Turbo equalisation of time varying multipath channel under class-A impulsive noise

A.E. El-Mahdy and N.M. Namazi

Abstract: The paper explores the problem of turbo equalisation and channel estimation under class-A impulsive noise. The channel is a frequency selective fading channel in which its time varying coefficients are expanded into a finite number of basis sequences and time invariant (TI) expansion parameters. Instead of the application of a maximum likelihood (ML) approach in its standard form, the proposed channel estimator is performed by an iterative approach based on the expectation maximisation (EM) algorithm and the steepest descent algorithm. The proposed estimator reduces the complexity of computations resulting from direct application of the ML approach and provides significant performance gain over the algorithms which are efficient in white Gaussian noise. Also, the proposed estimator is suitable for class-A impulsive noise and utilises the soft information obtained from the soft output Viterbi algorithm (SOVA) which is derived under class-A impulsive noise.

1 Introduction

Turbo equalisation is an iterative equalisation and decoding approach that can achieve good performance for multipath channels that suffer from intersymbol interference (ISI). The performance improvement is obtained by performing channel equalisation and channel decoding iteratively. Several articles in the area of turbo equalisation are included in literature. For example, [1] reports two new approaches that combine equalisation based on linear filtering with decoding. In the first approach, a maximum a posteriori probability (MAP) equaliser of the turbo receiver is replaced with a decision feedback equaliser (DFE) and, in the second approach, the equaliser is replaced with a linear equaliser (LE). In both approaches, the filter parameters are updated using minimum mean square error (MMSE) criteria. The two approaches are derived assuming that the channel coefficients are known. It is shown that the performance of these approaches is similar to the trellis-based receiver, while providing large savings in computational complexity. In this paper, the SOVA algorithm is used in the equaliser and decoder instead of the MAP algorithm. This is because the SOVA algorithm achieves optimal performance, despite having a reduced computational complexity. Moreover, the channel is assumed to be unknown. It is estimated using an algorithm that combines the EM algorithm with steepest descent algorithm and utilises the soft output of the SOVA algorithm. The authors in [2] present a comparative study of turbo equalisation schemes employing different classes of high-rate turbo codes, such as, block turbo codes, convolutional codes, and convolutional turbo codes. The results showed that the turbo equaliser system using convolutional turbo codes is the most robust system for all code rates investigated. Turbo equalisation is also used in various communication problems such as trellis code modulation [3] and code division multiple access (CDMA) [4]. The turbo equalisation receiver is complex since the equalisation and decoding are performed several times. Several techniques are introduced in the literature to reduce the complexity of the turbo equalisers. For example, the authors in [5] report a technique which tends to minimise the number of iterations carried by the turbo equaliser. Also the authors in [6] propose a set of receiver pre-filters to concentrate the energy of the channel in a small number of adjacent taps. It is shown that the proposed technique reduces the turbo equaliser complexity significantly at a small performance loss. A soft trellis decoding technique is proposed in [7] to reduce the complexity of the turbo equalisers. The proposed technique reduces the number of states adaptively and leads to smaller number of branch metrics to be calculated.

Most of the work on turbo equalisation assumes that the channel ambient noise is Gaussian. The work is motivated by the assumption that the transmitted data is corrupted by thermal noise. However, thermal noise is not the only noise accompanied with the communication systems. Impulsive non-Gaussian noise is present in many communication environments owing to man-made interference resulting from electronic devices and power lines [8, 9]. Therefore, a more realistic noise model might be an additive mixture of the Gaussian thermal noise and a non-Gaussian impulsive noise. A popular model, referred to as the class-A noise model, closely fits a variety of non-Gaussian noises and is an analytically tractable model of Gaussian/non-Gaussian noises [10].

In this paper, the problem of turbo equalisation and channel estimation in the presence of class-A impulsive noise is addressed. A frequency selective time varying fading channel is considered, specifically, the channel model described in [11, 12 and 13 (p. 383)] is used. In this model, the time varying channel taps are modelled by a finite linear...
combination of complex exponentials. The basic expansion approach of [11, 12] is used and the time varying channel coefficients are expanded into a set of basic sequences and expansion parameters. These basic sequences are assumed to be known while the expansion parameters are unknown and need to be identified. Direct application of ML approach to estimate these parameters leads to a complex and unpractical estimator. This is due to the unavailability of an efficient way to perform the maximisation of likelihood function. A less complex algorithm is proposed which iteratively estimates the channel parameters. The iterative algorithm is based on the EM algorithm and uses the steepest descent algorithm. The SOVA algorithm is derived for class-A impulsive noise and its soft output is utilised by the proposed algorithm. In class-A impulsive noise, the proposed algorithm exhibits significant performance gain over the algorithms which are efficient in Gaussian noise.

2 Class-A impulsive noise model and system description

2.1 Class-A impulsive noise model
Class-A impulsive noise model of Middleton is a generalised model of the Gaussian noise combined with a non-Gaussian impulsive noise. The class-A impulsive noise for complex channel has a probability density function (pdf), \( p(n_k) \), given by [9, 10]

\[
p(n_k) = \sum_{m_k=0}^{\infty} z_{m_k} p(n_k|m_k)
\]

(1)

where

\[
p(n_k|m_k) = \frac{1}{2\pi \sigma_m^2} \exp \left( -\frac{|m_k|^2}{2\sigma_m^2} \right)
\]

(2)

The subscript \( k \) represents the time index and the parameter \( z_{m_k} \) is given by

\[
z_{m_k} = A^{m_k} e^{-A}
\]

(3)

In (3) the parameter \( A \) is called the impulsive index: it defines the impulsiveness of the noise. For small values of \( A \), the noise becomes more impulsive and for larger values of \( A \), the statistical characteristics of the class-A impulsive noise approach those of Gaussian noise. The variances \( \sigma_{m_k}^2 \) are related to the physical parameters and are given by

\[
\sigma_{m_k}^2 = \sigma_g^2 \frac{(m_k/A) + \Gamma}{1 + \Gamma}; \quad m_k = 0, 1, 2, \ldots \quad (4)
\]

where the parameter \( \sigma_g^2 \) defines the mean variance of the class-A impulsive noise. The model of the white class-A noise combines the presence of an additive man-made noise component with variance \( \sigma_g^2 \) and a white Gaussian noise component with variance \( \sigma_G^2 \). The parameter \( \Gamma = \sigma_g^2 / \sigma_G^2 \) in (4) is the ratio of the variances of the Gaussian noise component to the non-Gaussian impulsive noise component. In [9] and [10], it is explained that the pdf \( p(n_k|m_k) \) is considered to be Gaussian distributed with variance \( \sigma_{m_k}^2 \) conditioned on the state \( m_k \). The variance \( \sigma_{m_k}^2 \) is determined by the realisation of the random channel state \( m_k \), 0, 1, 2, \ldots using (4). The channel states have a Poisson distribution and a certain state \( m_k = m \) occurs with probability \( P(m) = x_m \), where 0 < \( m < \infty \) and \( P(m) \) is the Poisson distribution. It is noted that the channel state is unknown discrete random variable and its estimate at each symbol interval is required. Since there is an infinite number of channel states, we have to truncate their number to a finite value \( \Xi \). The estimate of the channel state \( \hat{m}_k \) is obtained by choosing the state among the finite number of states \( \Xi \) which maximises the noise probability density function

\[
p(n_k) = \max_{0 \leq m_k \leq \Xi} \left[ e^{-A} A^{m_k} \frac{|m_k|^2}{(m_k)^2 \sigma_m^2} \exp \left( -\frac{|m_k|^2}{2\sigma_m^2} \right) \right]
\]

(5)

This maximisation can be expressed as [9]

\[
p(n_k) = e^{-A} A^{m_k} \frac{|m_k|^2}{(m_k)^2 \sigma_m^2} \exp \left( -\frac{|m_k|^2}{2\sigma_m^2} \right),
\]

(6)

where the limits \( \zeta_{m_k}, 0 \leq \hat{m}_k \leq \Xi - 1 \), are given by \( \zeta_0 = 0 \) and

\[
\zeta_m = \frac{2\sigma_{m-1}^2}{\sigma_m^2} \ln \left( \frac{\sigma_{m-1}^2}{\sigma_m^2} \right), \quad 1 \leq m \leq \Xi - 1
\]

and \( \zeta_{\Xi} = \infty \). It should be mentioned that for \( \Xi = 1 \) and \( \Xi = 2 \), (5) and (6) are equivalent.

2.2 System description
The communication system model of coded data symbols over fading channel is shown in Fig. 1. A block of binary source bits \( d = [d_1, d_2, \ldots, d_K]^T \) of length \( K \) and with bit duration \( T_b \), is encoded by a binary convolution encoder in order to yield the encoded data \( c = [c_1, c_2, \ldots, c_K]^T \) of length \( K \). The data block is formed from the data bits and a known preamble bits \( z = [z_1, z_2, \ldots, z_p]^T \) that are inserted in the beginning of the block. Then, the encoded data are channel interleaved to produce the data block \( x = [x_1, x_2, \ldots, x_L]^T \). The interleaved data is passed to a modulator to generate a BPSK signal \( s(t) \). The transmitted signal is distorted by a time varying frequency selective fading channel. The channel is modelled as a discrete time filter with coefficients \( \{h(k, l)\} \) and the output sequence \( \{y_k\} \) is represented as

\[
y_k = \sum_{l=0}^{K-1} x(k-l) h(k, l) + n(k)
\]

(7)

where \( \{y_k\} \) is the input sequence to the channel (which can take values (+1, -1)), \( L \) is the length of channel memory and \( n(k) \) are i.i.d complex valued zero mean white class-A impulsive noise samples. In this paper, we consider the channel, which their time varying coefficients \( h(k, l) \) can be approximated by a linear combination of a finite number of basis sequences \( b_n(k) \)

\[
h(k, l) = \sum_{n=1}^{N} \theta_{nl} b_n(k)
\]

(8)

---

Fig. 1 Communication system model

---

where \( \theta_d \) are non-random expansion parameters and \( b_k(n) \) are basis sequences. For mobile radio channels, these basis sequences are expressed as \( b_k(n) = \exp(j2\pi f_k n) \), where \( z_{k_0}, n = 1, 2, \ldots, N \), are some known frequencies \([11, 12]\). It is noted that, since the channel and the noise are modelled as complex quantities, a more bandwidth efficient modulation, such as QAM, could also be used.

3 Turbo equalisation and decoding under class-A impulsive noise

3.1 Mathematical formulation

In this Section, the log-likelihood metric of the received signal \( y_k \) is derived. Using (7) and (8), \( y_k \) can be expressed as

\[
y_k = \sum_{i=0}^{L-1} \sum_{n=1}^{N} \theta_{ni} x(k-i) b_k(n) + n(k)
\]

Let us define the following vectors

\[
\Theta = [\theta_{11}, \theta_{22}, \ldots, \theta_{NN}]^T
\]

and

\[
x_k = [x(k) \ x(k-1) \ \cdots \ x(k-L+1)]^T
\]

\[
b_k = [b_1(k) \ b_2(k) \ \cdots \ b_N(k)]^T
\]

where the superscript \( T \) denotes matrix transposition. Let the parameters \( \Theta \) be assembled into the \((N \times L)\) unknown matrix \( \Theta \)

\[
\Theta = [\theta_0] \Theta_1 \cdots \Theta_{L-1}
\]

Note that the matrix \( \Theta \) collects the unknown channel parameters from all paths; hence, it can be referred to as the channel parameters matrix (CPM). Using the above definitions, we can rewrite (9) in the following vector/matrix representation

\[
y_k = \mathbf{b}_k^T \Theta x_k + n_k
\]

Since the observation is assumed to be class-A impulsive noise, the simplified conditional log-likelihood function, \( \log(p(y_k|x_k, \Theta)) \), of the received signal is given by

\[
\log(p(y_k|x_k, \Theta)) = \max_{0 \leq m_k \leq \infty} \left\{ \ln \frac{e^{-A_m}A_{m_k}}{m_k^2\pi\sigma^2_{m_k}} \right\} - \frac{1}{2\sigma^2_{m_k}} |y_k - \mathbf{b}_k^T \Theta x_k|^2
\]

In the next Section, the equaliser and decoder are described under class-A impulsive noise assuming that the CPM \( \Theta \) is known. The estimation of this matrix will be investigated in Section 4.

3.2 SISO equaliser and decoder

The aforementioned turbo equaliser of the fading channel consists of a SISO equaliser, a SISO decoder, and a CPM estimator as shown in Fig. 2. The SOVA algorithm is used for both the SISO equaliser and the SISO decoder. The SISO decoder uses the bits \( \{c_k\} \), whereas the SISO equaliser operates with the bits \( \{x_k\} \). The soft information generated by the SISO decoder under class-A impulsive noise is derived as follows. Let \( s_k = \{s_1, s_2, \ldots, s_L\} \) denotes a Markov state sequence up to time \( t \), which represents a path in the trellis, starting at state \( s_1 \) and terminating at state \( s_L \). There are two paths \((n = 1, 2)\) terminating into each state of the trellis, each one has its own accumulated metric and each one is related to different source bit; i.e., \( d_k = 1 \) and \( d_k = 0 \). The branch metric that is associated with the transition from state \( s_{k-1} \) to state \( s_k \) is derived as follows. Let \( n_k^{(n)}(s_{k-1}, s_k) = \ln p(y_k|c_k^{(n)}) + \ln P(d_k^{(n)}) \). For each state, the VA selects the survivor path with the largest accumulated metric \( M_k^{(n)} \) and discards the other. The accumulated metric, \( M_k^{(n)}(s_k) \), is associated with state \( s_k^{(n)} \) is given by the summation of \( n_k^{(n)}(s_{k-1}, s_k) \) for \( n = 1, 2, \ldots, k \); that is,

\[
M_k^{(n)}(s_k) = \sum_{l=1}^{k} n_k^{(n)}(s_k-l, s_k)
\]

Using (15) and knowing that \( n_k^{(n)} = \pi c_k^{(n)} \), where \( \pi \) is the permutation matrix, the accumulated metric \( M_k^{(n)}(s_k) \) can be written in a recursive form as follows

\[
M_k^{(n)}(s_k) = \sum_{l=1}^{k-1} n_k^{(n)}(s_k-l, s_k) + \max_{0 \leq m_k \leq \infty} \left\{ \ln \left( \frac{e^{-A_m}A_{m_k}}{m_k^2\pi\sigma^2_{m_k}} \right) - \frac{1}{2\sigma^2_{m_k}} |y_k - \mathbf{b}_k^T \Theta x_k|^2 \right\} + \ln(P(d_k^{(n)}))
\]

The soft outputs are generated with a backward recursion carried from time \( k \) to time \( k-\delta \), where \( \delta \) is the depth at which the most paths are merged at \( k-\delta \). Assume that path 1 corresponds to the source bit \( d_k = 1 \) with accumulated metric \( M_k^{(1)} \) and path 2 is associated with the source bit \( d_k = 0 \) with accumulated metric \( M_k^{(2)} \). Also, assume that at time \( j \), \( k - \delta \leq j \leq k \), path 1 is selected (i.e. \( M_k^{(1)} > M_k^{(2)} \)), then the probability that the source bit at time \( j \) takes the value \( +1 \) (i.e. \( d_j = 1 \) given the two merging paths, can be written as \( p_j = \frac{1}{1 + \exp(\beta_j)} \), where \( \beta_j = M_j^{(1)} - M_j^{(2)} \). Therefore, the soft value of this path decision is \( L_r(d_j) = \ln(p_j/1-p_j) = \beta_j \), \( k - \delta \leq j \leq k \). Then using (16) the soft value \( L_r(d_j) \) can be written as \( L_r(d_j) = L_E(d_j) + L_A(d_j) \) where \( L_E(d_j) \) is the extrinsic information which is given by

\[
L_E(d_j) = (M_j^{(1)} - M_j^{(2)}) + \max_{0 \leq m_k \leq \infty} \left\{ \ln \left( \frac{e^{-A_m}A_{m_k}}{m_k^2\pi\sigma^2_{m_k}} \right) - \frac{1}{2\sigma^2_{m_k}} |y_j - \mathbf{b}_k^T \Theta x_k^{(1)}|^2 \right\} - \max_{0 \leq m_k \leq \infty} \left\{ \ln \left( \frac{e^{-A_m}A_{m_k}}{m_k^2\pi\sigma^2_{m_k}} \right) - \frac{1}{2\sigma^2_{m_k}} |y_j - \mathbf{b}_k^T \Theta x_k^{(2)}|^2 \right\}
\]
and $L_d(d_j)$ is the a priori information and it is given by $L_d(d_j) = \ln(P(d_j = 1)/P(d_j = 0))$. Then for time $k$, the function $L_P(d_j)$ at the merge point for $j = k - \delta$ constitutes the soft output for the symbol $d_{k-\delta}$. The information $L_P(d_j)$ is ultimately used to reach the final decoding decisions at the end of the turbo iterations. Similarly, the de-interleaved information $L_{\epsilon}(\epsilon_j)$ can be obtained. This information is subtracted from the a posteriori information of the coded bits $L_P(\epsilon_j)$ to provide the extrinsic information $L_E(\epsilon_j)$. The generated extrinsic information $L_E(\epsilon_j)$ is interleaved and used as a priori input information of the equaliser $L_d(x_j)$ in the next iteration. The SISO equaliser of the iterative receiver, shown in Fig. 2, generates the a posteriori log-likelihood $L_d(x_j)$, $k - \delta \leq j \leq k$, upon receiving the observation sequence $\{y_k\}$ and the a priori log-likelihood $L_{\epsilon}(\epsilon_j)$ provided by the SISO decoder. Since the sequence $\{x_j\}$ is an interleaved version of $\{\epsilon_j\}$, we have $L_P(x_j) = \pi L_P(\epsilon_j)$. Consequently, the a posteriori information $L_P(x_j)$ can be expressed as $L_P(x_j) = L_d(x_j) + L_{\epsilon}(\epsilon_j)$ where $L_E(\epsilon_j)$ is the extrinsic information of the received symbols and it is given by

$$L_E(\epsilon_j) = (M^{(1)}_{j-1} - M^{(2)}_{j-1})$$

$$+ \max_{0 \leq m_k \leq 2} \left\{ \ln \frac{e^{-\frac{d_m^4}{m_k^2 + 2\sigma_n^2}}}{m_k^2 + 2\sigma_n^2} \right\} \left\{ y_j - b_j^T \Theta x_j^{(1)} \right\}^2$$

$$- \max_{0 \leq m_k \leq 2} \left\{ \ln \frac{e^{-\frac{d_m^4}{m_k^2 + 2\sigma_n^2}}}{m_k^2 + 2\sigma_n^2} \right\} \left\{ y_j - b_j^T \Theta x_j^{(2)} \right\}^2$$

(18)

and $L_{\epsilon}(\epsilon_j)$ is the a priori information given by $L_{\epsilon}(\epsilon_j) = \ln(P(x_j = 1)/P(x_j = -1))$. Note that, the a priori log-likelihood $L_{\epsilon}(\epsilon_j)$ is set to zero in the first turbo iteration since the transmitted bits are assumed to be equally likely. In general, several iterations (two to three) can be performed in order to decrease the bit error probability. Note that the a posteriori log-likelihoods $L_P(d_j), L_P(x_j)$ and consequently $L_P(\epsilon_j)$ depend on the impulsive index $A$ which implies that the performance of the turbo equaliser depends also on this parameter. This dependence is studied in the simulation section.

4 Channel estimation

In turbo equalisation scheme, the iterations between the SISO equaliser and the SISO decoder require a CPM estimate $\Theta$. The maximum likelihood (ML) approach can be used for this estimation. For $K$ observed symbols, it requires joint maximisation of the log-likelihood function (LLF), $\Delta(\Theta) = \sum_{k=1}^{K} \ln(p(y_k|x_k, \Theta))$, with respect to $\Theta$ and $\{x_k\}$. The optimal solution to this maximisation problem requires an exhaustive search over all possible values of $\Theta$ and $\{x_k\}$ which is prohibitively complex and impractical. A more practical solution is to use a known training sequence and adaptive channel estimators, e.g., conventional minimum square (LMS) algorithm, to estimate $\Theta$. The estimation of $\Theta$ using the conventional LMS algorithm is given by

$$\Theta_{k+1}^{(i)} = \Theta_{k}^{(i)} + \eta \cdot (y_k - b_j^T \Theta_{k}^{(i)} x_k) x_k^T$$

(19)

where $\mu$ is the step size parameter and $i$ is the iteration number. Unfortunately, the performance of the LMS algorithm becomes unstable for strong noise impulsive samples [14]. This is in contrast to the white Gaussian noise case in which the LMS algorithm is efficient and has a good performance. Also it is reported in [9] that, the algorithms which have been derived based on an MMSE criterion are not suited for non-Gaussian noise.

We derive an iterative estimator which is stable in class-A impulsive noise and avoids the above-mentioned difficulties. The EM algorithm in collaboration with the steepest descent approach can be used to derive the algorithm for estimation of $\Theta$. The derivation of the EM algorithm relies on the concept of hypothetical, so-called complete unobservable data, which in our problem are chosen to be $\{y_k, x_k\}$. The complete data would ease the estimation of $\Theta$, if they could be observed. The observed symbols $y_k$ are referred to as the incomplete data within the EM framework. Since complete data is not observable, at the $i$th iteration, the EM algorithm computes in a first step, called the expectation step (E-step), the estimate $Q(\Theta|\Theta^{(i)}) = \sum_{k=1}^{K} \mathbb{E} \left[ \ln(p(y_k|x_k, \Theta)) \right| y_k, \Theta^{(i)}]$ (20)

where $\mathbb{E}(\cdot)$ is the conditional expectation given the observation $y_k$ and assumes that $\Theta$ equals its estimate calculated at iteration $i$. The log-likelihood function can be written as

$$\log[p(y_k|x_k, \Theta)] = \max_{0 \leq m_k \leq 2} \left\{ \ln \frac{e^{-\frac{d_m^4}{m_k^2 + 2\sigma_n^2}}}{m_k^2 + 2\sigma_n^2} \right\} \left\{ y_k - b_j^T \Theta x_k \right\}^2$$

which is simplified to

$$\log[p(y_k|x_k, \Theta)] = \ln \left( \frac{e^{-\frac{d_m^4}{m_k^2 + 2\sigma_n^2}}}{m_k^2 + 2\sigma_n^2} \right)$$

$$- \frac{1}{2\sigma_n^2} \left( y_k - b_j^T \Theta x_k \right)^2$$

(21)

where $m_k$, $0 \leq m_k \leq 2 - 1$, is chosen to satisfy the limits $\sqrt{2\sigma_n} \leq \left( y_k \right)^2 + b_j^T \Theta x_k x_k^T b_j^T$. Then for time $k$, the term $b_j^T \Theta x_k$ becomes

$$= \mathbb{E} \left[ y_k^T b_j^T \Theta x_k \right] = b_j^T \Theta x_k + b_j^T \Theta x_k b_j^T$$

$$- 2 \mathbb{E} \left[ y_k^T b_j^T \Theta x_k \right]$$

(22)

Taking the conditional expectation of (22), one obtains

$$Q(\Theta|\Theta^{(i)}) = \sum_{k=1}^{K} \left\{ \ln \left( \frac{e^{-\frac{d_m^4}{m_k^2 + 2\sigma_n^2}}}{m_k^2 + 2\sigma_n^2} \right) \right\}$$

$$- \frac{1}{2\sigma_n^2} \left( y_k^T b_j^T \Theta x_k \right)$$

$$- 2 \mathbb{E} \left[ y_k^T b_j^T \Theta x_k \right]$$

(23)
where $\mathbf{H}_K = E\{(\mathbf{x}_k - \mathbf{f}_k^{(i)})(\mathbf{x}_k - \mathbf{f}_k^{(i)})^T | y_k, \Theta^{(i)}\}$ is the input covariance matrix. Finally, the maximisation step of the EM algorithm is then given by
\[
\Theta^{(i+1)} = \arg\max_{\Theta} (Q(\Theta | \Theta^{(i)}))
\]
\[
= \arg\min_{\Theta} \left( -\sum_{k=1}^{K} E[\log(p(y_k | x_k, \Theta)) | y_k, \Theta^{(i)}] \right)
\]  
(24b)

The minimisation of (24b) with respect to $\Theta$ can be performed iteratively by applying the steepest descent algorithm
\[
\Theta^{(i+1)} = \Theta^{(i)} + \mu \frac{\partial Q(\Theta | \Theta^{(i)})}{\partial \Theta}_{\Theta=\Theta^{(i)}}
\]
(25)
where $\mu$ is the step size parameter which is chosen as a compromise between the speed response and the stability of the algorithm. Fast convergence requires a larger step size but at the price of a larger estimation error. To decrease the estimation error, a smaller step size is used but this has the effect of slowing down the rate of convergence. Then, normalising the gradient by $1/2\sigma_n^2$ and using (23) and (25), the resulting iterative algorithm for updating $\Theta^{(i+1)}$ at the $i+1$ iteration is given by
\[
\Theta^{(i+1)} = \Theta^{(i)} + \mu v_{\Theta} \sum_{k=1}^{K} \{ 2b_1 b_k^{T} \Theta^{(i)} (\hat{H}_k^{(i)} + \mathbf{f}_k^{(i)} \mathbf{f}_k^{(i)T}) \\
-2\text{Re}(y_k^* b_k \mathbf{f}_k^{(i)T}) \}
\]
(26)
where $v_{\Theta} = 4/\sigma_n^2$, $0 < \tilde{m} \leq \Xi - 1$. The evaluation of (26) requires determination of $\mathbf{f}_k^{(i)}$ and $\hat{H}_k^{(i)}$ which can be obtained using the output of the SOVA algorithm. The mean vector $\mathbf{f}_k^{(i)}$ is obtained from the feedback information obtained from the SISO decoder at iteration $i$, which is the \textit{a priori} information $L_{x_k}^{(i)}(x_k)$ (see Fig. 2).
\[
\mathbf{f}_k^{(i)} = E\{x_k | y_k, \Theta^{(i)}\}
\]
\[
= \sum_{x_k \in \{-1, 1\}} x_k P(x_k = x) = \tanh[L_{x_k}^{(i)}(x_k)]/2
\]
(27)
The covariance matrix $\hat{H}_k^{(i)}$ is obtained as
\[
\hat{H}_k^{(i)} = \sum_{x_k \in \{-1, 1\}} |x_k - \mathbf{f}_k^{(i)}|^2 P(x_k = x) = \mathbf{1}_{L \times L} - \mathbf{f}_k^{(i)} \mathbf{f}_k^{(i)T}
\]
(28)

where $\mathbf{1}_{L \times L}$ denotes an $L \times L$ matrix where all elements are ones. In the beginning, the training sequence $z$ is used to initialise the iterative estimation process and then the soft information $L_z(x_k)$ is used to improve the estimation in the next iteration. At each turbo iteration, the iterative algorithm calculates $\mathbf{f}_k^{(i)}$ and $\hat{H}_k^{(i)}$ using (27) and (28) respectively and then uses (26) to estimate the CPM at the next iteration. Since the true CPM is time invariant, the task of the estimation algorithm in (26) is to converge to the true CPM rapidly as opposed to tracking it. The estimation algorithm provides this convergence in an iterative manner. It requires two to three iterations to converge. This algorithm is well suited to turbo receiver. This is because the turbo receiver requires also two to three iterations to provide good performance. Therefore, the number of iterations of the estimation algorithm is set equal to the number of iterations of turbo receiver. This leads to the advantage of having not only a small but also fixed number of iterations for the estimation algorithm.

It is noted that in order to properly use the steepest descent algorithm, the cost surface has to be convex. Otherwise, the algorithm may converge to some other local minima. In the following, we study the convexity of the surface considered in our problem. As mentioned before, the steepest descent approach is used to maximise the auxiliary function $Q(\Theta | \Theta^{(i)})$ or to minimise its negative. Then, the cost surface, $J(\Theta)$, is defined as the negative of the auxiliary function. Since the auxiliary function is the mean of the LLF given the observation and the CPM estimate at iteration $i$ and also knowing that if $-Q(\Theta | \Theta^{(i)}) \leq -Q(\Theta | \Theta)$ (which is guaranteed by the M-step), then $-\Delta(\Theta^{(i)}) \leq -\Delta(\Theta)$ [15] (which indicates that the EM algorithm decreases the LLF at each iteration).

Then, instead of using $-Q(\Theta | \Theta^{(i)})$ as the cost surface, we can use the negative LLF of the received data given the input data $x_k$ and the CPM $\Theta$. That is
\[
J(\Theta) = -\sum_{k=1}^{K} \log(p(y_k | x_k, \Theta))
\]
(29)
It is clear that $J(\Theta)$ is a scalar-valued quadratic function of $\Theta$ and therefore it is a paraboloid in shape and then it is convex.

5 Simulation and results

In this Section, the performance of the proposed channel estimation and the turbo receiver is evaluated for the mobile radio-fading channel defined in (8). The parameters of the simulation are as follows. The transmitted signal is assumed to be BPSK modulation with a rectangular pulse shape employing a recursive systematic convolutional code with a rate 1/2 and a generator function $g = [111, 101]$. The channel has two time varying taps, each one is a linear combination of three basis sequences denoted by $b_1(k) = 1$, $b_2(k) = 30 \exp\{j nk/60\}$, and $b_3(k) = 30 \exp\{j nk/100\}$ [11,12]. The values of the expansion parameters $\theta_{1L}$ are $\theta_0 = \theta_{11} = 1, \theta_{20} = j, \theta_{21} = 0.5, \theta_{30} = 2$, and $\theta_{31} = -j$. The interleaver is generated randomly for each block of data. The block length is 800 bits and the simulation is performed for 50 blocks of random bits so the resulting number of processed bits is 40,000. Unless stated, the length of preamble is 25 bits. An impulsive white class-A noise with $\Gamma = 1$ is simulated and added to the signal at the input of the receiver. The value of the step size parameter of the channel estimator is $\mu = 0.0001$. The signal-to-noise ratio is defined as $\text{SNR} = 10 \log(E_s/N_0)$ where $N_0 = 2\sigma_n^2$ is the impulsive noise power spectral density and $E_s$ is the energy per bit.

The performance of the turbo receiver under class-A impulsive noise for $\Delta = 0.01, 0.1$, and 1 is shown in Fig. 3. The figure shows that, for a given iteration, the performance of the turbo receiver is improved as the value of $\Delta$ increases. For example, at the third iteration and at $E_s/N_0 = 2$ dB, the BER in the case of $\Delta = 1$ is $10^{-4}$ and in case of $\Delta = 0.01$ is around $10^{-2}$. This is because at $\Delta = 0.01$, the impulsiveness of the noise is stronger than at $\Delta = 1$ which causes performance degradation. Also, as $\Delta$ increases the statistical characteristics of the impulsive noise approach those of the Gaussian noise which has better performance.

The performance comparison among the turbo receiver equipped with the proposed channel estimator (proposed
receiver), the turbo receiver equipped with LMS estimator (turbo/LMS), the turbo receiver with known channel parameters (reference receiver) and the per-survivor processing (PSP) receiver evaluate the bit error rate (BER) as a function of $E_b/N_0$. In the reference receiver, the channel is assumed to be known, therefore this receiver represents unrealistic case and its performance can be considered as a lower bound for comparison purpose of other receivers. In the PSP receiver, the channel estimation is obtained using the data sequence associated with each survivor path in the branch metric calculation. Therefore, for each survivor, independent channel parameters are updated using LMS or RLS algorithms. We call the PSP receiver which uses the LMS algorithm for channel estimation as PSP/LMS and the PSP receiver which uses the RLS algorithm as PSP/RLS. In simulations, for LMS, the value of the step size parameter is $\mu = 0.001$ and for RLS, the value of the forgetting factor is $\lambda = 1$. In the beginning, the known preamble bits are used to obtain an initial estimate of $\Theta$. The results of comparison of the above mentioned receivers are shown in Figs. 4 and 5 for impulsive index $A = 0.1$. Figure 4 illustrates the performance of the turbo receivers at the second iteration, while Fig. 5 illustrates the performance at the third iteration. These figures show that there is a performance loss between the proposed receiver and the reference receiver. The reason for this performance loss is the CPM estimation error which is high at low values of $E_b/N_0$. At these values, the noise is dominant and affects the estimation greatly. This estimation error introduces an error in the metrics calculations which causes degradation in the receiver performance. In the reference receiver, this error does not exist and the performance of the receiver is affected only by the SNR. As $E_b/N_0$ increases, the estimation error decreases and then the performance loss decreases. These figures also show that, at the third iteration (Fig. 5), the gap in performance between the proposed receiver and the reference receiver is less than this gap at the second iteration (Fig. 4). This is because the accuracy of the soft information $L_s(x_k)$ at the third iteration is greater than that one at the second iteration and this is reflects the benefit of the turbo receiver. Figures 4 and 5 also show that the proposed receiver offers significant performance gain over the turbo/LMS receiver.

The value of this gain at $BER = 10^{-4}$ is around 1.7 dB at the second iteration of the turbo receiver and 1.2 dB at the third iteration. That is, the gap in performance between the proposed receiver and the turbo/LMS receiver increases as the number of iteration increases. The reason for this performance gain is that the LMS algorithm is inadequate for class-A impulsive noise. It is sensitive to strong impulsive samples. From Figs. 4 and 5, we see that the performance of the turbo receiver, whether it is equipped with the LMS estimator or the proposed estimator, significantly outperforms the performance of the PSP/RLS and PSP/LMS receivers which again reflects the advantage of the turbo receiver. The PSP/RLS has better performance than the

---

**Fig. 3** Performance of the turbo receiver under class-A impulsive noise for different values of impulse index $A$

**Fig. 4** Performance comparison between different turbo receivers at the 2nd iteration and PSP receivers at $A = 0.1$

**Fig. 5** Performance comparison between different turbo receivers at the 3rd iteration and PSP receivers at $A = 0.1$
PSP/LMS especially at high $E_b/N_0$. This is because the RLS algorithm exhibits better convergence speed than the LMS algorithm at the expense of complexity.

The effect of the known preamble length on the performance of the proposed receiver is studied. The results are shown in Fig. 6. The figure shows that as the preamble length decreases, the performance of the receiver degrades. Short preamble causes large initial uncertainty in the channel estimate and this affects the convergence speed of the estimation algorithm which has significant effect on the receiver performance.

In Fig. 7, the performance of the proposed receiver is evaluated when a frequency offset, $e_f$, is introduced between transmitter and receiver. The figure is plotted for impulsive indices $A = 0.01, 0.1$ and for $E_b/N_0 = -4$ and $5 \text{ dB}$. At $E_b/N_0 = 5 \text{ dB}$, the results show that when the frequency offset is small the receiver is able to detect the signal reliably. When the frequency offset increases, the receiver performance degrades rapidly. The reasons for this degradation are: (a) the increase in frequency offset introduces an error in the branch, accumulated, and the a posteriori metrics calculations and this error affects the decision of the turbo receiver; (b) the increase in frequency offset causes increase in the residual error in estimation of the CPM. This is because the estimation of $\Theta$ depends on the observation which is shifted in frequency, and this increases the mismatch between the estimation of $\Theta$ and the received signal, and accordingly, the BER degrades rapidly. At $E_b/N_0 = -4 \text{ dB}$, the noise dominates the performance and the BER is high. The results illustrated in Fig. 7 also show that at $E_b/N_0 = 5 \text{ dB}$, the performance of the turbo receiver is invariant to frequency offset over a range of frequencies. This range is increased as the impulsive index $A$ becomes larger, for example, at $A = 0.01$, this frequency range is up to $e_fT = 0.0006$ while at $A = 1$, this range is extended to $e_fT = 0.005$. This is because as $A$ increases, the impulsiveness become weaker and then the receiver robustness against frequency offset increases. Thus we can say, at smaller values of $A$, the degradation in performance occurs faster than the degradation at higher values.

### 6 Conclusions

In this paper, a turbo equaliser equipped with channel estimator under class-A impulsive noise environment has been presented. The channel parameters have been estimated continuously using an iterative estimation technique which utilises the soft values from the SOVA. The proposed estimator reduces the complexity of computations resulting from direct application of ML approach. The performance of the proposed receiver has been evaluated for different values of the noise impulsive index $A$ and has been compared with a reference receiver, turbo/LMS receiver, and PSP receiver. It has been shown that the turbo receiver provides significant performance gain over the turbo/LMS receiver and PSP receiver. The effect of the frequency offset on the performance of the turbo receiver has been studied for different values of $A$. It has been shown that, when the frequency offset increases, the receiver performance degrades rapidly at smaller values of $A$.

### 7 Acknowledgments

The authors wish to thank Professor H. Rashvand and the referees for very insightful comments and suggestions.

### 8 References


