Burst Mode Non-Causal Decision-Feedback Equalization and Blind MLSE†

Dirk T.M. Slock and Elisabeth De Carvalho
Institut EURECOM, 2229 route des Crêtes, B.P. 193,
06904 Sophia Antipolis Cedex, FRANCE
Tel: +33 93002606 Fax: +33 93002627
email:{slock,carvalho}@eurecom.fr

ABSTRACT

We propose a decision-aided equalizer that uses past symbol decisions, like classical decision-feedback equalizers, but also future decisions, hence the name Non-Causal Decision-Feedback Equalizer (NCDFE). Its derivation is done in a burst processing and multichannel context, using Maximum Likelihood Sequence Estimation (MLSE). The forward filter of the NCDFE is proportional to the channel matched filter and the feedback filter is the cascade of the channel and the forward filter (without the central coefficient). This NCDFE corresponds to the Unbiased MMSE solution. Blind MLSE is usually solved using alternating minimizations. We propose to replace herein the repeated use of the Viterbi algorithm with the much less complex NCDFE. Finally, some simulation results are presented

1 Introduction

In wireless communications, multipath propagation introduces Intersymbol Interference (ISI). Techniques to combat ISI include decision-aided ISI cancellation, like decision-feedback equalizers, where the ISI corresponding to past symbols is synthesized and subtracted from the slicer input. It is however possible to extend this concept to future decisions. If past and future symbols as well as a perfect model of the ISI process are known, all ISI can be eliminated from the signal, and the matched filter bound can be achieved. This was first proposed by Proakis [1]: future decisions are tentative decisions given by another equalizer. Gersho and Lim [2] have determined the structure of such an equalizer, using the MMSE criterion: the forward filter of their equalizer is proportional to the channel matched filter and the feedback filter is the cascade of the channel and the forward filter (without the central coefficient). However, the proportionality factor for the MMSE solution differs from the one in the Unbiased MMSE solution.

We propose the derivation of an ISI canceler using past and future symbols, starting from the Maximum Likelihood (ML) criterion, in a burst processing mode, i.e. symbol detection is done burst by burst, using only the symbols present in the burst.

The work of Elisabeth de Carvalho was supported by Laboratoires d'Electronique PHILIPS under contract Cifre 297/95. The work of Dirk Slock was supported in part by Laboratoires d'Electronique PHILIPS under contract LEP 95FAR008.

2 The Multichannel Model

We consider here a FIR Multichannel model. The multiple FIR channels are due to oversampling of a single received signal and/or the availability of multiple received signals from an array of antennas (in the context of mobile digital communications). To further develop the case of oversampling, consider linear digital modulation over a linear channel with additive noise so that the cyclostationary received signal can be written as

$$\mathrm{y}(t) \;=\; \sum_k \mathrm{h}(t-kT)a(k) \;+\; \mathrm{v}(t) \qquad (1)$$

where the $a(k)$ are the transmitted symbols, T is the symbol period and $\mathrm{h}(t)$ is the channel impulse response. The channel is assumed to be FIR with duration NT (approximately). If the received signal is oversampled at the rate $\frac{m}{T}$ (or if m different received signals are captured by m sensors every T seconds, or a combination of both), the discrete input-output relationship can be written as:

$$\boldsymbol{y}(k) \;=\; \sum_{i=0}^{N-1} \boldsymbol{h}(i)a(k-i) + \boldsymbol{v}(k) \;=\; \boldsymbol{H}A_N(k) + \boldsymbol{v}(k)\,,$$
$$\boldsymbol{y}(k) = \begin{bmatrix} y_1(k) \\ \vdots \\ y_m(k) \end{bmatrix}, \boldsymbol{v}(k) = \begin{bmatrix} v_1(k) \\ \vdots \\ v_m(k) \end{bmatrix}, \boldsymbol{h}(k) = \begin{bmatrix} h_1(k) \\ \vdots \\ h_m(k) \end{bmatrix}$$
$$\boldsymbol{H} = [\boldsymbol{h}(N{-}1)\cdots\boldsymbol{h}(0)]\,, A_N(k) = \left[a^H(k{-}N{+}1)\cdots a^H(k)\right]^H \qquad (2)$$

where the subscript i denotes the i^{th} channel and superscript ^{H} denotes Hermitian transpose. In the case of oversampling, $y_i(k)\,, i=1,\ldots,m$ represents the m phases of the polyphase representation of the oversampled signal: $y_i(k) = \mathrm{y}(t_0 + (k+\frac{i}{m})T)$. In the polyphase representation of the oversampled signals, we get a discrete-time circuit in which the sampling rate is the symbol rate. Its output is a vector signal corresponding to a SIMO (Single Input Multiple Output) or vector channel consisting of m SISO discrete-time channels where m is the sum of the oversampling factors used for the possibly multiple antenna signals. Let channel transfer function. Consider additive independent white Gaussian noise $\boldsymbol{v}(k)$ with $r_{\boldsymbol{vv}}(k-i) = \mathrm{E}\,\boldsymbol{v}(k)\boldsymbol{v}^H(i) = \sigma_v^2 I_m\,\delta_{ki}$. Assume we receive M samples:

$$\boldsymbol{Y}_M(k) \;=\; \mathcal{T}_M(\boldsymbol{H})\,A_{M+N-1}(k) + \boldsymbol{V}_M(k) \qquad (3)$$

where $\boldsymbol{Y}_M(k) = [\boldsymbol{y}^H(k-M+1)\cdots\boldsymbol{y}^H(k)]^H$ and similarly for $\boldsymbol{V}_M(k)$, and $\mathcal{T}_M(\boldsymbol{H})$ is a block Toepliz matrix with M block rows and $[\boldsymbol{H}\;\; 0_{m\times(M-1)}]$ as first block row.

3 Non-Causal DFE

3.1 Burst Mode Detection

In burst mode processing, which is particularly well adapted to mobile communications where the data is organized in bursts, only the symbols present in the burst are processed. On the other hand, in continuous processing, all the data (from $t=-\infty$ to $t=+\infty$) is available.

In burst mode, symbol decisions are then made on a burst by burst basis. Furthermore, the channel is supposed to be constant during the transmission of a burst.

As the processing is the same for each burst, we adopt simplified notations in the whole paper. The burst we are trying to detect, A, is of length $M+N-1$, and composed of elements $\left[a^H(-N+1)\cdots a^H(M-1)\right]^H$. The corresponding observations are $\boldsymbol{Y} = \left[\boldsymbol{y}^H(0)\cdots\boldsymbol{y}^H(M-1)\right]^H$, and the noise is denoted $\boldsymbol{V}$. The relationship (3) is then written:

$$\boldsymbol{Y} = \mathcal{T}A + \boldsymbol{V} \qquad (4)$$

3.2 Derivation

We suppose here that the channel is known and we propose to determine the input symbols from the received signal $\boldsymbol{Y}$, by solving the Maximum Likelihood criterion:

$$\min_{a(k)\in \mathrm{A}} \|\boldsymbol{Y} - \mathcal{T}A\|^2 \qquad (5)$$

where A is the discrete constellation alphabet. We shall detect the symbols recursively. Hence consider the detection of the symbol $a(k)$. For that purpose, we decompose the quantity $\mathcal{T}A$ as follows:

$$\mathcal{T}A = \mathcal{T}_{\bar{A}}(k)\bar{A}(k) + \mathcal{T}_a(k)a(k) \qquad (6)$$

$\mathcal{T}_a(k)$ is the column of $\mathcal{T}$ that gets multiplied by $a(k)$, $\mathcal{T}_{\bar{A}}(k)$ is the matrix $\mathcal{T}$ from which the column $\mathcal{T}_a(k)$ has been eliminated, $\bar{A}(k)$ contains past and future symbols w.r.t. $a(k)$. $\mathcal{T}_a(k)a(k)$ contains the contribution in $\mathcal{T}A$ from the symbol $a(k)$ to be detected, and $\mathcal{T}_{\bar{A}}(k)\bar{A}(k)$ the contribution from the other symbols. Fig. 1 shows more explicitly the structure of this decomposition. The blocks in the matrix $\mathcal{T}$ represent the channel $\boldsymbol{H}$, here of length N=3, composed of the coefficients $\boldsymbol{h}(0)$, $\boldsymbol{h}(1)$ and $\boldsymbol{h}(2)$. Using this decomposition, criterion (5) can be rewritten as:

$$\min_{a(k)\in \mathrm{A}} \|\left[\boldsymbol{Y} - \mathcal{T}_{\bar{A}}(k)\bar{A}(k)\right] - \mathcal{T}_a(k)a(k)\|^2 \qquad (7)$$

Its solution gives:

$$\begin{cases} \hat{a}(k) = \mathrm{dec}\left\{z(k)\right\} \\ z(k) = (\mathcal{T}_a^H(k)\mathcal{T}_a(k))^{-1}\mathcal{T}_a^H(k)\left[\boldsymbol{Y} - \mathcal{T}_{\bar{A}}(k)\bar{A}(k)\right] \end{cases} \qquad (8)$$

where dec{.} is the decision operation that chooses the element in the alphabet A closest to its argument.

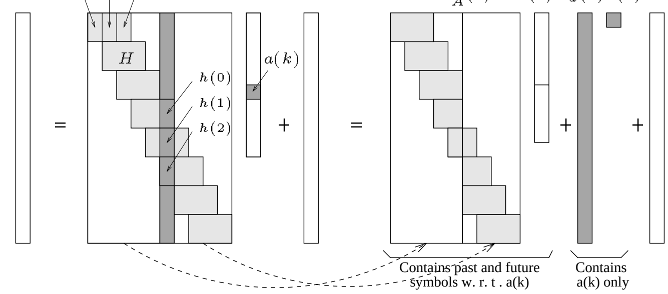

Figure 1: Decomposition of $\mathcal{T}A$ in the case of a channel of length N=3

3.3 Decision Feedback Structure

This solution may be thought of in terms of decision feedback. We consider here that for the detection of $a(k)$, we dispose of past and future decisions w.r.t. $a(k)$, grouped in the vector $\widehat{\bar{A}}(k)$. If we suppose that those decisions are correct so that $\widehat{\bar{A}}(k) = \bar{A}(k)$, then, using (8):

$$\begin{aligned} z(k) = \;& \left[(\mathcal{T}_a^H(k)\mathcal{T}_a(k))^{-1}\mathcal{T}_a^H(k)\right]\boldsymbol{Y} \\ & - \left[(\mathcal{T}_a^H(k)\mathcal{T}_a(k))^{-1}\mathcal{T}_a^H(k)\mathcal{T}_{\bar{A}}(k)\right]\widehat{\bar{A}}(k) \end{aligned} \qquad (9)$$

We can now recognize a decision feedback structure.

The forward filter, applied to $\boldsymbol{Y}$, is $(\mathcal{T}_a^H(k)\mathcal{T}_a(k))^{-1}\mathcal{T}_a^H(k)$, the feedback filter, applied to the past and future decisions, is $(\mathcal{T}_a^H(k)\mathcal{T}_a(k))^{-1}\mathcal{T}_a^H(k)\mathcal{T}_{\bar{A}}(k)$.

The feedback filter synthesizes the cascade of the channel and the forward filter, except for the central coefficient corresponding to the symbol to be detected $a(k)$. It then eliminates the contribution of past and future symbols present in the signal at the output of the forward filter. If past and future decisions are correct, all the ISI can be eliminated and the only symbol present at the input of the decision device is then $a(k)$. In that case, this equalizer would attain the matched filter bound.

Unlike classical decision-feedback equalizers, past, but also future symbol decisions are used: the feedback filter is not causal anymore but becomes anticausal, this is why we call it **Non-Causal Decision-Feedback Equalizer** (NCDFE).

3.4 Forward and Feedback Filters

Let's take a closer look at the forward and feedback filters. $\mathcal{T}_a^H(k)$ contains the coefficients of $\boldsymbol{H}$, surrounded by zeros: multiplying $\boldsymbol{Y}$ by $\mathcal{T}_a^H(k)$ gives one output sample of the filtering of $\boldsymbol{Y}$by the channel matched filter. The product $\mathcal{T}_a^H(k)\mathcal{T}_a(k)$ being simply a scalar equal to $\|\mathbf{H}\|^2$, the forward operation is:

$$(\mathcal{T}_a^H(k)\mathcal{T}_a(k))^{-1}\mathcal{T}_a^H(k)\boldsymbol{Y} = \frac{\boldsymbol{H}^{t*}}{\|\mathbf{H}\|^2}\boldsymbol{Y}_N(k+N-1) \qquad (10)$$

where $\boldsymbol{H}^t$ is the block-wise transpose of $\boldsymbol{H}$, $\boldsymbol{H}^t = \left[\boldsymbol{h}(N-1)^T\cdots\boldsymbol{h}(0)^T\right]$.

Because the channel is FIR, only N observation samples are needed to detect one symbol. Furthermore, as the signal at the output of the matched filter $\mathcal{T}_a^H(k)$ contains $N-1$

and $N-1$ past decisions are needed.

The feedback filter, as already mentioned, is the cascade of the channel and the matched filter, except for the central element of the cascade. The feedback operation is:

$$(\mathcal{T}_a^H(k)\mathcal{T}_a(k))^{-1}\mathcal{T}_a^H(k)\mathcal{T}_{\bar{A}}(k)\hat{\bar{A}}=\frac{\boldsymbol{H}^{t^*}}{||\mathbf{H}||^2}\mathcal{T}\hat{A}_{2N-1}(k)-a(k) \qquad (11)$$

where $\hat{A}_{2N-1}(k)$ contains the $N-1$ past and $N-1$ future decisions w.r.t. $a(k)$.

The above description of the forward and feedback filters, however, does not take into account the edge effects, which are due to the fact that we are processing in a burst mode. The expressions in (9) remain valid though. The edges correspond to the $N-1$ first and $N-1$ last symbols of the burst. In continuous processing, where all the data necessary for detection is assumed to be available, we need N present and future received samples, and $N-1$ future and $N-1$ past symbols for a channel impulse response of length N. Burst mode processing corresponds to continuous processing, except for the two edges.

Consider again fig. 1: at the edges, the vector $\mathcal{T}_a(k)$ does not contain all the coefficients of the channel. It is a truncated version of the continuous processing forward filter. The edge feedback filter is not a truncated version of the continuous processing feedback filter, as it results from the multiplication of a truncated version of the matched filter by a truncated version of the convolution matrix $\mathcal{T}$.

To sum up, the forward and feedback filters are time-varying at the edges, and time-invariant in between, corresponding then to the continuous processing case.

3.5 Continuous Processing Case

We see then that it is immediate to extend the results obtained for burst detection to the case of continuous processing.

The forward filter is simply the matched filter to the multichannel filter $\mathbf{H}(z)$, i.e. $\mathbf{H}^{\dagger}(z)$, where $\mathbf{H}^{\dagger}(z) = \mathbf{H}^{H}(1/z^*)$, and the feedback filter $\mathbf{H}^{\dagger}(z)\mathbf{H}(z) - ||\mathbf{H}||^2$. The scaling factor $||\mathbf{H}||^{-2}$ is added after the feedback loop. Figure 2 shows the set-up of the continuous NCDFE.

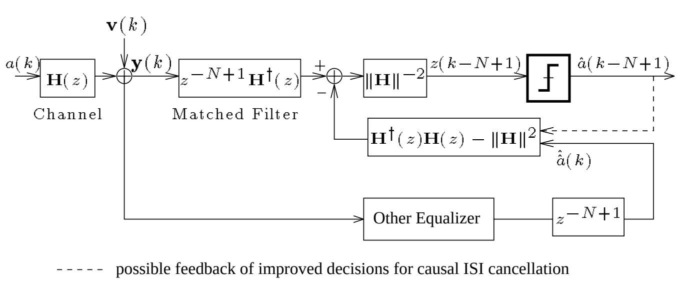

Figure 2: Continuous NCDFE

The future decisions are given by another (classical) equalizer (linear, or causal DFE), the past symbol decisions are either given by the NCDFE itself, or by the first equalizer. The output of the NCDFE can be used as feedback for tation would be the same as the one above, except for the processing of the edges, for which the filters are time-varying. This structure represents an alternative to the Viterbi algorithm, less complex as it consists of FIR filtering and slicing.

3.6 MMSE Criterion

In [2] and [3], a similar non-causal DFE structure was determined starting from the MMSE criterion:

$$\min_{F,B} |F(q)\boldsymbol{y}(k) - B(q)a(k) - a(k)|^2 \qquad (12)$$

where F is the forward filter and B the feedback filter. We show now that our solution corresponds to the unbiased MMSE solution.

It is easy to see from criterion (12) that the feedback filter is equal to the cascade of the channel and the forward filter, from which the central coefficient is eliminated. The feedback filter can be written as $F\mathcal{T}_{\bar{A}}(k)$.

According to the decomposition (6), if we denote $Z = \boldsymbol{Y} - \mathcal{T}_{\bar{A}}(k)\bar{A}(k) = \mathcal{T}_a(k)a(k) + V$, the input to the decision device is:

$$z(k) = FZ = F\mathcal{T}_a(k)a(k) + FV \qquad (13)$$

The MMSE solution to (12) is:

$$z_{MMSE}(k) = \sigma_a^2\mathcal{T}_a^H(k)\left(\sigma_a^2\mathcal{T}_a(k)\mathcal{T}_a^H(k) + \sigma_v^2 I\right)^{-1} Z \qquad (14)$$

Using the Matrix Inversion Lemma:

$$z_{MMSE}(k) = \frac{\mathcal{T}_a^H(k)}{\mathcal{T}_a^H(k)\mathcal{T}_a(k) + \frac{\sigma_v^2}{\sigma_a^2}} Z \qquad (15)$$

$$z_{MMSE}(k) = \frac{\mathcal{T}_a^H(k)\mathcal{T}_a(k)}{\mathcal{T}_a^H(k)\mathcal{T}_a(k) + \frac{\sigma_v^2}{\sigma_a^2}} a(k) + \frac{\mathcal{T}_a^H(k)}{\mathcal{T}_a^H(k)\mathcal{T}_a(k) + \frac{\sigma_v^2}{\sigma_a^2}} V \qquad (16)$$

Let's compare it with our ML solution:

$$z_{ML}(k) = \frac{\mathcal{T}_a^H(k)}{\mathcal{T}_a^H(k)\mathcal{T}_a(k)} Z = a(k) + \frac{\mathcal{T}_a^H(k)}{\mathcal{T}_a^H(k)\mathcal{T}_a(k)} V \qquad (17)$$

We notice that the ML estimate of $a(k)$ is unbiased, but also we have the following relationship:

$$z_{ML}(k) = \left(\frac{\mathcal{T}_a^H(k)\mathcal{T}_a(k)}{\mathcal{T}_a^H(k)\mathcal{T}_a(k) + \frac{\sigma_v^2}{\sigma_a^2}}\right)^{-1} z_{MMSE}(k) \qquad (18)$$

It was proven in [4] that this simple scaling of the MMSE estimate leads to the optimum unbiased MMSE estimator.

The structure we derived from the ML criterion corresponds to the unbiased MMSE NCDFE. This estimator, although it increases the SNR at the output of the equalizer, has the advantage to reduce the error probability when the input constellation is not constant modulus.

4 Blind Adaptation

4.1 Blind Estimation Algorithm

We now assume that we do not know the channel, as well as the input symbols. Our purpose is to find a blind algorithm to determine them. Our minimization criterion is:

$\boldsymbol{H},a(k)\in \mathrm{A}$

To solve it we use a technique of alternating minimizations, where we minimize alternatively w.r.t. the channel coefficients and w.r.t. the input symbols.

Here is the principle of the algorithm, where an alternating minimization is used at step 1 and 2:

0. The initialization is furnished by another classical blind algorithm which gives symbol estimates. Alternatively, the blind algorithm may provide a channel estimate, with which the symbols are determined using a classical equalizer (linear or DFE).

1. If we suppose the symbols known, then the estimation of the channel is the one obtained from a training sequence by the least-squares criterion. By exploiting the commutativity of the convolution, we get: $\mathcal{T}A = \mathcal{A}\boldsymbol{H}^{tT}$, where:
$$\mathcal{A} = A_{M,N} \otimes I_m \quad (\otimes = \text{tensor product})$$
$$A_{M,N} = \begin{bmatrix} a(-N+1) & a(-N+2)\cdots & a(0) \\ a(-N+2) & \cdot^{\cdot^{\cdot}} \quad \cdot^{\cdot^{\cdot}} & \vdots \\ \vdots & \cdot^{\cdot^{\cdot}} \quad \cdot^{\cdot^{\cdot}} & \vdots \\ a(M-N+2) & \cdots & \cdots a(M-1) \end{bmatrix}$$
Criterion (5) admits as solution: $\boldsymbol{H}^{tT}=(\mathcal{A}^H\mathcal{A})^{-1}\mathcal{A}^H\boldsymbol{Y}$

2. If we suppose the channel known, each symbol $a(k)$ of the burst is determined by the NCDFE, according to equation (8). We have two possibilities for past symbol decision feedback. We can use the previous iteration of the algorithm for past and future decisions w.r.t. a(k). But we can also use the past decisions given by the NCDFE during the current iteration, the future decisions still being given by the previous iteration. Then we return to point 1.

5 Simulations

For all simulations, we consider bursts of length 100. The input symbols, of variance $\sigma_a^2 = 1$, are -1 and 1. Different SNRs are also considered: 7dB and 10dB. What we call SNR is the average SNR per subchannel at the channel output. It is defined as $SNR = \dfrac{1}{m}\dfrac{\|\mathbf{H}\|^2\sigma_a^2}{\sigma_v^2}$.

The measure chosen is the error probability: we average the number of errors at the equalizers output over different random instances of the input burst symbols and noise and over some symbols of the burst. In burst mode, performance depends on the position of the symbol in the burst: when no symbols at the edges are known, the performance degrades at the edges w.r.t. to the middle. When at least $N-1$ symbols are known at each edge, the performance gets better at the edges compared to the middle. In the simulations we consider the average number of errors over all symbols in the burst but also over the 50 central symbols for which performance is asymptotically equivalent to the continuous processing level.

In a first simulation, we compare the performance of the classical DFE and the NCDFE. The channel is assumed known; its coefficients are randomly chosen ($N = 7$ and $m = 3$). The output of the DFE is used to initialize the NCDFE; 5 iterations of the NCDFE are then done. In fig. 3, we plotted the average number of output errors for SNR=7dB and SNR=10dB, as well as the Matched Filter Bound (MFB) and the ideal error probabilility of the DFE (for an unbiaised equalizer, the error probability can be found from the MSE using the Gaussian approximation [4]): in the left figure all the symbols of the burst are considered for averaging, in the right figure only the middle symbols are considered. The error probability of the DFE corresponds to iteration 0.

We observe that the NCDFE improves performance w.r.t. the DFE as more and more iterations are done. It gets below the ideal error probabilities of the DFE. It can be noted that the simulated error probability are higher than the ideal ones: this is due to the error propagation.

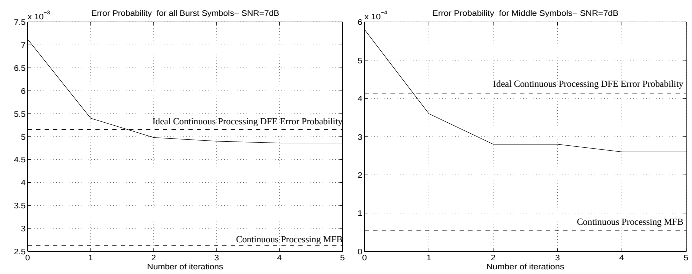

Figure 3: Error Probabilities for the DFE and the NCDFE.

5.2 Blind Algorithm Simulations

To illustrate the blind algorithm used to adapt the NCDFE, we propose some simulations. The channel considered has length 3 (N=3) and 2 subchannels (m=2):

$$\boldsymbol{H} = \begin{bmatrix} 1.1650 & 0.0751 & -0.6965 \\ 0.6268 & 0.3516 & 1.6961 \end{bmatrix}.$$

The algorithm is initialized by estimates of the symbols given by a Burst Multichannel Zero-Forcing Linear Equalizer (ZFLE). This Zero-Forcing Equalizer uses itself a blind estimate of the channel. As a first approach, we model this blind estimate by taking a perturbation of the true channel: $\boldsymbol{H}^{(0)} = \boldsymbol{H} + \widetilde{\boldsymbol{H}}$, where the coefficients of $\widetilde{\boldsymbol{H}}$ are gaussian, i.i.d., of variance $\alpha\dfrac{\mathrm{trace}(\boldsymbol{H}^H\boldsymbol{H})}{mN}$, where $\alpha =$ [0 0.1 0.2 0.3 0.4 0.5]. For $\alpha = 0$, we simply evaluate the performance of the (non-blind) NCDFE. Five iterations of the algorithm are done.

In fig. 4, the error probability for SNR=7dB and SNR=10dB is plotted. We see a good behavior of the algorithm for small values of α ($\alpha = 0$ and 0.1): the error probability is close to the MFB, and inferior or close to the error probability of the ZFLE. However for greater values of α, the algorithm seems to converge towards local minima, nish a good initialization to our blind algorithm. Remark that in the multichannel case, with sufficient diversity, the ZFLE performance is not that far from the MFB.

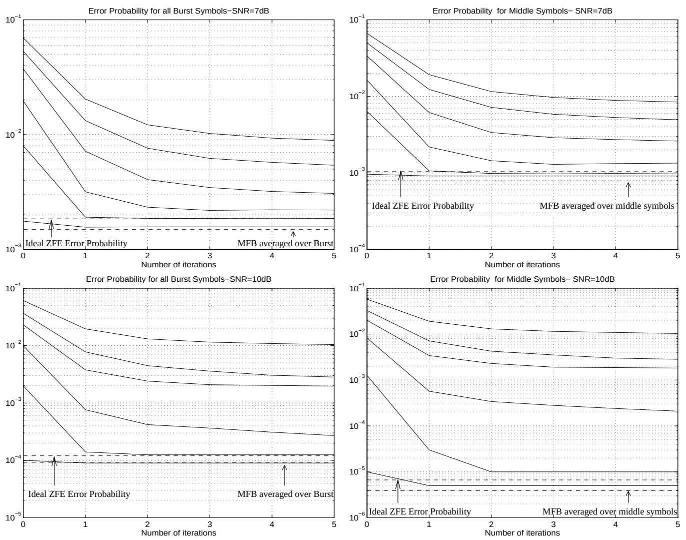

Figure 4: Error Probabilities over all symbols and middle symbols of the burst for $\alpha = 0$ (lower curve) to $\alpha = 0.5$ (upper curve) for SNR=7dB and SNR=10dB.

In a second approach, the channel is initialized by a blind algorithm based on multichannel linear prediction [5]. Results are shown in figure 5 where the error probability over all symbols burst at each iteration for blind initialization is compared to the non blind case for SNR=7dB and SNR=7dB.

5.3 Blind Algorithm Improvements

The proposed blind algorithm with blind initialization works well in our example. However, some channels cannot be well estimated by blind techniques and a bad initialization could result in a convergence to a local minimum.

Several solutions to improve the algorithm were tried. Instead of taking the decisions of the previous iteration as past decisions, we took the decisions of the current iteration, hoping this would reduce error propagation. Simulations did not shown any significant improvements w.r.t. to the simulations presented in section 5.2.

We also compared the middle error probability in the case where no symbols at the edges are known and in the case where N-1 symbols are known at each edges. In the last case, the error probability at the edges being lower than at the middle, it could result in less error propagation. Once again simulations did not show significant improvements.

Another variation we did not exploit yet is to make soft decisions instead of hard decisions.

5.4 Channel Estimation Performance

The previous simulation study concentrated on the detection performance of the symbols. However, the blind equalization algorithm considered here provides an estimate of both the symbols and the channel. The main point we want to make here is that the exploitation of the finite alphabet nature of the symbols significantly improves the blind estimation performance of the channel, compared to the recent wave of blind channel estimation algorithms that are based on the second-order statistics of the received signal only. The following reasoning leads to a lower bound of the channel estimation error covariance matrix. By exploiting the finite alphabet of the symbols, the symbols actually get detected. Assume that these detections are error-free. Then all symbols present in the problem act as training sequence for the channel estimation. Since we use the ML criterion, the channel estimation error covariance matrix will hence equal the Cramer-Rao bound (CRB) for training-sequence based channel estimation, with all symbols constituting the training sequence. Due to detection errors, this training sequence CRB only constitutes a lower bound on the channel estimation error covariance matrix. However, the bound will be tight if the probability of error (using the ML channel estimate) is low. In summary, since the burst length will normally be significantly longer than typical training sequences, the channel estimation performance of a blind approach that exploits the symbol alphabet will normally be better than that of a classical training-sequence based approach.

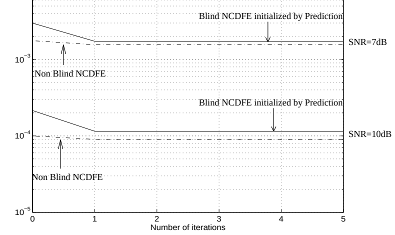

Figure 5: Error Probabilities over all burst symbols for SNR=7dB and SNR=10dB with blind channel initialization.

References

[1] J. G . Proakis. "Adaptive Non Linear Filtering Techniques for Data Transmission". *IEEE Symp. on Adaptive Processes*, pages XV.2.1–5, 1970.

[2] A. Gersho and T. L. Lim. "Adaptive Cancellation of Intersymbol Interference for Data Transmission". *Bell Syst. Tech. J.*, 60(11):1997–2021, Nov. 1981.

[3] M. S. Mueller and J. Salz. "A Unified Theory for Data-Aided Equalization". *Bell Syst. Tech. J.*, 60(9):2023–2038, Nov. 1981.

[4] J. M. Cioffi, G. P. Dudevoir, M. V. Eyuboglu, and G. D. Forney Jr. "MMSE Decision-Feedback Equalizers and Coding - Part I: Equalization Results". *IEEE Trans. Com.*, 43(10):2582–2594, Oct. 1995.

[5] E. de Carvalho and D. T. M. Slock. "Maximum-Likelihood Blind Equalization of Multiple FIR Channels". In *Proc. ICASSP 96 Conf.*, Atlanta, USA, May 1996.