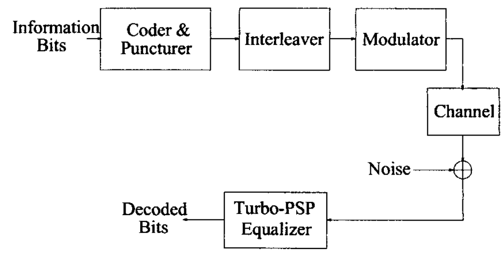

Figure 1: Equalized communications system

the Channel's Impulse Response (CIR) h_i and then the channel noise $e(n)$ 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 equalizer. This equalizer performs joint blind channel equalization and channel decoding in an iterative fashion. We will describe the operation of the turbo-PSP equalizer in the next section.

3. TURBO-PSP EQUALIZER DESCRIPTION

The turbo-PSP equalizer performs joint channel equalization and decoding using the schematic shown in Figure 2. In this figure the symbol Π denotes the interleaver. The system consists of a PSP [1] equalizer, providing the so-called *LogLikelihood Ratio* values LLR_{eq} for the subsequent de-interleaver, where the LLRs are defined as:

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

The operating principle of the system is as follows. Firstly, the PSP equalizer attempts to remove the Intersymbol Interference (ISI) from the received signal

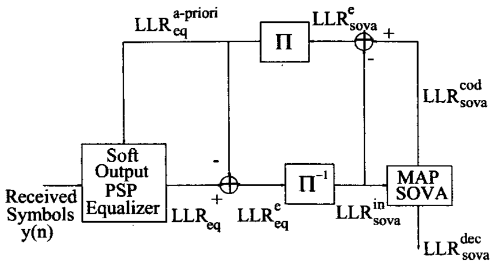

Figure 2: The turbo-PSP equalizer

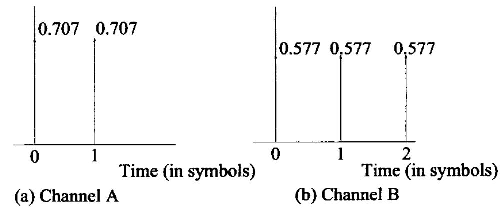

Figure 3: The CIRs used in our simulations

and at the same time performs demodulation, providing the bit LLR values, which are input to the channel decoder. Subsequently, the channel decoder improves the soft estimates of the PSP equalizer's output, which is then used as "a-priori" information for the next iteration of the PSP equalizer. Following a number of iterations, the impairments imposed by the channel are gradually removed from the signal.

Let us now discuss the turbo-PSP equalizer in more detail. The equalizer's a-priori LLR values, $LLR_{eq}^{a-priori}$ are used as a-priori knowledge, assisting the module in providing improved confidence values at its output. These values stem from the channel decoded information, which are then substracted from the equalizer's output LLR_{eq} in order to provide the input LLR_{eq}^{e} of the subsequent de-interleaver. This subtraction takes place for ensuring that in the next iteration the channel decoder takes into account only the equalizer's contribution but not the equalizer's a-priori values, which had been provided by the channel decoder itself. This concept is similar to that used in trained turbo equalization, although the lack of training information requires the replacement of the MAP equalizer by a PSP equalizer. Following the PSP-based equalization, the de-interleaver rearranges the LLR_{eq}^{e} values according to the interleaving algorithm used. Following the de-interleaver, the Soft Output Viterbi Algorithm (SOVA) performs channel decoding based on the *Maximum A-Posteriori (MAP)* criterion and provides the LLR_{sova}^{cod} as well as LLR_{sova}^{dec} values for its coded and decoded outputs, respectively. The coded LLR values are again interleaved so as to align them appropriately with the channel output symbols. The resultant LLR values $LLR_{eq}^{a-priori}$ are then fed back to the PSP equalizer as a-priori values.

4. DIFFERENTIAL CODING

Blind equalizers may converge to a tap setting, which results in a rotated phasor constellation in the complex plane, as compared to the ideal setting. Specifically, when QAM is employed, the symmetry of the modulation constellation implies that the blind equalizer can converge to four different settings, each corresponding to one of the quarters of the complex plane. A common solution to this problem inherent in any blind equalizer is the employment of differential coding. Seshadri [1] observed this phenomenon for the case of a PSP equalizer. For the turbo-PSP equalizer, however, this solution cannot be used, since differential encoding and decoding of soft values would have to be performed. However, the differential encoding of soft values results in converging to zero soft values. Thus, other techniques of overcoming this problem have to be used. We will describe an appropriate method below.

It can be observed that the equalizer typically converges to that specific setting, which it has been initialized closest to. If the initialization is close to the correct setting, then equalization will converge and in the next iteration the equalizer will be fed with a-priori values, which will improve its new CIR estimate and its soft output values. By contrast, if the equalizer's initialization is closer to a setting which is not the correct one, then it will start converging to this particular setting. However, the SOVA will be fed with phase-rotated values and in the next iteration the equalizer's a-priori values will drive the equalizer to instability. A way of overcoming this problem is to shift the estimated CIR in phase, so that it matches the actual CIR as closely as possible. The problem that arises is that of estimating the phase shift. This can be achieved by initially invoking the PSP equalizer without iterations and transmitting a few pilot symbols. As we will see in the next section, a low number of training pilot symbols is sufficient, resulting in an insignificant reduction of the bandwidth. After the selection of the right coordinate quadrant, equalization continues as before. The number of iterations is adjusted according to an estimate of the grade of convergence provided by the equalizer.

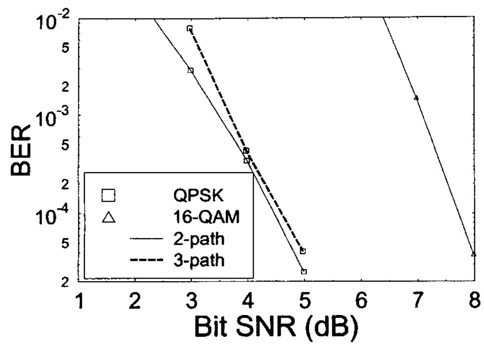

Figure 4: BER versus Bit SNR curves over static channels, exhibiting the impulse response shown in Figure 3 and using the turbo-PSP equalizer parameters given in Table 1. The iteration control obeyed D1.

5. PERFORMANCE RESULTS

In this section we present performance results for the turbo PSP-equalizer described in Section 3. The results are based on computer simulations. The channels assumed are static, having the CIRs given in Figure 3.

Iteration or convergence control has been considered in two different ways. In Detector 1 (D1) the variance of the input LLR values of the de-interleaver was measured and then its slope was estimated. This variance provides an estimate of our confidence in the LLR values, since the higher the absolute values of these LLRs, the more confident the decisions. The associated LLR variance is expected to increase iteration after iteration and reach its peak value, when the iterations can no longer improve the BER. This condition is detected by estimating the slope of the variance by utilizing the least squares method. According to D1 the iterations are curtailed, when the slope becomes less than a threshold value. According to our second detector (D2) the variance of the difference between the previous and current equalizer output LLRs is measured and compared against a fixed threshold. The iterations are curtailed, when the variance becomes less than the threshold. In our simulations D1 and D2 are evaluated both in terms of their performance and complexity reduction. The general turbo-PSP equalizer parameters are given in Table 1. In Figure 4 the BER versus Bit SNR curves are given for both QPSK and 16-QAM, over the static channels of Figure 3. The Bit SNR is defined as:

$$\text{Bit SNR} = \frac{E_a}{\text{IBPS} \cdot E_e}, \quad (3)$$

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, i.e. the number of convolutional coded bits per symbol. This definition assists in the realistic assessment of the benefits of using strong convolutional coding for protecting the data.

Frame length (bits)	174
Interleaver Type	Bit
Interleaver block length	5x174
Carrier Frequency	1900MHz
Symbol Rate	2.6MBaud
Doppler Speed	48Km/h
Code rate	1/2
Convolutional Polynomial 1	17_{OCT}
Convolutional Polynomial 2	13_{OCT}
Equalizer step-size	$5\text{x}10^{-3}$

Table 1: The turbo-PSP equalizer parameters used in the simulations.

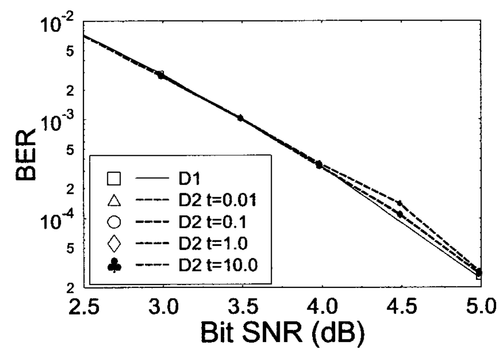

Figure 5: Comparison between D1 and D2 in terms of their BER versus Bit SNR performance. The channel is assumed to be static, obeying the impulse response of Figure 3(a) and the turbo-PSP equalizer parameters are given in Table 1. The variable t denotes the "threshold".

In Figure 5 we compare D1 and D2 in the context of QPSK over the static channels of Figure 3(a). Finally, in Figure 6 we portrayed the associated average number of iterations for both detectors D1 and D2. Firstly, we demonstrated in Figure 5 that the BER is not affected by the choice of D1 or D2. This implies that both detection methods assign a sufficient number of iterations for correct decoding. The number of iterations required by D1 is lower than that of D2 at low SNRs and higher than that of D2 at high SNRs. This can be readily interpreted, if we consider that the operation of D1 is not significantly affected by the noise, since it depends on the variance slope, which is in some sense an average, because it exhibits memory. In other words, a number

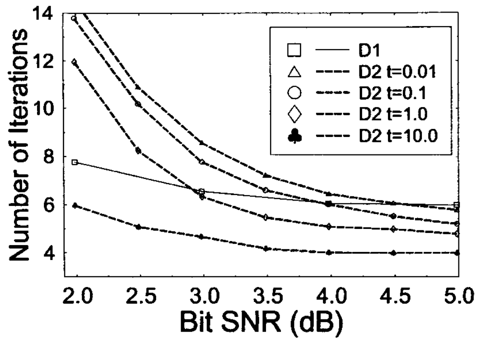

Figure 6: Comparison between D1 and D2 in terms of the average number of iterations versus Bit SNR curves. The channel is assumed to be static having the impulse response shown in Figure 3(a) and the turbo-PSP equalizer parameters given in Table 1. The variable t denotes the "threshold".

of previous points was taken into account in order to estimate it. This implies that a "standard" number of iterations will take place in any case. By contrast, D2 takes into account only the current value of a variance and therefore it detects convergence immediately, when this value becomes less than a threshold, independently of any previous values. At high SNRs the number of iterations required depends directly on the threshold value and convergence is typically achieved within a few iterations. At low SNRs, however, convergence is not readily obtained because of the excessive noise and hence typically a high number of iterations is necessary. In the case of D1, it may seem surprising that the required number of iterations increases with the SNR. However, this is expected to be so, since the noise assists D1 in detecting convergence earlier due to the randomness imposed by the noise on the LLRs' variance. In Figure 6 we observe that a lower D2 threshold requires more iterations, which was expected. We also observe in Figure 5 that the BER performance is good, approaching the performance of a turbo equalizer using perfect channel estimation for a sufficiently large number of iterations. This is because the PSP equalizer requires a considerable number of input symbols to converge, especially for higher-order QAM. The feedback loop of the PSP equalizer using the channel decoder's output information expedites convergence. The larger the number of iterations, the better the convergence of the PSP equalizer.

In Figure 7 we provide BER performance results for the *2*- and *3*-path fading channels using the parameters given in Table 1. The performance is again good compared to our benchmarker using perfect CIR estimation. It is anticipated that the performance becomes

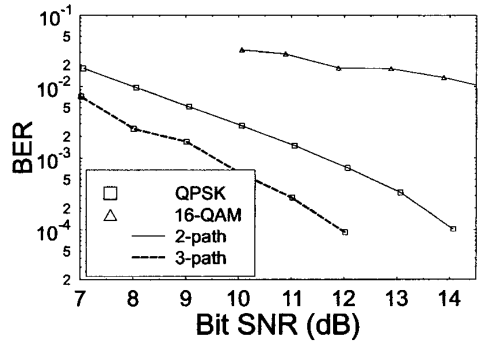

Figure 7: BER versus Bit SNR curves over fading channels having the impulse response shown in Figure 3 and using the turbo-PSP equalizer parameters given in Table 1.

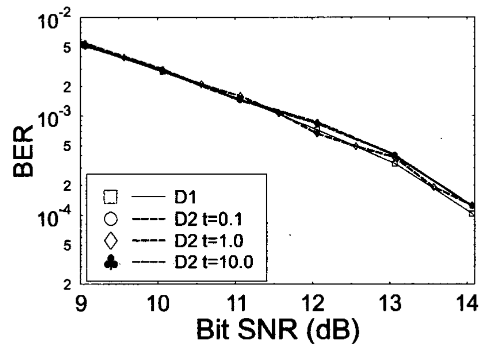

Figure 8: Comparison between D1 and D2 in terms of their BER versus Bit SNR curves. The channel is assumed to be Rayleigh fading, obeying the impulse response of Figure 3(a) and the turbo-PSP equalizer parameters are given in Table 1.

better over the 3-path channel, than that over the 2-path channel, since it is less likely for three paths to fade at the same time than for two paths. A comparison similar to that over the previously studied static channels is given for D1 and D2 in Figure 8.

Finally, in Figure 9 we portrayed the associated average number of iterations for comparison.

We observe again in Figure 8 that the associated BER performance is similar for D1 and D2, whilst the number of iterations is now clearly lower for D2, even at low SNRs. This is because in the previous static channel's scenario the channel's z-domain transfer function contained a zero on the unit circle, rendering convergence slower and less accurate. This resulted in a larger number of iterations required for the variance to be below the threshold value. In the fading channel's sce-

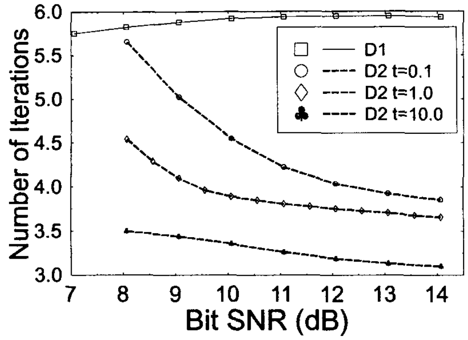

Figure 9: Comparison between D1 and D2 in terms of the required average number of iterations versus Bit SNR. The channel is assumed to be Rayleigh fading, using the impulse response of Figure 3(a) and the turbo-PSP equalizer parameters are given in Table 1. The variable t denotes the "threshold".

nario, such channel transfer functions do not occur very frequently, hence the associated convergence is typically faster. Similarly, D1 also requires less iterations in this case for the same reason. However, the number of iterations has to exceed a minimum in order to overcome the averaging-induced latency associated with this detector.

In the context of our investigations presented so far the phase rotation problem has been neglected, since the phase estimation has been assumed to be perfect. In a more realistic scenario, a few pilot symbols can be used for estimating the phase, without creating a significant overhead. For example, if we use three pilots out of 174 symbols of a transmission frame, then the overhead is around 1.7%. This corresponds to an SNR-shift of about 0.07dB, if we take it into account in the BER versus Bit SNR curves, which is almost negligible. In Figure 10 we have plotted the BER curves for both the static and fading channels of Figure 3 using both perfect phase estimation (PPE) and in conjunction with estimation using three pilot symbols. The performance is clearly very similar for both the static and fading channels and even for low SNR values and for twin-pilot assisted estimation. This justifies the employment of the phase estimation approach.

6. CONCLUSIONS AND FUTURE WORK

A new blind equalizer was proposed, performing jointly channel equalization and channel decoding by combining turbo equalization with PSP. The benefit of feeding soft channel decoding information back to the equalizer's input in a closed loop renders this equalizer robust against noise, unlike common blind equalizers. Future work in this field includes the application of this equalizer over fast fading channels and the study of its convergence regions.

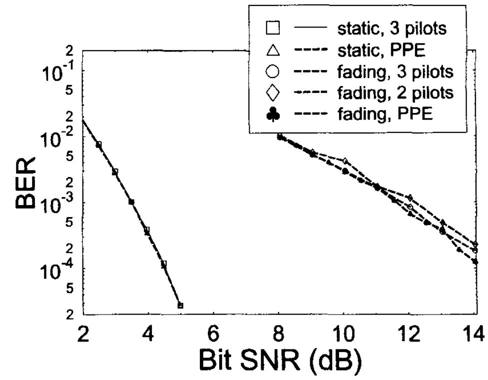

Figure 10: BER comparison between perfect phase estimation (PPE) and pilots phase estimation. The channel is assumed to be static or fading, the impulse response of which is shown in Figure 3(a) and the turbo-PSP equalizer parameters are given in Table 1.

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. Bauch, H. Khorram, and J. Hagenauer, "Iterative equalization and decoding in mobile communications systems," in *Proceedings of The Second European Personal Mobile Communications Conference (EPMCC'97)*, (Bonn, Germany), pp. –, September 1997.