Blind Turbo Equalization for Wireless DPSK Systems

Xiao-Ming Chen and Peter A. Hoeher
Information and Coding Theory Lab
Faculty of Engineering, University of Kiel
Kaiserstr. 2, D-24143 Kiel, Germany
E-mail: {xc, ph} @tu-kiel.de

Abstract— A turbo equalization scheme consisting of an inner blind soft-in soft-out (SISO) equalizer and an outer SISO channel decoder is proposed and investigated. The receiver is suitable for wireless differential phase shift keying (DPSK) systems. The basic structure of the blind SISO equalizer consists of an iterative approximation of the joint maximum-likelihood (ML) solution with respect to (w.r.t.) channel estimation and sequence detection. The approximation is enhanced by adaptive channel estimation and by using a priori information delivered by the outer channel decoder. The DPSK/BI super trellis is explicitly exploited for blind equalization rather than just doing differential decoding for resolving the problem of phase ambiguity. Phase ambiguity and shift ambiguity associated with trellis-based blind equalization are discussed for coded systems. Furthermore, new approaches are proposed to combat the phase ambiguity and the shift ambiguity.

Keywords— Blind turbo equalization, joint equalization and channel estimation, maximum-likelihood sequence estimation, iterative processing

I. INTRODUCTION

In digital mobile communications, the two main distortions are intersymbol interference (ISI) caused by time dispersion of the multipath channel, and Doppler spread due to the movement of the mobile receiver. To combat ISI, a variety of equalization algorithms have been developed in the last decades [1], [2].

An obvious problem coupled with wireless equalization is channel estimation. Typically, the channel impulse response (CIR) is estimated by inserting a training sequence in each data block. For more efficient usage of bandwidth, blind equalization techniques [3] attract more and more attention. Furthermore, blind detection schemes may be embedded in existing systems as an add-on in order to improve the system performance in difficult environments. Blind equalization techniques may be classified in three categories.

The first category is based on statistical properties of the received signal, which is either sampled at baud rate or oversampled. In the former case, it is necessary to exploit high order statistics [4], [5]. In the latter case, second order statistics are sufficient because fractionally-spaced channel outputs are cyclostationary [6]. Possible problems with these types of statistical approaches are, depending on the algorithm, a slow convergence rate, a possible convergence to local minima, and a lack of robustness against Doppler spread, noise, and interference. On the other hand, dramatic improvements were reported in the last few years, particularly for mobile radio applications [7], [8].

The second category exploits the algebraic structures of oversampled systems, which can be modeled as multichannel systems, in order to estimate CIRs. In a so-called deterministic approach [9], CIRs of the polyphase channels are estimated by solely exploiting the channel outputs rather than estimating statistical terms. The cross correlation between different channel outputs and corresponding CIRs is used to form a set of linear equations. This technique is closely connected to subspace methods for blind equalization [10]. Possible drawbacks of these methods are constraints w.r.t. the channel matrix; conditions on the identifiability of channels are given in [9].

The third category covers trellis-based and tree-based methods. Classically, an adaptive channel estimator (e.g. based on the least mean square (LMS) or recursive least-squares (RLS) algorithm) is implemented in parallel to a trellis-based equalizer (e.g. a ML sequence estimator). The most likely survivor, given a suitable decision delay, is fed into the channel estimator, i.e., this approach estimates the channel and the data jointly [1]. The optimum ML joint data and channel estimator in the absence of training (as well as suboptimum variations thereof) was presented in [11], [12]. Note that the per-survivor-processing technique [13] may be considered as a special case of the technique introduced in [11]. In order to exploit properties of the multipath fading channel and to track the time varying channel, model-fitting algorithms were used in [14], [15], [16], for example. While all of these trellis-based techniques are applied for coherent detection, non-coherent blind equalization techniques are an interesting alternative area of current research [17], [18], [19], [20].

Our focus is on trellis-based algorithms, suitable for short block sizes and noisy environments. This goal is approached by iterative processing of the ML joint data and channel estimator [11]. Significant improvements are par-

1This work is supported by Germany Science Foundation (DFG) under Grant No. Ho 2226/1.
ticularly obtained by incorporating adaptive channel estimation [21], [1] into the equalizer and by using a priori information, if available [24]. The possibility of using a priori information of the data symbols is a nice feature of trellis-based algorithms. Training corresponds to perfect a priori information for all symbols, whereas a (truly) blind mode corresponds to no a priori information. Furthermore, with trellis-based algorithms it is possible to deliver reliable soft outputs and to perform iterative processing between the equalizer and a channel decoder. As opposed to the common understanding that differential encoding is used just to resolve the phase ambiguity both in channel and data estimation, we explicitly use the structure of the recursive DPSK/ISI super trellis [24].

In Section II, we describe the system model under investigation and formulate the joint estimation problem. The proposed blind turbo processor is presented in Section III, which also discusses the distinct novel features. Simulation results are provided in Section IV. Finally, conclusions are drawn in Section V.

II. SYSTEM MODEL

Throughout this paper we use the complex baseband notation.

A. Transmitter

We consider an M-ary DPSK system. The output symbols of the differential encoder can be written as

\[ x(k) = x(k-1) \cdot d(k), \quad x(0) = +1, \]  

where \( d(k) \), \( k > 0 \), are M-ary PSK symbols, which may be interleaved outputs from the outer channel encoder, and \( x(0) \) serves as a reference symbol.

B. Channel Model

The pulse shaping filter, multipath fading channel, receiving filter, and the sampling can be represented by a tapped delay-line baud-rate model [1]. (We restrict ourselves to baud-rate sampling, an extension to fractionally-spaced sampling is straightforward.) For simplicity, the channel is assumed to be time-invariant within a data block. (The proposed receiver is able to track a time-varying channel, however.) The corresponding outputs of the equivalent discrete-time channel model can be written as

\[ y(k) = \sum_{l=0}^{L} x(k-l) \cdot h_i + n(k), \]  

where \( \{h_i\}, 0 \leq l \leq L \), are the coefficients of the discrete-time channel model (in the following referred to as the channel coefficients), \( L \) is the effective channel memory length (after suitable truncation), and \( \{n(k)\} \) is a sequence of additive noise samples. In the following, \( \{n(k)\} \) is assumed to be an i.i.d additive white Gaussian noise (AWGN) sequence with one-sided power spectral density \( N_0 \). \( L \) is assumed to be known at the receiver.

In most mobile communication systems the data symbols are transmitted block-wise. If we assume \( K \) data symbols per block (excluding the reference symbol), we can write (2) in matrix/vector notation as:

\[ y = X \cdot h + n, \]  

where

\[ y = [y(0), y(1), \ldots, y(K)]^T, \quad h = [h_0, h_1, \ldots, h_L]^T, \quad n = [n(0), n(1), \ldots, n(K)]^T, \]

and

\[ X = \begin{bmatrix} x(0) & 0 & \cdots & 0 \\ x(1) & x(0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x(L) & x(L-1) & \cdots & x(0) \\ \vdots & \vdots & \ddots & \vdots \\ x(K) & x(K-1) & \cdots & x(K-L) \end{bmatrix}. \]

(If the data symbols are not transmitted block-wise, if the block size is large, or if the channel is fast time-varying, \( K \) may define a subblock.)

C. Receiver

The task of the receiver is twofold: Primarily, we are interested in an estimate of the data vector \( d = [d(1), \ldots, d(K)]^T \). A coherent receiver must also obtain estimates of each element of \( h \) in amplitude and phase.

For coherent DPSK reception, we can perform joint channel and data estimation based on the ISI trellis followed by differential decoding, or directly using the DPSK/ISI super trellis, which combines the differential encoder and the ISI trellis. In the following, we operate on the DPSK/ISI super trellis. It is easy to show that the super trellis and the ISI trellis have the same number of states.

Denoting \( \hat{x} \) and \( \hat{h} \) as hypotheses for the data vector and the channel coefficient vector, respectively, we can formulate the joint channel/data estimation problem according to the ML-criterion in the presence of AWGN as

\[ \left( \hat{x}, \hat{h} \right) = \arg\max_{\tilde{x}, \tilde{h}} \left\{ p(\tilde{y} | \tilde{x}, \tilde{h}) \right\} \]

\[ = \arg\min_{\tilde{x}, \tilde{h}} \sum_{k=0}^{K} \left| y(k) - \sum_{l=0}^{L} \hat{x}(k-l) \cdot \hat{h}_l \right|^2 \]
\begin{equation}
\hat{\mathbf{h}} = \arg \min_{\mathbf{h}} \| \mathbf{y} - \hat{\mathbf{X}} \cdot \mathbf{h} \|_F^2,
\end{equation}

where we assume that the noise variance is constant within a block and where we (for the moment) assume that no prior information about the data sequence is available. Here, \( p(\mathbf{y} | \hat{\mathbf{X}}, \hat{\mathbf{h}}) \) is the probability density function of the received vector conditioned on data and channel hypotheses, and \( \hat{\mathbf{x}} \) and \( \hat{\mathbf{h}} \) are the corresponding estimates. The notation \( \| \cdot \|_F \) is the Frobenius norm with \( \| \mathbf{A} \|_F^2 = \text{tr}(\mathbf{A}^H \cdot \mathbf{A}) \), which is equivalent to the Euclidean distance if \( \mathbf{A} \) is a column vector. Given no prior information about the channel coefficients, the optimal solution for the joint estimation problem (5) is a least-squares channel estimator for each possible hypothesis \( \hat{x} \). The ML-estimate is the pair \((\hat{\mathbf{x}}, \hat{\mathbf{h}})\) which minimizes the Euclidean distance. The complexity of this exhaustive search approach inhibits its practical applications, however.

The joint minimization problem in (5) can also be formulated, without loss of performance, as a two-step minimization:

\begin{equation}
(\hat{\mathbf{x}}, \hat{\mathbf{h}}) = \arg \min_{\hat{\mathbf{h}}} \left\{ \min_{\hat{\mathbf{x}}} \| \mathbf{y} - \hat{\mathbf{X}} \cdot \hat{\mathbf{h}} \|_F^2 \right\}.
\end{equation}

Equation (6) motivates an intuitive iterative procedure, where data estimation is performed given channel estimates from the last iteration:

\begin{equation}
\hat{\mathbf{h}}^{(i+1)} = \arg \min_{\hat{\mathbf{h}}} \| \mathbf{y} - \hat{\mathbf{X}}^{(i+1)} \cdot \hat{\mathbf{h}} \|_F^2.
\end{equation}

and where afterwards the channel coefficients are evaluated based on the data estimates from (7):

\begin{equation}
\hat{\mathbf{x}}^{(i+1)} = \arg \min_{\hat{\mathbf{x}}} \| \mathbf{y} - \hat{\mathbf{X}}^{(i+1)} \cdot \hat{\mathbf{h}} \|_F^2.
\end{equation}

This procedure may be continued until a convergence is observed:

\begin{equation}
\hat{\mathbf{x}}^{(i+1)} = \hat{\mathbf{x}}^{(i)}.
\end{equation}

or

\begin{equation}
\hat{\mathbf{h}}^{(i+1)} = \hat{\mathbf{h}}^{(i)}.
\end{equation}

Note, however, that the two-step iterative scheme (7) and (8) is suboptimal since not all hypotheses are tested. Typically, (7) is solved by means of the Viterbi algorithm (VA) and (8) is solved by means of least-squares (LS) channel estimation:

\begin{equation}
\hat{\mathbf{h}}^{(i+1)} = (\hat{\mathbf{X}}^{H (i+1)} \cdot \hat{\mathbf{x}}^{(i+1)})^{-1} \cdot \hat{\mathbf{X}}^{H (i+1)} \cdot \mathbf{y} = (\hat{\mathbf{X}}^{(i+1)})^\dagger \cdot \mathbf{y},
\end{equation}

where the superscript \((\cdot)^H\) denotes the Hermitian transpose and \((\hat{\mathbf{X}}^{(i+1)})^\dagger\) is the pseudo inverse of \(\hat{\mathbf{X}}^{(i+1)}\).

This two-step iterative method is referred to as the conventional iterative scheme in the following. Given initial channel estimates obtained by training and using (10) as convergence criterion, this technique has been investigated in [22]. Excellent results have been reported in this context. In the context of blind equalization, however, we observed that the convergence speed is much slower and that the possibility of a convergence to local minima is much higher.

Another suboptimal iterative technique is based on the expectation-maximization (EM) algorithm [25], which iteratively generates channel estimates with non-decreasing likelihood. Common drawbacks of suboptimal iterative methods are a slow convergence speed and a possible convergence to local minima.

III. PROPOSED RECEIVER STRUCTURE

We propose a blind turbo receiver, which consists of an adaptive blind SISO equalizer and a SISO channel decoder.

A. Blind SISO Equalizer

The main features of the blind SISO equalizer are as follows:

- **We incorporate an adaptive channel estimator within a trellis-based soft-output equalizer operating on the DPSK/ISI super trellis.** Due to the channel estimator operating on the super trellis, data estimation and channel estimation can be solved in just one step. Suitable channel estimators are for example based on the LMS algorithm or the RLS algorithm, which converges faster than the LMS algorithm at the price of additional complexity [21]. Suitable equalizers are sequence oriented, such as the soft-output Viterbi algorithm (SOVA), the so-called Max-Log-APP algorithm [26] (APP: a posteriori probability), or reduced-state sequence estimators (RSSE) with soft outputs [27]. The main advantage of the joint equalizer/channel estimator is that the convergence rate (i.e. the acquisition behavior) improves significantly compared to the conventional iterative scheme. Simultaneously, the adaptive channel estimator is capable to track time-varying channels to some extent.

- **We tackle the problem of local minima.** An important observation is as follows: Typically, blind equalizers are only able to provide estimates of the product \( x(k-l) \cdot h_l \) for all \( 0 \leq l \leq L \), compare (5). In the binary case this implies that \( \hat{x}(k) = \pm x(k-l) \) all are possible decisions, where \( k \in \{0, \pm 1, \pm 2, \ldots, \pm L\} \). The sign ambiguity is well known as the phase ambiguity problem, which is solved by differential encoding. The shift ambiguity also has been reported in the literature for trellis-based blind equalization with/without channel order mismatch [11], [23]. How-
ever, no suitable method has been proposed to resolve the
shift ambiguity problem. Since the corresponding noiseless
channel estimates are $\hat{h}_t = \pm h_{t+n}$, any $\kappa \neq 0$ corresponds to a local minimum. Besides this class of minima, other local minima exists as well. The main problem related to
the shift ambiguity is that $\kappa$ channel coefficients are shifted out of the observation interval $L + 1$. In order to combat
shift ambiguity, in [24] we proposed to expand the observation
interval and to maximize the power of estimated channel coefficients within the original observation interval. An alternative solution of the shift ambiguity problem is to shift the estimated data vector before performing the corresponding LS channel estimation (compare (5) and (11)):
\begin{equation}
\hat{k} = \arg\min_m \| y - \hat{X}^{(m)} \cdot (\hat{X}^{(m)})^\dagger \cdot y \|_F^2,
\end{equation}
where $\hat{X}^{(m)}$ is constructed from the estimated data sequence $\hat{x}$ according to $\hat{x}_k^{(m)} = \hat{x}_{k+m}$. The minimization in (12) is over all possible shift indices $m$ with $-L \leq m \leq L$. In the latter solution, equation (12), the size of the DPSK/ISI super trellis does not have to be expanded.

We take a priori information into account, which is provided by the outer channel decoder. This also helps to solve
the problem of local minima.

As mentioned before, a nice feature of trellis-based blind
equalization is the possibility to make use of a priori information. If a priori information is available, we can replace
the ML sequence estimator assumed so far by a maximum
a posteriori (MAP) sequence estimator. Then, the branch
metrics of the DPSK/ISI super trellis are modified as [28],
[26]:
\begin{equation}
\gamma'(S^{<i-j>}(k-1) \rightarrow S^{<j>}(k))
= -\frac{1}{2 \sigma^2_n} \cdot \left| y(k) - \sum_{l=0}^{L} x^{\langle i-j \rangle}(k-l) \cdot \hat{h}_l(k-1) \right|^2
\end{equation}
+ \log P(d^{<i-j>}(k)),
\end{equation}
where $\gamma'(S^{<i-j>}(k-1) \rightarrow S^{<j>}(k))$ denotes a state transition
and $\sigma^2_n$ is the noise variance. Moreover, $P(d^{<i-j>}(k))$
denotes the a priori probability of the state transition
$\gamma'(S^{<i-j>}(k-1) \rightarrow S^{<j>}(k))$.

For binary systems, (13) can be rewritten as
\begin{equation}
\gamma'(S^{<i-j>}(k-1) \rightarrow S^{<j>}(k))
= -\frac{1}{2 \sigma^2_n} \cdot \left| y(k) - \sum_{l=0}^{L} x^{\langle i-j \rangle}(k-l) \cdot \hat{h}_l(k-1) \right|^2
+ \frac{1}{2} L(d(k)) \cdot d^{<i-j>}(k),
\end{equation}
where $L(d(k))$ denotes the given log-likelihood ratio (LLR)
of symbol $d(k)$, compare (1). (Symbol-by-symbol MAP
estimation is not recommendable here due to the lack of
survivors; the locally best surviving paths are necessary for
channel estimation.) The significance of (13) and (14) is a
generic receiver structure, which is the same for the wide
range of blind equalization without a priori information
(where $L(d(k)) = 0$ for all $k$) to a training-based equalizer
(where $|L(d(k))| \rightarrow \infty$ for some $k$). For more details on
the blind SISO equalizer see [24].

B. Blind Turbo Processor

Furthermore, besides incorporating a priori information,
our trellis-based equalizer is capable to deliver soft outputs
to subsequent processing stages, e.g. the outer channel
decoder. Therefore, the “turbo principle” [29] can be applied
in the receiver consisting of the adaptive SISO blind
equalizer and a channel decoder. The overall scheme is called
blind turbo processor in the following.

Two different turbo equalization schemes have been proposed
based on the iterative processing method introduced
for serial turbo codes [30], [31]. In [30], the ISI channel is interpreted as an inner encoder. In [31], the trellis-based
equalizer is replaced by a soft interference canceler based on
the minimum mean squared error (MMSE) criterion with
the intention of complexity reduction. The interference
canceler consists of a matched filter and an ISI estimator.
Recently, blind turbo equalization techniques have been
proposed in [32], [33], where the channel coefficients and
the noise variance were estimated iteratively using the EM
algorithm [25], and in [34], where a blind channel estimator
based on high order statistics is used. The latter technique
[34] is suitable for fading channels. Our approach is
suitable for short blocks. Moreover, the unknown CIR and
data sequence are jointly estimated on the DPSK/ISI super
trellis, which makes our method different from previously
proposed approaches. Phase ambiguity and shift ambiguity
problems are taken into consideration, which may make
our approach much more robust than related algorithms.

The system model and the detailed turbo processor are illustrated in Fig. 1 and Fig. 2, respectively.

In Fig. 1, $u$ and $d'$ are the data vectors before and after
channel encoding, respectively. Note from Fig. 1 that
the “inner encoder” (represented by the DPSK/ISI super
trellis) is recursive, which is an important feature of serially
concatenated turbo codes [35]. This feature is missing in the ISI turbo schemes [30], [31], [32], [33].

In Fig. 2, $L_a(\cdot)$, $L_E(\cdot)$ and $L_D(\cdot)$ denote a priori information and a posteriori information obtained through the equalizer and channel decoder, respectively. $L_E(\cdot)$ and $L_D(\cdot)$ are the corresponding extrinsic information. Only extrinsic information should be exchanged between the adaptive blind equalizer and the channel decoder.

For simplicity, we consider the binary systems in the following and discuss the impacts of phase/shift ambiguity on the blind turbo processor. New approaches are also proposed to solve these problems.

B.1 Discussion about Phase Ambiguity

Assuming we have got the channel estimate $\hat{h} = -h$, the branch metric for blind equalization can be formulated as

$$
\gamma'(S_i^{<j>}(k-1) \rightarrow S_i^{<j>}(k)) = -\frac{1}{2\sigma_n^2} \left| y(k) + \sum_{l=0}^{L} x_i^{<i-j>}(k-l) \cdot h_l \right|^2 + \frac{1}{2} L(d(k)) \cdot d_i^{<i-j>}(k).
$$

The corresponding term with known CIR is

$$
\gamma(S_i^{<j>}(k-1) \rightarrow S_i^{<j>}(k)) = -\frac{1}{2\sigma_n^2} \left| y(k) - \sum_{l=0}^{L} x_i^{<i-j>}(k-l) \cdot h_l \right|^2 + \frac{1}{2} L(d(k)) \cdot d_i^{<i-j>}(k).
$$

Comparing (15) with (16), we get the following relationship:

$$
\gamma'_{i \rightarrow j;k} = \gamma'(x_i^{<i-j>}(k-L), \ldots, x_i^{<i-j>}(k)) = \gamma(-x_i^{<i-j>}(k-L), \ldots, -x_i^{<i-j>}(k)).
$$

Given the same initial conditions for the forward/backward recursion in the Max-Log-APP algorithm [26], we have

$$
\alpha'_{i;j;k} = \alpha'(x_i^{<j>}(k-L+1), \ldots, x_i^{<j>}(k)) = \alpha(-x_i^{<j>}(k-L+1), \ldots, -x_i^{<j>}(k)),
$$

where $\alpha'_{i;j;k} = \log p(y_{t \leq k}, S_i^{<j>}(k); \hat{h})$ and $t$ denotes the time index.

Similarly, $\beta'_{i;k-1} = \log p(y_{t \geq k} | S_i^{<j>}(k-1); \hat{h})$ can be obtained:

$$
\beta'_{i;k-1} = \beta'(x_i^{<j>}(k-L), \ldots, x_i^{<j>}(k-1)) = \beta(-x_i^{<j>}(k-L), \ldots, -x_i^{<j>}(k-1)).
$$

Finally, the LLR about $d_k$ can be achieved:

$$
L'(d_k) = \max_{\tilde{d}(k) = 1} \{ \alpha'_{i;k-1} + \gamma'_{i \rightarrow j;k} + \beta'_{i;k} \} - \max_{\tilde{d}(k) = -1} \{ \alpha'_{i;k-1} + \gamma'_{i \rightarrow j;k} + \beta'_{i;k} \} - \max_{\tilde{d}(k) = 1} \{ \alpha'_{i;k-1} + \gamma'_{i \rightarrow j;k} + \beta'_{j;k} \} - \max_{\tilde{d}(k) = -1} \{ \alpha'_{i;k-1} + \gamma'_{i \rightarrow j;k} + \beta'_{j;k} \} = L(d_k),
$$

i.e., using differential encoding we obtain the correct LLR values about data symbols under the condition $\hat{h} = -h$.

However, we must be careful with $L'(d_k)$ due to the differential encoding: $x(1) = x(0) d(1) = d(1)$. If the phase reference symbol $x(0)$ is also transmitted, (20) can be applied further to get $L'(d_1)$, while the receiver assumes that $x(0)$ is unknown. In this case, the set $\{L'(d_k)\}, k \geq 1$, is used for further iterative processing. Otherwise, $L'(d_1)$ is evaluated as follows:

$$
L'(d_1) = \max_{\tilde{d}(1) = 1} \{ \alpha'_{i;0} + \gamma'_{i \rightarrow j;1} + \beta'_{j;1} \} - \max_{\tilde{d}(1) = -1} \{ \alpha'_{i;0} + \gamma'_{i \rightarrow j;1} + \beta'_{j;1} \} = \max_{\tilde{d}(1) = 1} \{ \alpha'_{i;0} + \gamma'_{i \rightarrow j;1} + \beta_1 \} - \max_{\tilde{d}(1) = -1} \{ \alpha'_{i;0} + \gamma'_{i \rightarrow j;1} + \beta_1 \} = -L(d_1).
$$

If $L'(d_1)$ is delivered to the channel decoder, it can cause error propagation during the iterative processing. This negative effect associated with the phase ambiguity has been reported, without further analysis, in [32].

B.2 Discussion about Shift Ambiguity

In coded systems, the LLR values from the channel decoder may be exploited to combat the shift ambiguity problem. For simplicity, the following conditions are assumed:

$$
\hat{h}_l = h_{l + \kappa}, \quad 0 \leq l \leq L - \kappa;
$$

$$
\hat{h}_l = 0, \quad 0 \leq l \leq \kappa - 1;
$$

$$
\hat{h}_l = 0, \quad L - \kappa + 1 \leq l \leq L.
$$

Fig. 2. Blind turbo processor
Let us consider the very first iteration, where no a priori information about \( d_k \) is available. The branch metric for blind equalization is then formulated as

\[
\gamma' (S^{<i>}(k-1) \rightarrow S^{<j>}(k)) = -\frac{1}{2\sigma_n^2} \left| y(k) - \sum_{i=0}^{L} x^{<i-j>}(k-l) \cdot h_{l+n} \right|^2. \tag{23}
\]

The corresponding term using a known CIR is

\[
\gamma (S^{<i>}(k-1) \rightarrow S^{<j>}(k)) = -\frac{1}{2\sigma_n^2} \left| y(k) - \sum_{i=0}^{L} x^{<i-j>}(k-l) \cdot h_{i} \right|^2. \tag{24}
\]

New branch metrics are evaluated for different shifts according to

\[
\gamma^{(m)} (S^{<i>}(k-1) \rightarrow S^{<j>}(k)) = -\frac{1}{2\sigma_n^2} \left| y(k) - \sum_{i=0}^{L-m} x^{<i-j>}(k-l-m) \cdot h_{l+n} \right|^2, \tag{25}
\]

where \(-L \leq m \leq L\).

Comparing (25) and (24), we get the following equation:

\[
\gamma^{(m)} (S^{<i>}(k-1) \rightarrow S^{<j>}(k)) = \gamma (S^{<i>}(k-1) \rightarrow S^{<j>}(k)). \tag{26}
\]

By means of calculating the LLRs of the uncoded symbols for different shifts, \( \{L^{D(m)}(u(n))\} \), the estimated shift index can be obtained as

\[
\hat{k} = \arg\max_{m} \frac{1}{KR} \sum_{n=1}^{KR} \left| L^{D(m)}(u(n)) \right|, \tag{27}
\]

where \( R \) is the code rate of the channel encoder.

\[\text{C. Overall Receiver}\]

In the following, \( \xi^{(i)}(k) \) denotes a value at time \( k \) in the \( i \)th iteration, where \( \xi \) may be a scalar, a vector, or a matrix. With this notation, a concise description of the proposed receiver is as follows:

1. \textbf{Initialization}:
   In the absence of training data, the channel coefficient vector is initialized by a fixed vector, e.g. with elements \( \hat{h}^{(l)}_1(0) = 1/\sqrt{L + 1}, l = 0, \cdots, L \). Note that \( L \) is the channel order used in sequence and channel estimation.

2. \textbf{Adaptive channel estimation}:
   An adaptive channel estimator is incorporated in adaptive blind Max-Log-APP equalizer. During the forward recursion, the equalizer delivers tentative decisions to the channel estimator. In case of LMS adaptation, the \( i \)th iteration, \( 1 \leq i \leq N_{\text{iter}} \), can be written as [1]

\[
e^{(i)}(k-D) = y(k-D) - \hat{h}^{(i)}(k-1) \cdot \hat{x}^{(i)}_k(k-D), \tag{28}
\]

\[
\mathbf{h}(k) = \mathbf{h}(k-1) + \Delta \cdot \mathbf{e}(k-D) \cdot \hat{x}^{(i)}_k(k-D), \tag{29}
\]

where \( \hat{x}^{(i)}_k(k-D) \) is the tentative data vector (which is obtained by tracking back the trellis from the best state at time \( k \) given a certain decision delay \( D \)), the estimated channel coefficient vector, the corresponding estimation error and LMS step size, respectively.

3. \textbf{Soft-output equalization}:
   The backward recursion [26] can be performed using the transition probabilities calculated in the forward recursion, compare (13), so that we can get LLRs \( L^{E}(d(k)|y, \hat{h}^{(i)}) \) for the channel estimate of the \( i \)th iteration. \( \hat{h}^{(i)}(K) \) is the final channel estimate of the \( i \)th iteration.

4. \textbf{SESO channel decoding}:
   Using the extrinsic information from the equalizer, the channel decoding is performed using a Max-Log-APP algorithm. The extrinsic information delivered by the Max-Log-APP decoder is fed back to the inner SISO equalizer for the next iteration.

5. \textbf{Test of stopping criteria}:
   If \( i = 1 \), the possible shift in channel/data estimation is estimated using (25) and (27). LLR values corresponding the optimal shift index are used for further iteration procedure.

6. \textbf{Final data estimation}:
   The last estimated channel coefficients are selected as the channel estimate. The LLRs from the channel decoder \( L^{D}(\hat{u}) \) deliver the hard decision for uncoded data sequence and corresponding reliabilities.

If processing is done on the DPSK/ISI super trellis, there is a one-to-one mapping between the differentially encoded data sequence and the info sequence. Otherwise, differential decoding has to be performed after equalization of the ISI trellis.

IV. Simulation Results

The performance of the proposed receiver has been tested and optimized by Monte Carlo simulations. In order to emulate a system which is similar to the GSM system (GSM: Global System for Mobile communications), we investigated a binary DPSK system with a block length of \( K = 148 \) symbols. Within this paper, the channels under consideration are the two worst case static channels reported in [1] for \( L = 3 \) and \( L = 4 \), respectively:

\[
\mathbf{h}_a = [0.38, 0.6, 0.6, 0.38]^T;
\]
$$\mathbf{h} = [0.29, 0.5, 0.58, 0.5, 0.29]^T.$$  
In the concatenated system a rate $R = 1/2$ recursive, systematic convolutional code with 4 states was used. The corresponding generator polynomials are:

$$g_1(D) = 1; \quad g_2(D) = \frac{1 + D^2}{1 + D + D^2},$$

and channel decoding was performed by means of a Max-Log-APP algorithm [26]. The following design parameters were chosen:

- LMS step size: $\Delta = 0.02$;
- equalization by means of a Max-Log-APP with $2^L$ states;
- decision delay in the Max-Log-APP equalizer: 5 symbols.

Simulation results were plotted in Fig. 3 and Fig. 4. As a reference, the performance for the coded system with known CIR was selected (with just one iteration, i.e., no feedback). Note that $N_{\text{iter}}$ is the number of iterations.

![Fig. 3. BER vs. SNR for channel $h_a$](image)

For $h_a$, a better performance is achieved with just two iteration compared to the reference system using a known CIR. For $h_b$ with four iterations, the loss w.r.t the reference system is less than 1.0 dB at a bit error rate of $10^{-3}$. With more iterations, the performance can be improved further.

V. CONCLUSIONS

Based on an approximation of a blind maximum-likelihood sequence estimator (MLSE) [11], [12], a low-cost blind turbo equalizer/channel decoder is proposed. As opposed to the conventional two-step approach with alternating equalization and channel estimation, (i) an adaptive channel estimator is incorporated into a trellis-based equalizer and (ii) techniques to reduce the problem of local minima are proposed. Hence, the number of iterations and the complexity can be significantly reduced. The use of a priori information and the possibility to deliver reliability information to subsequent processing stages are powerful features of trellis-based equalizers.

Some interesting observations are as follows:

- Equalization based on the DPSK/ISI super-trellis is as complex as equalization based on the ISI trellis. The number of states is exactly the same and there is a one-to-one mapping between the states of the super trellis and the states of the ISI trellis.
- Blind equalization and the use of a priori knowledge of the data are not exclusive. In effect, a priori information helps to reduce the problem of local minima and helps to reduce the average number of iterations. The proposed receiver is suitable for the wide range from blind equalization to training based equalization without changing the receiver structure.
- Blind equalization is not only affected by a phase ambiguity, but also by a shift ambiguity.
- Our simulation results demonstrate the strength of blind coherent equalization/channel estimation techniques, particularly in the presence of an outer channel decoder.

REFERENCES
