Modified LMMSE Turbo Equalization

Sen Jiang, Li Ping, Member, IEEE, Hong Sun, Member, IEEE, and Chi Sing Leung

Abstract—This letter presents a modified linear minimum mean square error (LMMSE) turbo equalization scheme that uses an augmented real matrix representation for quadrature modulation systems. In the proposed scheme, the estimates of the two quadrature components of the transmitted symbol have their individual variances and their covariance is also considered. Hence, the new scheme is able to achieve a noticeable performance improvement while retaining a complexity similar to that of the existing LMMSE turbo equalization.

Index Terms—turbo equalization, intersymbol interference, iterative methods.

I. INTRODUCTION

INSPIRED by turbo codes [1], turbo equalization was introduced to improve the performance of digital communication systems with intersymbol interference (ISI) [2]. A turbo equalization system consists of two soft-in-soft-out elements: a channel equalizer and a channel decoder. These two elements are operated in an iterative manner. However, the computational complexity of the conventional maximum a posteriori (MAP) equalizer is very high [2-4]. A low-cost approach based on a linear minimum mean square error (LMMSE) equalizer is proposed in [5,6]. It is assumed implicitly in [5,6] that the estimates of the two quadrature components of the transmitted symbol have the same conditional variance and zero covariance. This assumption simplifies the derivation but it potentially leads to certain performance degradation.

In this letter, we present a modified LMMSE equalizer based on an augmented real matrix representation. The estimates of the two quadrature components are described by a joint Gaussian distribution and their conditional variances and covariance are estimated individually. Compared with [5,6], the proposed scheme provides a more accurate modeling of the statistics of the two components and can achieve improved performance without increasing receiver cost.

Manuscript received June 20, 2003. The associated editor coordinating the review of this letter and approving it for publication was Prof. K. Narayanan.

II. SYSTEM DESCRIPTION

The communication system considered in this letter is shown in Fig.1. A binary input data sequence is encoded using a convolutional code and permuted into a coded sequences \( \{c_i\} \) by an interleaver \( \pi \). The coded sequence is then partitioned into segments. Each segment is of length \( Q \) and is mapped into a symbol \( x_n \) selected from a quadrature symbol constellation \( A = \{\alpha_1,\alpha_2,\ldots,\alpha_Q\} \). The resultant symbol sequences \( \{x_n\} \) are then transmitted over an ISI channel.

At the receiver side, there are an APP decoder and a LMMSE equalizer operating in an iterative manner described in [6]. Based on the received symbol sequences \( \{y_n\} \) and a priori probabilities \( \{P(x_n = \alpha_i)\} \), the LMMSE equalizer first computes the LMMSE estimates \( \{\hat{x}_n\} \) of the transmitted symbols and then generates the corresponding likelihood values \( \{p(\hat{x}_n | x_n = \alpha_i)\} \). Afterwards, the de-mapper computes the extrinsic log-likelihood ratio (LLR) values \( \{L_x(c_j)\} \) that are de-interleaved and used as the a priori information in the APP decoder [1-5]. The outputs of the APP decoder are the extrinsic LLRs \( \{L_y(c_j)\} \) that are used to generate the a priori information \( \{P(x_n = \alpha_i)\} \) for the next iteration.

A detailed discussion of the overall turbo process can be found in [5-6]. This letter focuses on the LMMSE equalizer.

III. A MODIFIED LMMSE EQUALIZER

A. Existing LMMSE Equalizer—CLMMSE

In [5], the likelihood values \( \{p(\hat{x}_n | x_n = \alpha_i)\} \) are evaluated under the assumption that the distribution of the conditional estimate \( \hat{x}_n \) is complex Gaussian. The real and imaginary parts of \( \hat{x}_n \) are assumed to have the same conditional variance.
and zero covariance. This assumption does not hold in general (see discussion below) which may lead to performance degradation. We denote such an equalizer a complex-number-notation LMMSE (CLMMSE) equalizer. In the following, we will derive an alternative equalizer based on the real number notation. We treat the real and imaginary parts of \( x_n \) as two correlated real variables and evaluate the conditionally joint distribution of their estimates during the equalization process. We call this new equalizer a real-number-notation LMMSE (RLMMSE) equalizer.

**B. Modified LMMSE Equalizer-RLMMSE**

Let \( a, b, \) and \( c \) be complex scalars. Multiplication of two complex scalars \( a = b \cdot c \) can be expressed in a vector form as

\[
\begin{bmatrix}
  \text{Re}(a) \\
  \text{Im}(a)
\end{bmatrix}
= \begin{bmatrix}
  \text{Re}(b) & -\text{Im}(b) \\
  \text{Im}(b) & \text{Re}(b)
\end{bmatrix}
\begin{bmatrix}
  \text{Re}(c) \\
  \text{Im}(c)
\end{bmatrix}
\]

(1)

where \( \text{Re}(\cdot) \) and \( \text{Im}(\cdot) \) denote the real and imaginary parts respectively.

Assume that we know the channel impulse response (CIR) coefficients \( \{ h_n \}_{n=0}^{N-1} \). Let \( \{ w_n \} \) be i.i.d. complex AWGN samples with zero-mean and variance \( \sigma^2 \) per dimension. In general, \( x_n, y_n, h_n \) and \( w_n \) are complex numbers. From (1), the equivalent discrete-time model of an ISI channel with AWGN can be described by an augmented real matrix equation

\[
\begin{bmatrix}
  n \times 1 \\
\end{bmatrix}
= \begin{bmatrix}
  \begin{bmatrix}
    H_{n,0} & H_{n,1} & \cdots & H_{n,N-1}
  \end{bmatrix} \\
  H_{0,0} & H_{0,1} & \cdots & H_{0,N-1}
\end{bmatrix}
\begin{bmatrix}
  n \times 1 \\
\end{bmatrix}
+ \begin{bmatrix}
  n \times 1 \\
\end{bmatrix}
\]

(2)

where \( H \) is a \( 2N \times 2(N + M - 1) \) compound real matrix:

\[
H = \begin{bmatrix}
  H_{M-1} & H_{M-2} & \cdots & H_0 \\
  H_0 & H_{M-2} & \cdots & H_M \\
  \vdots & \ddots & \ddots & \vdots \\
  H_{M-2} & \cdots & H_0 & H_{M-1}
\end{bmatrix}
\]

(3a)

Each \( H_m \) in (3a) is a 2x2 block given by

\[
H_m = \begin{bmatrix}
  \text{Re}(h_m) & -\text{Im}(h_m) \\
  \text{Im}(h_m) & \text{Re}(h_m)
\end{bmatrix}
\]

(3b)

The other compound vectors in (2) are defined by

\[
\begin{bmatrix}
  n \times 1 \\
\end{bmatrix}
= \begin{bmatrix}
  X_{n-M-M+1} & X_{n-M-M+2} & \cdots & X_{n-M+1} \\
  X_{n-M-M+2} & \cdots & \cdots & \cdots \\
  \vdots & \ddots & \ddots & \ddots \\
  X_{n-M+1} & \cdots & \cdots & X_{n+1}
\end{bmatrix}
\]

(4a)

\[
\begin{bmatrix}
  n \times 1 \\
\end{bmatrix}
= \begin{bmatrix}
  Y_{n-M-M+1} & Y_{n-M-M+2} & \cdots & Y_{n-M+1} \\
  Y_{n-M-M+2} & \cdots & \cdots & \cdots \\
  \vdots & \ddots & \ddots & \ddots \\
  Y_{n-M+1} & \cdots & \cdots & Y_{n+1}
\end{bmatrix}
\]

(4b)

\[
\begin{bmatrix}
  n \times 1 \\
\end{bmatrix}
= \begin{bmatrix}
  W_{n-M-M+1} & W_{n-M-M+2} & \cdots & W_{n-M+1} \\
  W_{n-M-M+2} & \cdots & \cdots & \cdots \\
  \vdots & \ddots & \ddots & \ddots \\
  W_{n-M+1} & \cdots & \cdots & W_{n+1}
\end{bmatrix}
\]

(4c)

Each entry of the compound vectors in (4) is a 2x1 block. For example, \( x_n = \begin{bmatrix}
  \text{Re}(x_n) \\
  \text{Im}(x_n)
\end{bmatrix} \), etc. The parameter \( N = N_1 + N_2 + 1 \) is the width of the observation window for estimating \( x_n \), where \( N_1 \) and \( N_2 \) indicate the length of the non-causal and the causal parts, respectively.

Using the real number notation, a signal point \( \alpha_i \) in the constellation can be expressed as a vector \( \alpha_i = \begin{bmatrix}
  \text{Re}(\alpha_i) \\
  \text{Im}(\alpha_i)
\end{bmatrix} \). Following [5], we assume that the constellation is symmetric with \( \sum_{i=0}^{2^n} \alpha_i = \mathbf{0}_{2^n} \) and \( 2^n \sum_{i=0}^{2^n} \alpha_i \alpha_i^\text{T} = \frac{1}{2} \mathbf{I}_2 \), where \( \mathbf{0}_{2^n} \) is an \( l \times m \) zero matrix, and \( \mathbf{I}_2 \) is an \( l \times l \) identity matrix. Let \( \{ P(x_n = \alpha_i) \} \) be a set of \textit{a priori} probabilities of the transmitted symbol \( x_n \), which is initialized with \( \{ P(x_n = \alpha_i) = 2^{-\nu}, \forall i \} \) and updated during the iterative process as described in Section II [5].

The means of \( x_n, y_n \) and \( v_n \) and the covariance matrix of \( x_n \) (denoted by \( V_n = \text{Cov}(x_n, x_n) \)) are given by

\[
E(x_n) = \sum_{\alpha_i} \alpha_i P(x_n = \alpha_i)
\]

(5a)

\[
E(y_n) = \left[ E(x_{n-N_2-M+1})^\text{T}, \cdots, E(x_{n+M})^\text{T} \right]^\text{T}
\]

(5b)

\[
E(v_n) = H^\text{T} E(x_n)
\]

(5c)

\[
V_n = \sum_{\alpha_i} \alpha_i \alpha_i^\text{T} P(x_n = \alpha_i) - E(x_n)E(x_n)^\text{T}.
\]

(5d)

In the above \( V_n \) is a \( 2 \times 2 \) symmetric matrix. The two diagonal entries of \( V_n \) give the variances of \( \text{Re}(x_n) \) and \( \text{Im}(x_n) \) respectively and the two (equal) off-diagonal entries of \( V_n \) give the covariance of \( \text{Re}(x_n) \) and \( \text{Im}(x_n) \). Initially, we assume that there is no \textit{a priori} information, i.e., \( P(x_n = \alpha_i) = 2^{-\nu}, \forall i \) and set \( V_n = \frac{1}{2} \mathbf{I}_2 \), \( \forall n \), in (5d).

Here is the main difference between the RLMMSE equalizer discussed in this letter and the CLMMSE equalizer introduced in [5]. In both approaches, after the first iteration, \( \{ P(x_n = \alpha_i) \} \) are updated using the feedback LLRs from the APP decoder (see Section II). The covariance matrix \( V_n \) for the RLMMSE equalizer may have un-equal diagonal entries and non-zero off-diagonal entries, implying that the variances of \( \text{Re}(x_n) \) and \( \text{Im}(x_n) \) can be different and their covariance can be nonzero. As a comparison, the CLMMSE approach uses only one real number scalar \( \text{Var}(x_n) \) to describe the variance of \( x_n \) (equivalent to introducing a constraint that \( V_n = \frac{1}{2} \text{Var}(x_n) \mathbf{I}_2 \), i.e., \( V_n \) is a scaled unit matrix, in the RLMMSE approach). Clearly, the RLMMSE approach provides a more accurate statistical modeling of \( x_n \) and so it may potentially achieve improved performance, as we will see later.

Based on (2)-(5), we can derive the RLMMSE estimator. The general principle follows the discussion in [5], except that the matrices involved here, such as \( H, X_n, Y_n \) and \( W_n \), are all compound, i.e., their entries are blocks. Due to this, we will omit the derivation details and only list the results below. The so-called “extrinsic” LMMSE estimate \( \hat{x}_n \) of \( x_n \) is given by

\[
\hat{x}_n = F_n^\text{T} (y_n - E(y_n) + SE(x_n))
\]

(6)
where
\[
S = R_{2×2} + M_{2×2} = [I_2, 0_{2×2N_2}]
\]
\[
F_n = \left(\begin{array}{c} \Sigma + S(\frac{1}{2}I_2 - V_n)S^T \end{array}\right)^{-1} S
\]
\[
\Sigma_n = \text{Cov}(y_n, y_n) = \sigma^2 I_2 + HV_nH^T
\]
\[
V_n = \text{Diag}(V_{n-N_2-M+1}, \ldots, V_n, \ldots, V_{n+N_2}).
\]

In (6), \(\hat{x}_n\) is a vector whose two entries are the LMMSE estimators for the real and imaginary parts of \(x_n\), respectively. It can be verified that the computation of the \(a\) \(priori\) information related to \(x_n\), i.e., \(V_n\) and \(E(x_n)\), is cancelled in (6). This indicates that \(\hat{x}_n\) is “extrinsic”, which conforms to the fundamental turbo principle developed in [1].

We assume that the conditional PDFs of \(\hat{x}_n\) (given \(x_n = \alpha_n\)) are joint Gaussian with a 2×1 mean vector
\[
m_{\hat{x}_n} = E(\hat{x}_n | x_n = \alpha_n) = F_n^T S \alpha_n
\]
and a 2×2 covariance matrix
\[
R_{\hat{x}_n} = \text{Cov}(\hat{x}_n, \hat{x}_n | x_n = \alpha_n) = F_n^T (\Sigma_n - S V_n S^T) F_n.
\]

According to the joint Gaussian assumption and (6), the likelihood value of \(\hat{x}_n\) is given by
\[
p(\hat{x}_n | x_n = \alpha_n)
\]
\[
= K_1 \exp \left(-\frac{1}{2}(\hat{x}_n - m_{\hat{x}_n})^T R_{\hat{x}_n}^{-1}(\hat{x}_n - m_{\hat{x}_n}) \right)
\]
\[
= K_2 \exp \left(\alpha_n^T \Gamma \Phi - V_n \right) - I_2 \left(\Gamma + \frac{1}{2} \Phi \alpha_n \right) \right)
\]
where \(K_1, K_2\) are constants for every \(\hat{x}_n\) and
\[
\alpha_n = E(x_n) - \alpha_n
\]
\[
\Gamma = S^T \Sigma_n^{-1} \left(V_n - E(V_n) \right)
\]
\[
\Phi = S^T \Sigma_n^{-1} S.
\]

In (9), the matrix inversion lemma is applied to avoid direct computation of \(R_{\hat{x}_n}^{-1}\) but we have omitted the details. The derivation is very similar to that in [5].

The likelihood values \(p(\hat{x}_n | x_n = \alpha_n)\) constitute the outputs of the RLMMSE equalizer.

C. Calculation of \(\Sigma_n^{-1}\) and Complexity

Directly computing \(\Sigma_n^{-1}\) in (10) has a complexity order of \(N^3\). Alternatively, we can adopt a recursive algorithm [5] to compute \(\Sigma_n^{-1}\) with complexity \(O(N^2)\). In [5], \(\Sigma_n\) is a \(N \times N\) complex number matrix while \(\Sigma_n\) is a \(2N \times 2N\) real number matrix. The complexities involved in processing these two matrices are nearly the same. By similar reasoning, it can be also verified that the RLMMSE and CLMMSE approaches have similar complexity.

IV. SIMULATION RESULTS

We now compare the performance of CLMMSE and RLMMSE approaches for an 8-PSK modulation system. A rate \(R = 1/2\) convolutional code with generator
\[
\left[ D^2 + D + 1 \right]
\]

is used as the channel code. The frame length of data is 2049 bits. The interleaver is generated randomly. We adopt two time-invariant ISI channels as in [7] with
\[
C1: \{h_m \}_{m=0}^{M-1} = \{0.227, 0.46, 0.688, 0.46, 0.227\}
\]
\[
C2: \{h_m \}_{m=0}^{M-1} = \{0.29, 0.5, 0.58, 0.5, 0.29\}.
\]

The width of the observation window \(N\) is set to 15 (\(N_1 = 5, N_2 = 9\)). The same signal mapper as that in [5] is used. Simulation results are shown in Fig. 2. At BER=10^{-5}, the RLMMSE equalizer achieves about 1dB performance gain over the CLMMSE equalizer in both channels.

V. CONCLUSION

A modified LMMSE equalizer for turbo equalization has been derived based on an augmented real matrix representation. It provides a more accurate model for the computation of the fundamental turbo principle. The derivation is very similar to that in [5].

The likelihood values \(p(\hat{x}_n | x_n = \alpha_n)\) constitute the outputs of the RLMMSE equalizer.

C. Calculation of \(\Sigma_n^{-1}\) and Complexity

Directly computing \(\Sigma_n^{-1}\) in (10) has a complexity order of \(N^3\). Alternatively, we can adopt a recursive algorithm [5] to compute \(\Sigma_n^{-1}\) with complexity \(O(N^2)\). In [5], \(\Sigma_n\) is a \(N \times N\) complex number matrix while \(\Sigma_n\) is a \(2N \times 2N\) real number matrix. The complexities involved in processing these two matrices are nearly the same. By similar reasoning, it can be also verified that the RLMMSE and CLMMSE approaches have similar complexity.

IV. SIMULATION RESULTS

We now compare the performance of CLMMSE and RLMMSE approaches for an 8-PSK modulation system. A rate \(R = 1/2\) convolutional code with generator
\[
\left[ D^2 + D + 1 \right]
\]

is used as the channel code. The frame length of data is 2049 bits. The interleaver is generated randomly. We adopt two time-invariant ISI channels as in [7] with
\[
C1: \{h_m \}_{m=0}^{M-1} = \{0.227, 0.46, 0.688, 0.46, 0.227\}
\]
\[
C2: \{h_m \}_{m=0}^{M-1} = \{0.29, 0.5, 0.58, 0.5, 0.29\}.
\]

The width of the observation window \(N\) is set to 15 (\(N_1 = 5, N_2 = 9\)). The same signal mapper as that in [5] is used. Simulation results are shown in Fig. 2. At BER=10^{-5}, the RLMMSE equalizer achieves about 1dB performance gain over the CLMMSE equalizer in both channels.

V. CONCLUSION

A modified LMMSE equalizer for turbo equalization has been derived based on an augmented real matrix representation. It provides a more accurate model for the