Beyond BAD:A Parallel Arbitration Framework for Low-Complexity Equalization*

Gwen Barriac
Dept. of Electrical Engineering
University of California at Santa Barbara
Santa Barbara, CA 93106
barriac@engineering.ucsb.edu

Noah Jacobsen
Dept. of Electrical Engineering
University of California at Santa Barbara
Santa Barbara, CA 93106
jacobsen@engineering.ucsb.edu

Upamanyu Madhow
Dept. of Electrical Engineering
University of California at Santa Barbara
Santa Barbara, CA 93106
madhow@pablo.ece.ucsb.edu

Abstract

Since optimal equalization has complexity exponential in channel length and information rate, the suboptimal Decision Feedback Equalizer (DFE), whose complexity is linear in channel length, is a popular alternative for practical communication systems. The recently proposed BAD (Bidirectional Arbitrated DFE) algorithm provides substantial performance gains over the standard DFE by arbitrating between the outputs of a forward and reverse DFE which are run in parallel. The idea behind BAD is to generate two "sufficiently different"candidate data sequences, and to arbitrate between them based on which sequence best explains the received data around the symbol being demodulated. In this paper, we demonstrate that a natural generalization of this methodology- arbitrating between multiple candidate data sequences generated in parallel with low-complexity equalizers- can further improve performance while still incurring complexity which is comparable to that of the DFE. We go beyond the BAD algorithm in two key respects, by providing methods for generating additional candidate data sequences that have low correlations among their error patterns, and by using improved arbitration mechanisms.

1 Introduction

Optimal equalization, in the sense of either maximum likelihood (ML) or maximum a posteriori probability (MAP), has complexity growing as M^L, where M is the alphabet size and L the channel length, which makes it prohibitively complex for systems with large alphabet size or even moderate length channels. Many suboptimal equalization techniques have been proposed in the literature as less complex alternatives to ML or

*This research is supported by the Army Research Office under grant DAAD19-00-1-0567, and by the National Science Foundation under a research infrastructure grant CDA 96-24396.

MAP, the most popular being the classical decision feedback equalizer (DFE)[5]. The DFE has complexity which scales linearly with the channel length L and which is independent of the alphabet size. Though far less complex, DFE performance for typical channels is about 2-3 dB away from ML performance[4]. BAD (bidirectional arbitrated DFE), an algorithm which builds upon the conventional DFE, reduces this gap to about 1 dB while keeping the complexity on the same order as that of the DFE[6]. In this paper, we extend the concepts of BAD to show that this gap can be further reduced with suboptimal techniques, without incurring the exponential complexity inherent in optimal decoding.

BAD works by running a forward and reverse DFE in parallel over the entire block of data, and then choosing which decisions best match the received data. The algorithm has three stages[6]:

Bi-directional processing: Two sets of symbol decisions are obtained, using a forward and a reverse DFE.

Reconstruction: Two different estimates of the noise-free received data are produced by passing the symbol estimates obtained in stage one through the channel.

Arbitration: For each symbol being decoded, the symbol decision is chosen from either the forward or the reverse DFE, depending on which set of estimates provides a reconstructed received signal closest to the actual received signal in a local interval around the symbol of interest.

BAD obtains two candidate sequences using suboptimal techniques (forward and reverse DFE's) and arbitrates between the two for every symbol. The low correlation between the errors produced by the forward and reverse DFE's allows the arbitration stage to produce a symbol sequence much closer to the correct sequence than either of the original candidate sequences. While ML equalization chooses from all possible data sequences the one which best fits the received sequence, BAD reduces the number of candidate sequences to only two, and uses the locality of the arbitration mechanism to exploit the low correlation of errors between these sequences. The approach in this paper is to pursue a natural extension of BAD by introducing more candidate sequences, keeping variants of the DFE as the basic building block, whose error patterns have low correlation. We also propose a new arbitration technique tailored to the error propagation typical of DFEs [4]. We find that we are indeed able to improve the performance of the BAD algorithm using these techniques. However, the improvement is only a fraction of a dB, while the increase in complexity over BAD is significant. As of now, therefore, BAD still remains the best method for suboptimal equalization with complexity comparable to the DFE.

The suboptimal equalization techniques considered here are targeted to applications in which optimal equalization is too complex to be a viable alternative. However, in order to compare our algorithms with the benchmarks provided by optimal equalization, the simulation results reported are for models in which MAP equalization using the BCJR algorithm [1] can be run in a reasonable time period. Thus, the simulation results presented are for BPSK transmission for a moderate channel length. As in the paper reporting the BAD algorithm [6], we focus on uncoded (post-equalization) error probabilities in the range of 10^-1 to 10^-3 (the range required by error correction codes in typical use), and use two "toy" channel models to validate our ideas: (a) a hard to equalize symmetric channel from the standard text by Proakis (b) a maximum phase channel obtained by rearranging the coefficients of the symmetric channel.

Section 2 describes relevant background on equalization, and Section 3 explains the details of the proposed algorithms. Numerical results can be found in Section 4, and Section 5 concludes the work, recapitualting important concepts and stating possible directions for future research.

2 Background

Consider the channel model shown in figure 1.

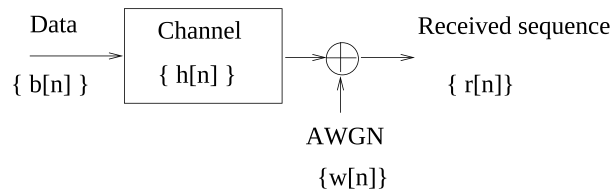

Figure 1: The equivalent discrete-time channel model

The information frame being transmitted is a sequence of uncoded, binary phase-shift keyed(BPSK) data denoted by $\mathbf{b} = \{b[n]\}_{n=1}^{N}$. The data is linearly modulated over a real baseband, discrete-time, symbol-spaced channel,and corrupted by additive Gaussian noise (AWGN). The output of of the channel at time n is given by

$$r[n] = \sum_{k=-L_1}^{L_2} h[k]b[n-k] + w[n], \tag{1}$$

where $\mathbf{h} = \{h[k], -L_1 \le k \le L_2\}$ is the channel impulse response, and $w[n]$ is additive white Gaussian noise (AWGN). The channel is assumed to be invariant over the enitre length of the frame, and it is also assumed that the receiver has full knowledge of both the channel impulse response $\mathbf{h}$ and the noise variance σ_w^2.

Though the preceding model was used for simplicity, the algorithms discussed in this paper also apply to complex baseband, fractionally spaced channels with two-dimensional symbol alphabets and colored noise.

2.1 The Classical DFE

When trying to decode bit $b[n]$, consider the the received vector $\mathbf{r}[n]$ in the nth observation interval. $\mathbf{r}[n]$, or $(r[n-k_1], \ldots, r[n+k_2])^T$, is defined by

$$\mathbf{r}[n] = b[n]\mathbf{u}_0 + \sum_{j \ne 0} b[n+j]\mathbf{u}_j + \mathbf{w}[n]$$

where $\mathbf{u}_0$ is the desired vector, which is the response to the desired symbol $b[n]$ truncated to the observation interval, $\{\mathbf{u}_j, j \ne 0\}$ are interference vectors modulating the ISI symbols $\{b[n+j], j \ne 0\}$, and $\mathbf{w}[n]$ is AWGN with variance σ_w^2 per dimension.

The classical DFE assumes that the past symbols have been decoded correctly, and therefore employs a feedforward correlator to linearly supress the ISI from future symbols $\{b[n+j], j > 0\}$ only. Interference from past symbols is not considered because the contribution from these symbols is to be subtracted out from the received vector using past symbol estimates. In this paper, we consider the MMSE-DFE, where the coefficients of the linear feedforward filter are chosen by minimizing the following equation:

$$E[|\langle \mathbf{c}_{ff}, \mathbf{r}_{future}[n] \rangle - b[n]|^2]$$

where

$$\mathbf{r}_{future}[n] = b[n]\mathbf{u}_0 + \sum_{j>0} b[n+j]\mathbf{u}_j + \mathbf{w}[n]$$

is a hypothetical received vector obtained if the contribution of the past symbols were perfectly cancelled out. We therefore obtain the standard LMMSE solution

$$\mathbf{c}_{ff} = \mathbf{R}_{future}^{-1}\mathbf{p}$$

where $\mathbf{R}_{future} = E\{\mathbf{r}_{future}[n](\mathbf{r}_{future}[n])^T\}$ and $\mathbf{p} = E\{b[n]\mathbf{r}_{future}[n]\}$. For uncorrelated symbols with $E\{|b[k]|^2\} = \sigma_b^2$, we obtain

$$\mathbf{R}_{future} = \sigma_b^2 \mathbf{U}_{future}\mathbf{U}_{future}^T + \sigma_w^2 \mathbf{I}$$

and

$$\mathbf{p} = \sigma_b^2 \mathbf{u}_0$$

where $\mathbf{U}_{future}$ denotes a matrix with columns containing the signal vectors $\{\mathbf{u}_j, j \geq 0\}$ modulating the current and future symbols for a given observation interval, and $\mathbf{I}$ denotes the identity matrix. For the cases considered in this paper,$\sigma_b^2 = 1$.

Because the decision statistic for symbol $b[n]$ originates from the inner product of the feedforward filter and the received vector $\mathbf{r}[n]$,which includes both past and future symbols, it is necessary to subtract out the contribution of the past symbols. Let the desicion statistic for bit $b[n]$ be defined by:

$$Z[n] = \langle \mathbf{c}_{ff}, \mathbf{r}[n]\rangle - \langle \mathbf{c}_{fb}, \hat{\mathbf{b}}[n]\rangle$$

where $\mathbf{c}_{ff}$ is the feedforward correlator of length $L_{ff} = k_1 + k_2 + 1$, $\hat{\mathbf{b}}[n] = (\hat{b}[n - L_{fb}], ..., \hat{b}[n-1])^T$ is the vector of estimates of the past symbols, and $\mathbf{c}_{fb}$ is the vector of feedback coefficients.

The response at the output of the feedforward filter to the past symbols is given by $\sum_{j<0} b[n+j]\langle \mathbf{c}_{ff}, \mathbf{u}_j\rangle$,and thus the feedback filter coefficients need be given by

$$\mathbf{c}_{fb}(j) = \langle \mathbf{c}_{ff}, \mathbf{u}_{-j}\rangle, 1 \leq j \leq L_{fb}.$$

In our numerical results, we assume that the number L_{fb} of feedback coefficients equals the number of past symbols falling in the observation interval. In practice, a smaller number of feedback taps may be chosen if the channel length is large.

The estimate for the $b[n]$ is obtained from $Z[n]$ based on the standard decision regions, e.g., for binary antipodal signaling, we have that the estimate is given by $\hat{b}[n] = \text{sign}(Z[n])$.

2.2 The BAD algorithm

The BAD algorithm consists of three stages: (1) bidirectional processing with MMSE-DFE's, (2) data reconstruction, and (3) symbol arbitration.

In the first stage, two MMSE-DFE's are run in parallel as shown in Figure 2. The first DFE produces an estimate $\hat{b}_{fwd}[n]$ for the sequence $b[n]$. The second DFE takes a time reversal of the received sequence, $\check{r}[n]$, and runs it through a time reversed DFE- a DFE based on the time reversed channel impulse response $\check{h}[n]$. The output is then time reversed again to produce another estimate $\hat{b}_{rvs}[n]$ of the original input sequence $b[n]$.

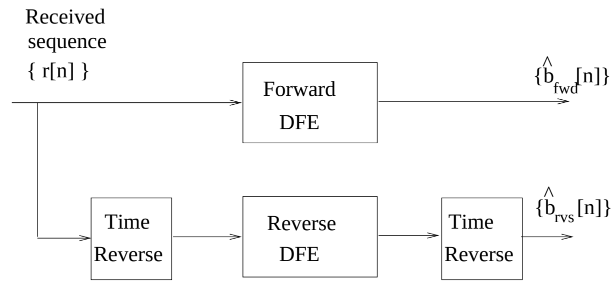

Figure 2: Bidirectional processing is used to get two different estimates of the original symbol sequence, denoted by $\{\hat{b}_{fwd}[n]\}$ and $\{\hat{b}_{rvs}[n]\}$, respectively.

In the reconstruction stage, both $b_{fwd}[n]$ and $b_{rvs}[n]$ are passed though the channel filter to produce noise free estimates of the received data $r[n]$. These noise-free versions of the received data are defined as follows:

$$\hat{r}_{fwd}[n] = \sum_{k=-L_1}^{L_2} h[k]\hat{b}_{fwd}[n-k], \tag{2}$$

$$\hat{r}_{rvs}[n] = \sum_{k=-L_1}^{L_2} h[k]\hat{b}_{rvs}[n-k]. \tag{3}$$

Finally, the BAD algorithm arbitrates between $\hat{b}_{fwd}[n]$ and $\hat{b}_{rvs}[n]$ to produce $\hat{b}_{BAD}[n]$. For each symbol being decoded, it is determined whether $\hat{r}_{fwd}[n]$ or $\hat{r}_{rev}[n]$ gives the closest fit to $r[n]$ in the local area of interest. To be more precise, the following metrics are computed:

$$\lambda_{fwd}[n] = \sum_{k=-W_1}^{k=W_2} |r[n+k] - \hat{r}_{fwd}[n+k]|^2 \tag{4}$$

$$\lambda_{rvs}[n] = \sum_{k=-W_1}^{k=W_2} |r[n+k] - \hat{r}_{rvs}[n+k]|^2$$

As can be seen from the above equations, decsions are made from looking at a window of length $W = W_1 + W_2 + 1$ around the desired symbol. The final decison is then made according to which metric is smaller.

$$\hat{b}_{BAD}[n] = \begin{cases} \hat{b}_{fwd}[n] & \lambda_{fwd}[n] < \lambda_{rvs}[n] \\ \hat{b}_{rvs}[n] & \lambda_{fwd}[n] > \lambda_{rvs}[n] \end{cases}$$

3 New Algorithms

The new algorithms focus on generating more candidate data sequences at complexity comparable with that of the DFE. The final reconstruction and arbitration stage are as in BAD, except that some of the candidates result from applying a different arbitration mechanism at an intermediate stage.

3.1 Partial Supression DFE (PSDFE)

Our first idea was to extend BAD by producing four, rather than two, candidate bit sequences before proceeding to the arbitration stage. The additional candidates come from running a forward and reverse Partial Suppression DFE (PSDFE), in which the effect of the past symbols at the output of the feedforward filter is attenuated, in order to reduce the effect of error propagation. Thus, the feedforward filter in a PSDFE is designed to not only supress future interference, but to also partially supress past interference. The feed forward filter is now given by

$$\mathbf{c}_{ff} = \tilde{\mathbf{R}}^{-1}\mathbf{p}$$

However, now the following formula applies for $\tilde{\mathbf{R}}$

$$\tilde{\mathbf{R}} = \mathbf{U}_{future}\mathbf{U}_{future}^T + \alpha * \mathbf{U}_{past}\mathbf{U}_{past}^T + \sigma_w^2\mathbf{I}$$

where $\mathbf{U}_{past}$ contains the signal vectors $\{\mathbf{u}_j, j < 0\}$, and $\alpha \in [0, 1)$ is the effective interference energy of the past symbols (due to errors in the feedback) relative to those of the future symbols.

Arbitration between these four sequences for each symbol being demodulated produces the final output. We found that PSDFE's did not produce sequences different enough from those obtained using the classical DFE's to provide any improvement in overall performance, so other methods of producing candidate sequences were considered.

3.2 Partial Data DFE

The Partial Data DFE (PDDFE) is a variant of the classical DFE which was developed to produce candidate sequences different enough from those produced by the classical DFE so as to improve results after arbitration. The algorithm works by only looking at a subset of the received vector. In this way, if a particular sample is strongly corrupted by noise, avoiding it will help stem error propagation. The intuition is that, although the PDDFE produces candidate sequences which are generally much worse than classical DFE sequences because of the loss of information due to using only part of the received data, the resulting sequences are disparate enough to produce overall gain. The forward PDDFE outputs two candidate data sequences. The algorithm works as follows.

The received vector $\mathbf{r}[n]$, used to decode bit $b[n]$ is defined in the classical DFE by $\mathbf{r}[n] = (r[n-k_1], \ldots, r[n+k_2])^T$. For the PDDFE,

$$\begin{aligned} \mathbf{r}[n] &= (r[n-k_1], \ldots, r[n-3], r[n-1], r[n], r[n+1], r[n+3], \ldots, r[n+k_2])^T \quad n\ even \\ \mathbf{r}[n] &= (r[n-k_1], \ldots, r[n-4], r[n-2], r[n], r[n+2], r[n+4], \ldots, r[n+k_2])^T \quad n\ odd \end{aligned} \tag{5}$$

A second candidate sequence is produced by using the first equation for n odd, and the second for n even. After the vectors are thus defined, the feedforward and feedback coefficients for the PDDFE are computed exactly as in a DFE.

The PDDFE can also be run in reverse to produce an additional two candidate sequences. Thus, we have four additional candidate data sequences due to use of the PDDFE.

3.3 Lopsided Arbitration

Lopsided arbitration uses the Euclidean distance as a metric to arbitrate, as in (4), so that the arbitration for the ith candidate data sequence is still of the following form:

$$\lambda_i[n] = \sum_{k=-W_1}^{k=W_2} |r[n+k] - \hat{r}_{candidate i}[n+k]|^2$$

In "regular" arbitration, the bit being demodulated falls somewhere in the center of the window of received samples being considered, so that W_1 and W_2 are as large as possible (or as necessary) in "regular" arbitration. With lopsided arbitration, two new candiate sequences are created, first by arbitrating among all candidates with W_1 set to 0 and then by doing the same with W_2 set to zero. These two new sequences are then put back into the pool of candidate sequences and "regular" arbitration is used for the final decisions. The idea behind this is that DFE's tend to produce errors in clumps, so that regular arbitration may choose an incorrect bit in favor of the correct one if an error clump to either side of the bit in question sufficiently skews the data. Lopsided arbitration attempts to remedy this problem, although it cannot correct incorrect bits within an error clump if all candidate sequences are corrupted there. Lopsided arbitration is found to work the best in conjunction with PDDFEs.

4 BetterBAD

The BetterBAD algorithm combines all of the above ideas, and was found to work better than any of these ideas applied on its own. Eight candidate sequences are obtained in parallel from forward and reverse DFE's, PSDFE's, and PDDFE's. Rightsided and leftsided arbitration are used on these eight sequences to produce an additional two, and finally regular arbitration is used to decide between these two sequences and the sequence produced by the BAD algorithm.

5 Performance Results

The algorithms just discussed were run for two channel models, and the results compared to standard suboptimal equalization techniques as well as MAP. MAP detection was performed using the BCJR algorithm. Because of the computational difficulty of running BCJR for large symbol constellations, the input data was chosen to be packets of 500 randomly generated BPSK symbols. Of course, this does not imply that any of the algorithms need be limited to BPSK modulation, for the implementation of these algorithms can be easily extended to larger alphabet sizes. 2000 packets were run for each value of E_b/N_0 considered, and then the resulting bit error rate (BER) was plotted vs. E_b/N_0. The values of E_b/N_0 were chosen such that the BER ranged from around 10^-1 to 10^-3, the range required by most decoders in use today. For all simulations, the feedforward filter has 15 taps, the feedback filter 9, and the length W of the arbitration window is $3L$, where L is the channel length. Also the suppresssion parameter α in the Partial Supression DFE's is set to .1.

5.1 Stationary Channels

Two stationary channels were considered, a symmetric channel $H_{sym}(z)$, and a maximum phase channel $H_{max}(z)$. Their system functions are as follows:

$$H_{sym}(z) = 0.227 + 0.46z^{-1} + 0.688z^{-2} + 0.46z^{-3} + 0.227z^{-4}, \quad \text{and} \tag{6}$$

$$H_{max}(z) = 0.227 + 0.227z^{-1} + 0.46z^{-2} + 0.46z^{-3} + 0.688z^{-4}. \tag{7}$$

The symmetric channel is chosen because it is known to be difficult to equalize [5].

Figure 3 shows the results of various algorithms for the symmetric channel. Plotted are the results for the classical forward DFE, BAD, BetterBAD, and MAP. BAD is about 1 dB away from MAP, and BetterBAD closes this gap to about .6 dB. Localized MAP is about .25 dB away from MAP[7], which means that BetterBAD is only .35 dB away from localized MAP. Still, it seems clear that for all practical purposes, BAD gives the most improvement over the standard DFE for the least amount of added complexity.

Figure 4 shows the results for the maximum phase channel. In this case, the gap between BAD and MAP equalization was small to start with, about .4 dB. BetterBAD closes this gap to about .2 dB.

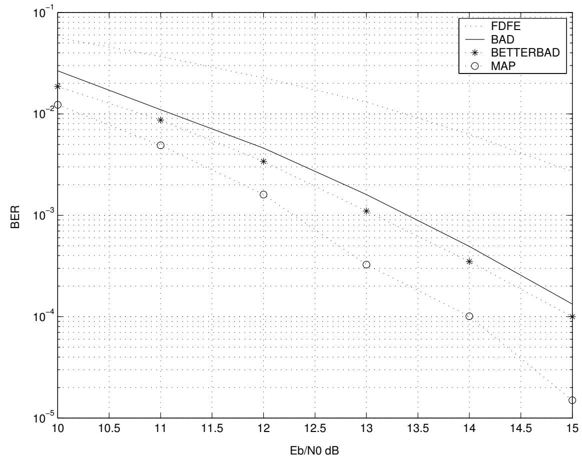

Figure 3: Bit error probability versus $Eb/N0$ for various equalization algorithms for symmetric channel $h = (.227, .46, .688, .46, .227)$.

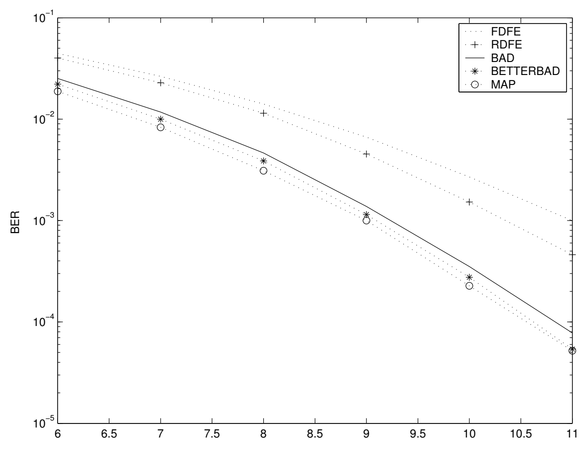

Figure 4: Bit error probability versus $Eb/N0$ for various equalization algorithms for maximum phase channel $h = (.227, .227, .46, .46, .688)$.

5.2 Complexity Considerations

It is useful to have an idea of how the variations of the BAD algortihm increase complexity. All of the following complexity calculations given are made on a per symbol basis. The DFE requires $K_1 + K_2 + 1$ multiplications, where K_1 is the number of feedforward taps and K_2 is the number of feedback taps. Reconstruction requires $\frac{N+L-1}{N}$ multiplications where L is the channel length and N is the length of data, and finally the arbitration phase gives an additional W multiplications where W is the arbitration window size[6]. The following table gives the complexity for various equalizers.

Table 1: Computational Complexity of Equalization Methods

Equalizer	calculations	operation
MMSE-DFE	$K_1 + K_2 + 1$	$\times$'s
BAD algorithm	$2(K_1 + K_2 + 1 + (\frac{N+L-1}{N})L + W)$	$\times$'s
BetterBAD algorithm	$8(K_1 + K_2 + 1) + 10(\frac{N+L-1}{N})L + 20W$	$\times$'s
MAP (BCJR)	M^{2L}	$\times$'s

BetterBAD and BAD have complexities which are linearly dependent on the length of the channel and the length of the equalizer, and even for small channel lengths and BPSK modulation, the complexity is much less than that of MAP. The large gap in complexity between the BAD based algorithms and MAP only increases for larger channel lenghts and larger alphabet sizes (M). However, comparing the difference in complexity between BAD and BetterBAD, it seems that BetterBAD's improved performace does not merit the increased complexity given the constraints of practical systems.

6 Conclusion

In terms of practical applications, it seems that the BAD algorithm is still the most viable upgrade of the classical DFE, in terms of producing the greatest gain for the least amount of added complexity. Our intuition, based on the extensive experiments we have done, is that BAD and BetterBAD essentially solves the error propagation problems of the DFE (performing better at high SNR than the DFE with perfect decisions fed back), but that we might be encountering a fundamental bottleneck because DFE-like schemes do not exploit the correlations in the noise samples that affect adjacent decisions. It remains to be seen whether such intuitions can be translated into suboptimal equalization schemes that narrow the gap to MAP equalization at reasonable complexity.

It is also of great interest to see whether suboptimal equalization schemes can be used in conjuction with decoding techniques to approach fundamental limits without incurring the complexity of optimal algorithms. For example, a low complexity joint decoder/eqaulizer could function as a possible alternative to the turbo equalization scheme of [2] and some of its approximations such as [9].

References

[1] L. R. Bahl, J. Cocke, F. Jelinek, J. Raviv, J. "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Transactions on Information Theory*, IT-20, pp. 284-287, March 1974.

[2] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, A. Glavieux, "Iterative correction of intersymbol interference: turbo equalization," *Eur. Trans. Telecommun.,* vol. 6, pp. 507-511, Sept.-Oct. 1995.

[3] A. Duel-Hallen, C. Heegard, “Delayed decision-feedback sequence estimation,” *IEEE Trans. Commun.,* vol. 37, no. 5, pp. 428-436, May 1989.

[4] E. A. Lee, D. G. Messerschmitt, *Digital Communication*, Kluwer Academic Publishers, Horwell, MA, pt. I, p. 453; pt. II, p. 460; pt. III, p. 476; pt. IV, p. 476, 1994.

[5] Proakis, J.G. *Digital Communications*, McGraw-Hill, New York, pp. 616-617, 1995.

[6] C. S. McGahey, A. C. Singer, U. Madhow, “BAD: a bi-directional arbitrated decision feedback equalizer,” *Proceedings of the 2000 Conference on Information Sciences and Systems (CISS2000)*, Princeton, NJ, March 15-17, 2000.

[7] A. C. Singer, U. Madhow, C. S. McGahey, J. K. Nelson, “BAD: Bidirectional Arbitrated Decision Feedback Equilization”, submitted for publication.

[8] Stuber, G.L. *Mobile Communication*,Kluwer Academic Publishers,Boston,1996.

[9] M. Tuchler, R. Koetter, A. Singer, “Iterative correction of ISI via equalization and decoding with priors,” *Proc. 2000 International Symposium on Information Theory (ISIT 2000)*, pp. 100.