EM-Based Joint Channel Estimation and Detection for Frequency Selective Channels Using Gaussian Message Passing

Qinghua Guo, Member, IEEE, and Defeng (David) Huang, Senior Member, IEEE

Abstract—It has been recently shown that the expectation-maximization (EM) algorithm may be viewed as message passing in factor graphs, and in particular, for a linear Gaussian system (with unknown coefficients), the EM algorithm may be purely implemented with Gaussian message passing. In this work, with a Gaussian assumption of the data symbols and a Forney-style factor graph representation for single-carrier transmission over frequency selective channels, a Gaussian message passing EM approach for joint channel estimation and detection is developed. The complexity of the proposed approach grows logarithmically with the length of the observation vector, enabling an efficient handling of (quasi-static and time-varying) frequency selective channels with a large number of channel taps.

Index Terms—Expectation-Maximization, factor graphs, message passing, channel estimation, detection.

I. INTRODUCTION

Joint channel estimation and maximum a posteriori (MAP) data detection has been investigated under the framework of the expectation-maximization (EM) algorithm, e.g., the work in [5]-[8], [15] and the references therein. The complexity of these EM-based approaches in general grows exponentially with the number of channel taps, thereby making them intractable when the number of channel taps is large, e.g., in broadband wireless communications where the inter-symbol interference (ISI) may span hundreds of symbols [1].

Recently, Dauwels et al. have shown that the EM algorithm may be fully described as message passing in factor graphs [11], [12], where the expectation step generates the so-called EM messages to be used in the maximization step. The EM message may be computed using a general local computation rule (see Table I in [12]), and the local expectations involved in the EM message computation may be obtained with the sum-product message passing. With the EM message from the expectation step, the maximization step may be implemented using the max-product message passing. For a linear Gaussian system with unknown coefficients, the EM message turns out to be Gaussian, leading to pure Gaussian message passing implementations of the EM algorithm [12].

EM-based joint channel estimation and MAP detection for single-carrier transmission has been formulated as message passing in factor graphs in [6] and [8]. However, the approach in [8] was designed for flat fading channels only. The approach in [6] handles frequency selective channels, but it involves complexity growing exponentially with the number of channel taps.

In this work, assuming that the data symbols are Gaussian (i.e., only their first and second moments are used as in the linear minimum mean square error (MMSE) detection), we develop a low-complexity Gaussian message passing EM approach for joint channel estimation and detection in single-carrier transmission over frequency selective channels. By using a frequency domain system model and introducing proper approximations in the message computations, the complexity per symbol of the proposed approach grows logarithmically with the length of the observation vector, enabling an efficient handling of frequency selective channels with a large number of channel taps. The proposed approach is then incorporated into turbo equalization [13], [14], for joint channel estimation, detection and decoding, and finally extended to handle time-varying frequency selective channels with the aid of the autoregressive (AR) model.

The contributions of this work can be summarized as follows. 1) By carrying out channel estimation and detection jointly, the proposed low-complexity EM-based approach can significantly outperform alternative approaches (with similar complexity) where channel estimation and detection are implemented separately, e.g., the approaches in [17], [18], and [19] (in which the uncertainty due to the estimate of the symbols is accommodated into the additive noise for channel estimation, and vice versa for symbol estimation). 2) To effectively and efficiently implement the EM approach using Gaussian message passing [10], [12], it is crucial to develop a proper system model for the system graph representation. We find out such a system graph representation, which enables proper approximations that reduce the implementation complexity to logarithmic level. The above makes the proposed approach attractive in broadband wireless communications.

Notations: Bold uppercase (lowercase) letters are used to denote matrices (column vectors). The superscripts ”T”, ”+”, and ”H” denote the transpose, conjugate, and conjugate transpose operations, respectively. $F_P$ denotes a normalized discrete Fourier transform (DFT) matrix with size $P \times P$.

1We note here that the overall process of joint channel estimation, detection and decoding can be described as message passing in factor graphs, e.g., as the approach in [20] for multiple input multiple output systems over time-varying flat fading channels.

Copyright (c) 2011 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

The authors are with the School of EECE, The University of Western Australia, Australia (e-mail: {qinghua.guo, david.huang}@uwa.edu.au). This work was supported by Australian Research Council’s Discovery Project DP1093000.
are as follows (refer to [11] and [12] for details):

\[ x = \sqrt{\frac{1}{2\pi}} e^{-i2\pi mn/P} \]

where \( i = \sqrt{-1} \), \( I_P \) denotes an identity matrix with size \( P \times P \), and \( 0_{M \times N} \) denotes a zero matrix with size \( M \times N \). \( Diag(a) \) denotes a diagonal matrix with the entries of \( a \) on its diagonal.

The real and imaginary parts of a complex variable \( a \) are denoted by \( a^{Re} \) and \( a^{Im} \), respectively.

II. EM USING MESSAGE PASSING

The EM algorithm is an iterative technique for parameter estimation in a probabilistic system with hidden (latent) variables [3]. Let vectors \( r, x \), and \( \theta \) denote observed variables, hidden variables, and the parameters to be estimated, respectively, and use \( f(r, x, \theta) \) to denote either a joint probability density function (PDF) of \( r \) and \( x \) parameterized by \( \theta \) or a joint PDF of \( r, x \), and \( \theta \). The following maximization with respect to \( \theta \)

\[ \hat{\theta} = \arg\max_{\theta} \int_x f(r, x, \theta)dx \]  

(1)

may be intractable. The EM algorithm solves this problem iteratively, and each iteration, e.g., the \( p \)th iteration, involves the following two steps:

- **E-Step:** \( Q(\theta; \hat{\theta}^{p-1}) = E[\log f(r, x, \theta) | r, \hat{\theta}^{p-1}] \)
- **M-Step:** \( \hat{\theta}^p = \arg\max_{\theta} Q(\theta; \hat{\theta}^{p-1}) \)

where \( E[\cdot] \) denotes the expectation operation with respect to \( x \), and \( \hat{\theta}^p \) is the estimate of \( \theta \) in the \( p \)th iteration.

A. Message Passing EM Algorithm

Consider the factorization \( f(r, x, \theta) = f_A(\theta)f_B(r, x, \theta) \), then (1) can be rewritten as

\[ \hat{\theta} = \arg\max_{\theta} f_A(\theta)\int_x f_B(r, x, \theta)dx \]  

(2)

where the two functions \( f_A \) and \( f_B \) in (2) may have nontrivial factorization and can be represented using their corresponding factor graphs. In this correspondence, we employ the Forney style factor graph (FFG) [10]. An example is shown in Fig. 1, where the upper part is the graph of \( f_A \) and the lower part is that of \( f_B \). The details of Fig. 1 will be explained later.

The message passing EM algorithm (for the \( p \)th iteration) are as follows (refer to [11] and [12] for details):

- **E-Step:** With \( \hat{\theta}^{p-1} \) plugged into \( f_B \), perform the sum-product message passing in the factor graph of \( f_B \) (called E-step graph in [8]) to compute the incoming messages for the local nodes associated with the entries of \( \theta \). Then, compute the EM messages for the entries of \( \theta \) with the rule given by Table I in [12].
- **M-Step:** With the EM messages from the E-step, perform the max-product message passing in the factor graph of \( f_A \) (called M-step graph in [8]) to obtain the new estimate \( \hat{\theta}^p \).

B. EM for Linear Gaussian Systems

For a linear Gaussian system (with unknown coefficients), the sum-product messages in the E-step are Gaussian, and can be calculated using the Gaussian message passing technique [10]. The unknown coefficients and the hidden variables in such a linear system give rise to multiplier nodes in the FFG of the system. It has been shown in [12] that, with Gaussian incoming messages, the EM messages out of such multiplier nodes properly grouped with “soft” Gaussian factors are Gaussian too. With the Gaussian EM messages from the E-step, the max-product message passing in the M-step coincides with the sum-product message passing [10], [12], and can also be implemented using the Gaussian message passing technique. The above lead to pure Gaussian message passing algorithms for EM.

III. GAUSSIAN MESSAGE PASSING ALGORITHM FOR JOINT CHANNEL ESTIMATION AND DETECTION

A. System Model

We employ a single-carrier transmitter where the data bits are encoded into a codeword, and the coded bits are then interleaved and mapped to a symbol sequence \( s \) using QPSK with gray mapping. To avoid the spectral overhead due to training sequences, we employ the superimposed training, \(^2\) i.e., the transmitted signal

\[ x = s + p \]  

(3)

where \( p \) is the training sequence.

Assume that the signal block \( x \) with length \( J \) is transmitted over an \( L \)-tap quasi-static frequency selective channel denoted by \( h = [h_0, h_1, ..., h_{L-1}]^T \). The received signal \( y \) (with length \( N = J + L - 1 \)) can be represented as

\[ y = h \ast x + n \]  

(4)

where \( \ast \) denotes the linear convolution, and \( n \) denotes an additive white Gaussian noise with zero mean and covariance matrix \( \sigma^2 I \).

By defining two length-\( N \) vectors

\[ \tilde{x} = [x^T, 0_{1 \times (N-L)}]^T \]  

(5)

and \( \tilde{h} = [h^T, 0_{1 \times (N-L)}]^T = Sh \) with \( S = [I_L, 0_{L \times (N-L)}]^T \), the linear convolution (4) can be rewritten as \( y = \tilde{h} \ast \tilde{x} + n \) where \( \ast \) denotes the cyclic convolution, leading to the following frequency domain model

\[ r = t \odot c + w \]  

(6)

where \( \odot \) denotes the elementwise product of two vectors, \( r = F_N y \), \( w = F_N n \) (it has the same distribution as \( n \)),

\[ t = F_N \tilde{x} \]  

(7)

and the frequency domain channel response

\[ c = \sqrt{N} F_N \tilde{h} = \sqrt{N} Fh \]  

(8)

with \( F = F_N S \).

The length of the frequency domain channel response \( c \) in model (6) is \( N \), but the actual degree of freedom of the channel is \( L \) (since the length of \( \tilde{h} \) is \( L \)). This means that there are some

\(^2\) The approach in this work can be easily extended for the time-multiplexed training since the transmitted signal stream for time-multiplexed training can be regarded as the superposition of a training sequence and a data sequence by properly inserting some zero symbols to the training symbols and the data symbols, respectively.
constraints among the entries of $c$, which should be properly exploited [19].

Assume that $N = KL$ ($K$ is a positive integer and this can be guaranteed by system design). Define an $L \times L$ diagonal matrix $D = \text{Diag}([1, e^{2\pi i/N}, \ldots, e^{2\pi i(L-1)/N}])$ and $\bar{F}_j^T$ as the $j$th row of matrix $\bar{F}$. It is not hard to verify that

$$\sqrt{L}F_L(D^H)^k = \sqrt{N} \begin{bmatrix} f_{j}^{T} & 0 \cdots & 0 \\ f_{j}^{T} & 0 \cdots & 0 \\ \vdots & \vdots & \vdots \\ f_{(L-1)}^{T} & 0 \cdots & 0 \end{bmatrix}. \tag{9}$$

We group the elements of frequency domain channel response $c$ into $K$ length-$L$ subvectors

$$c_k = [c_{0 \times K+k}, c_{1 \times K+k}, \ldots, c_{(L-1) \times K+k}]^T, k = 0, 1, \ldots, K-1$$  \tag{10}

where $c_j$ is the $j$th entry in $c$. From (8), (9) and (10), we have

$$c_k = \sqrt{L}F_L(D^H)^k h. \tag{11}$$

From (11), we have

$$c_{k-1} = F_LDF_L^H c_k, \quad k = K - 1, K - 2, \ldots, 1. \tag{12}$$

The above recursive relationship among $\{c_k\}$ restricts the degree of freedom of $c$ to be $L$.

With (8) and (9), we can rewrite the frequency domain model (6) into the following $K$ submodels

$$r_k = t_k \odot c_k + w_k, \quad k = 0, 1, \ldots, K-1 \tag{13}$$

where

$$r_k = [r_{0 \times K+k}, r_{1 \times K+k}, \ldots, r_{(L-1) \times K+k}]^T, \quad t_k = [t_{0 \times K+k}, t_{1 \times K+k}, \ldots, t_{(L-1) \times K+k}]^T$$

and

$$w_k = [w_{0 \times K+k}, w_{1 \times K+k}, \ldots, w_{(L-1) \times K+k}]^T (r_j, w_j \text{ and } t_j \text{ denote the } j \text{th entry in } r, w \text{ and } t \text{, respectively.).}$$

B. Joint Channel Estimation and Detection with Gaussian Message Passing EM

The FFG of the system model described by (12), (13) and (7) is shown in Fig. 1, where we partition the overall factor graph into two subgraphs: E-step factor graph and M-step factor graph. Note that the variables $c_k^e$ and $c_k^p$ (the “clones” of $c_k$) are introduced since no variable appears in more than two factors in an FFG [10]. As in [10] and [12], we use arrows to denote the directions of message passing, i.e., for a directed edge representing variable $a$, $\overrightarrow{\bar{m}_a}$ and $\overrightarrow{\bar{V}_a}$ denote the mean vector and covariance matrix of the Gaussian message along the direction of $a$, and $\overleftarrow{m_a}$ and $\overleftarrow{V_a}$ denote those along the opposite direction of $a$. $m_a$ and $V_a$ denote the marginal mean vector and marginal covariance matrix of $a$.

Treat the QPSK symbols (entries of $s$) to be Gaussian, and assume that the variance of a QPSK symbol is $\alpha$, and that its mean is zero. Then, from (3) and (5), we have

$$\bar{m}_k = [p^T, 0_{1 \times (L-1)}]^T \text{ and } \bar{V}_k = \text{Diag}([\alpha 1_{1 \times J}, 0_{1 \times (L-1)}])$$

where $1_{1 \times J}$ is an all-one row vector with length $J$.

The implementations of the E-step and M-step (for the $p$th EM iteration) are detailed in the following.

E-Step Implementation

First, we need to compute the incoming message $\bar{m}_{tk}$ and $\bar{V}_{tk}$ for each composite node (a multiplier node grouped with its adjacent “soft” node) indicated by the dashed boxes in Fig. 1. This involves the messages from $\{k', k'\neq k\}$ and $\hat{x}$, and the computational complexity is high ($O(N^3)$) due to non-trivial matrix operations. To simplify the computations, we introduce the following approximation

$$\bar{V}_{tk} \approx \alpha I_N. \tag{15}$$

Using the fact that $F_N$ is a unitary matrix and with the approximation in (15), it can be shown that the messages from $\{k', k' \neq k\}$ make no contributions to $\bar{m}_{tk}$ and $\bar{V}_{tk}$, i.e., $\bar{m}_{tk}$ and $\bar{V}_{tk}$ only depend on the message from $\hat{x}$, and can be computed as follows. From (7) and (15), we have

$$\bar{m}_t = F_N \bar{m}_{\hat{x}} \tag{16}$$
$$\bar{V}_t = F_N \bar{V}_{\hat{x}} F_N^H = \alpha I_N. \tag{17}$$

Let $\bar{m}_{t_j}$ denote the $j$th entry in $\bar{m}_t$. Then, from (14), the incoming messages of $t_k$

$$\bar{m}_{tk} = [\bar{m}_{t_0 \times K+k}, \bar{m}_{t_1 \times K+k}, \ldots, \bar{m}_{t_{(L-1) \times K+k}}]^T \tag{18}$$
$$\bar{V}_{tk} = \alpha I_L. \tag{19}$$

With the incoming messages $\{\bar{m}_{tk}, \bar{V}_{tk}\}$ and $\{\hat{c}_k^{p-1}\}$ (the estimates of $\{c_k\}$ in the $(p-1)$th EM iteration), we then compute the outgoing EM messages for $\{c_k\}$ of the composite nodes in the dashed boxes. According to rules III.5 and III.6 in Table III and rules IV.1 and IV.7 in Table IV in [12] (these rules are for real variables, but can be extended for complex variables), the EM message for $c_k$ can be computed as

$$\bar{V}_{c_k} = \sigma^2 (V_{t_k} + \text{Diag}(m_{t_k}^e \odot m_{t_k}))^{-1} \tag{20}$$
$$\bar{m}_{c_k} = \frac{1}{\sigma^2} \overrightarrow{\bar{V}}_{c_k}(m_{t_k}^e \odot r_k) \tag{21}$$

with

$$V_{t_k} = \left( \bar{V}_{t_k}^{-1} + \frac{1}{\sigma^2} \text{Diag}(\hat{c}_k^{p-1} \odot \hat{c}_k^{p-1}) \right)^{-1} \tag{22}$$
$$m_{t_k} = V_{t_k} \left( \bar{V}_{t_k}^{-1} \bar{m}_{t_k} + \frac{1}{\sigma^2} (\hat{c}_k^{p-1} \odot r_k) \right) \tag{23}$$

where $\bar{V}_{t_k}^{-1} = \frac{1}{\alpha} I_L$ (see (19)).

From the above, we can see that the approximation in (15) leads to diagonal matrices $\{\bar{V}_{tk}\}$, which make the calculations of (20)-(23) trivial (since all the matrices are diagonal). Moreover, the resultant diagonal matrices $\{\bar{V}_{c_k}\}$ also simplify the message calculations in the following M-step. Here we note that, $\{\bar{m}_{tk}, \bar{V}_{tk}\}$ do not change during the EM iteration, i.e., (16)-(19) only need to be calculated once.

M-Step Implementation

In this step, we need to run Gaussian message passing with the message computation rules for the equality constraint and the multiplication nodes in the M-step factor graph in Fig. 1 to
obtain the new estimates for \( \{ c_k \} \). The exact calculations incur high complexity due to the matrix inversions and products involved. As in [19], to reduce the complexity, the following approximation is adopted in the forward covariance matrix updating for the multiplication node

\[
\overline{\mathbf{V}}_{c_{k-1}} = \mathbf{F}_L \mathbf{D}^H_L \overline{\mathbf{V}}_{c_{k-1}} \mathbf{F}_L \mathbf{D}^H_L \approx \gamma_k \mathbf{I}_L
\]

(24)

where \( \gamma_k = L^{-1} \text{trace}(\overline{\mathbf{V}}_{c_k'}) \), and \( \text{trace}(\cdot) \) denotes the trace operation. For its forward mean vector updating, the exact computation rule is adopted, i.e.,

\[
\overline{\mathbf{m}}_{c_{k-1}} = \mathbf{F}_L \mathbf{D}^H_L \overline{\mathbf{m}}_{c_k'}.
\]

(25)

Note that (25) can be calculated efficiently using one \( L \)-point FFT and one \( L \)-point IFFT,\(^3\) and the calculation of (24) is trivial. Approximation in (24) simplifies the following message computation for the equality constraints in the FFG with rules II.1 and II.3 in [10] (noting that matrix \( \overline{\mathbf{V}}_{c_{k-1}} \) is the output from the E-step, and is diagonal)

\[
\overline{\mathbf{V}}_{c_{k-1}'} = (\overline{\mathbf{V}}_{c_{k-1}}^{-1} + \overline{\mathbf{V}}_{c_{k-1}} - 1)^{-1}
\]

(26)

\[
\overline{\mathbf{m}}_{c_{k-1}'} = \overline{\mathbf{V}}_{c_{k-1}'}^{-1} \left( \overline{\mathbf{V}}_{c_{k-1}}^{-1} \overline{\mathbf{m}}_{c_{k-1}} + \overline{\mathbf{V}}_{c_{k-1}}^{-1} \overline{\mathbf{m}}_{c_{k-1}} \right).
\]

(27)

Through a forward recursion,\(^4\) we can obtain \( \overline{\mathbf{m}}_{c_k'} \), which is actually the marginal mean vector of \( \mathbf{c}_k' \), i.e., \( \mathbf{m}_{c_k'} = \overline{\mathbf{m}}_{c_k'} \), since the edge representing \( \mathbf{c}_k' \) is located at the rightmost end of the M-step factor graph. According to the marginal computation rules for the equality constraint and the multiplication nodes (rules II.5 and II.6 in Table 2 and rules III.3 and III.4 in Table 3 in [10]), we have \( \mathbf{m}_{c_0} = \mathbf{m}_{c_0}' \), and the marginal mean vectors of \( \{ \mathbf{c}_k, k \neq 0 \} \) can be directly calculated from \( \mathbf{m}_{c_0} \) through the following recursion (noting that the backward message passing and message combining are not needed)

\[
\mathbf{m}_{c_k} = \mathbf{F}_L \mathbf{D}^H_L \mathbf{D}^H_L \mathbf{m}_{c_{k-1}}, \quad k = 1, 2, \ldots, K - 1.
\]

(28)

The new estimates \( \mathbf{c}_k'' \), \( k = 0, 1, \ldots, K - 1 \), will be employed by the E-step in the next EM iteration.

In the above approach, the marginal messages about \( \{ t_k \} \) (see (22) and (23)) are computed, which consist of the contributions of the frequency domain counterparts of the symbol sequence \( s \) and the known training sequence. In the final EM iteration, the estimate of the symbol sequence \( s \) can be obtained from \( \{ \mathbf{m}_t \} \) and \( \{ \mathbf{V}_t \} \). From \( \{ \mathbf{m}_t \} \) and \( \{ \mathbf{V}_t \} \), we have the mean vector \( \mathbf{m}_t \) and covariance matrix \( \mathbf{V}_t \) (it is a diagonal matrix since \{\( \mathbf{V}_t \)\} are diagonal). Then the marginal mean vector and covariance matrix of \( \bar{x} \) can be calculated as

\[
\mathbf{m}_{\bar{x}} = \mathbf{F}_N^H \mathbf{m}_t
\]

(29)

\[
\mathbf{V}_{\bar{x}} = \mathbf{F}_N^H \mathbf{V}_t \mathbf{F}_N.
\]

(30)

From (29), (30) and (3), the marginal mean of \( s_j \) (i.e., \( \mathbf{m}_{s_j} = \mathbf{m}_{x_j} - \mathbf{p}_j \)), where \( \mathbf{p}_j \) is the \( j \)-th entry of \( \mathbf{p} \), and the marginal covariances of \( \{ s_j \} \), \( \mathbf{V}_{s_0} = \mathbf{V}_{s_1} = \cdots = \mathbf{V}_{s_{j-1}} = \frac{1}{N} \sum_k \mathbf{V}_{t_k} \) (note that the calculation of (30) is not needed), where \( \mathbf{V}_{t_k} \) is the \( k \)-th diagonal entry of \( \mathbf{V}_t \). Assume that, for each QPSK symbol \( s_j, \mathbf{s}_j^{Re} \) and \( \mathbf{s}_j^{Im} \) are independent of each other and have the same variance \( \mathbf{V}_{s_j} / 2 \), so the log-likelihood ratios (LLRs) of the two bits in \( s_j \) can be calculated as [10]

\[
L(s_j^{Re}) = 2 \sqrt{2 \mathbf{m}_{s_j}^{Re} / \mathbf{V}_{s_j} \mathbf{m}_{s_j}^{Im} / \mathbf{V}_{s_j}}
\]

which may be used for soft-input decoding.

By checking the message computations in the EM algorithm described above, we can find that all the calculations involved either are trivial (e.g., (20)-(23), (26) and (27)), or can be efficiently implemented with (I)FFT(s) (e.g., (16), (25), (28) and (29)). It can be shown that the complexity of this approach is \( O(P \log L) + O(\log N) \) per symbol, where \( P \) is the number of EM iterations. In the approaches [17]-[19] where the channel estimation and detection are implemented separately, the complexity of detection is \( O(\log N) \) and that of channel estimation is \( O(\log (N)) \) (in [17] and [18]) or \( O(\log L) \) (in [19]). The complexity of the proposed approach depends on \( P \), but \( P \) can be a small number, e.g., in the simulations shown in Section IV, the maximum of \( P \) is only 5. The proposed approach can achieve significant performance improvement by using the EM iteration compared with the approaches in [17]-[19], which will be demonstrated in Section IV.

---

\(^3\)FFT can be used to calculate (I)DFT with complexity \( O(L \log L) \) for any length \( L \) [16].

\(^4\)Since there is no incoming message for the edge representing \( \mathbf{c}_{K-1} \), \( \overline{\mathbf{m}}_{c_{K-1}} = \mathbf{0} \) and \( \overline{\mathbf{V}}_{c_{K-1}} = \mathbf{0} \) [10].
C. Joint Channel Estimation, Detection and Decoding

The approach described in Section III.B can be easily incorporated into a turbo system [13], [14] for joint channel estimation, detection and decoding. In this case, the \( a \) priori means \( \{m_{j}^{a}\} \) and variances \( \{V_{j}^{a}\} \) of the data symbols, which are computed based on the feedback from the decoder [13], [14], can be accommodated into \( \hat{m}_{s} \) and \( \hat{V}_{s} \) according to (3) and (5), and the output LLRs, which should be extrinsic, can be calculated as

\[
L(s_{j}^{Re}) = 2\sqrt{2}(m_{j}^{Re}/V_{j} - (m_{j}^{a})^{Re}/V_{j}^{a})
\]

\[
L(s_{j}^{Im}) = 2\sqrt{2}(m_{j}^{Im}/V_{j} - (m_{j}^{a})^{Im}/V_{j}^{a}).
\]

The above result for QPSK modulation can be extended to high-order modulations. The output LLRs can be represented in the form of the extrinsic means and extrinsic variances of the data symbols. See [21] for details.

There are two iterative processes involved at the receiver, i.e., the EM iteration and the iteration between the detector and the decoder. We may treat the EM iteration as the inner iteration, which is incorporated into the iteration between the detector and the decoder (called outer iteration). From simulations, we found that when \( \alpha \) is small, the inner EM iteration does not lead to significant performance improvement. An intuitive explanation is that, when \( \alpha \) approaches zero, \( \{t_{k}\} \) are almost exactly known, so the channel estimation reduces to a conventional one with known training sequences (see (20)-(23)). To reduce the computational cost, we can set a threshold \( \alpha_{0} \) for \( \alpha \), and perform the EM iteration only once when \( \alpha < \alpha_{0} \).

D. Extension to Turbo Systems over Time-Varying Channels

As in [9], [17], [18] and [19], with the approximation that the channel is static within a short time duration, we partition the length-\( M \) signal block \( x \) into a number of short subblocks \( \{x^{z}\} \) with length-\( J \), i.e., \( x = (x^{0})^{T}, (x^{1})^{T}, ..., (x^{Z-1})^{T} \) (\( M = ZJ \)), and assume that each subblock \( x^{z} \) undergoes a static channel \( h^{z} \). The time-varying channel is approximately characterized by the following first-order AR model

\[
h^{z} = Gh^{z-1} + \omega^{z} \tag{31}
\]

where \( G = \phi I \) and \( \{\omega^{z}\} \) are independent zero mean complex Gaussian random variables with the same diagonal covariance matrix \( C \). The values of \( \phi \) and \( C \) can be determined as in [4]. Note that the frequency selective channel causes interference between adjacent neighbor subblocks (assuming that \( L < N \)), which can be handled using the interference cancellation technique [9], [17] (it can be efficiently carried out as in [9], with complexity dominated by two FFTs per subblock). For each \( x^{z} \), we have [9]

\[
y^{z} = h^{z} * x^{z} + n^{z} \tag{32}
\]

where \( y^{z} \) denotes the observation vector with length \( N = J + L - 1 \) after inter-subblock interference cancellation, and

\[\text{considering that (32) has the same form as (4), an FFG similar to that in Fig. 1 can be constructed for each subblock. The system model for the time-varying channel described by (31) - (33) can be represented by the FFG shown in Fig. 2, where the overall graph is also divided into an M-step graph and an E-step graph. In contrast to Section III.B, the variables in the M-step graph includes not only \( c^{z} \) but also \( h^{z} \) and \( \omega^{z} \). The message computations in the E-step factor graph for each subblock are the same as those in Section III.B. For the message computations in the M-step factor graph, besides the message computations in the M-step factor graph for each subblock as described in Section III.B, some extra message computations in the upper part of the graph representing (31) and (33) are needed. The message passing algorithm can be easily composed according to the computation rules (in Tables 2 and 3 in [10]),6 and are omitted here. It can be verified that the complexity of this approach is still dominated by that of (I)FFTs, and the complexity per symbol per iteration keeps the same order as that in Section III.B.

IV. Simulation Results

We examine the proposed approach in a turbo system over time-varying frequency selective channels. The number of channel taps \( L = 64 \), and the channel power delay profile is \( q^{l} = e^{-0.1l} \). We set carrier frequency \( f_{c} = 2 \)GHz, symbol duration \( T_{s} = 0.25 \mu s \), and mobile speed \( v_{m} = 500 \)km/h. Hence the normalized Doppler spread \( f_{d}T_{s} = 0.00023 \). Although we assume that the channel is static within each subblock in the design of the approach, the channels generated in the simulations vary symbol by symbol. To guarantee an approximately static frequency selective channel for each subblock, we set the subblock length \( J = 193 \) (i.e., \( Jf_{d}T_{s} = 4.439\% \)), so \( N = 256 \). For each channel realization, the average channel energy is normalized to 1, i.e., \( M^{-1}\sum_{m=0}^{M-1}\sum_{l=0}^{L-1}|h_{m,l}^{n}|^{2} = 1 \).

5A similar approximation as (24) is adopted in the forward covariance matrix updating for the multiplication node representing (33).
1, where \( h_{m,l}^q \) denotes the \( l \)th channel tap at time \( m \) in the \( q \)th Monte Carlo simulation. A rate-1/2 convolutional code with generator \((23,35)_{10}\) is employed. The transmitted signal block length \( M = 5790 \), i.e., one signal block consists of 30 subblocks. A periodically extended Chu sequence \([2]\) is defined as

\[
q = \left\{ \begin{array}{ll}
1 & \text{if } M^2 \leq N - 1 \\
0 & \text{otherwise}
\end{array} \right.
\]

with generator \( Q = [1 0 0 1 1 1 1] \). The mean square identification error (MSIE) is set to be \( 0.4 \) dB power overhead. The number of the outer iteration is set to be 10. The threshold \( \alpha_0 = 0.1 \), and the EM iteration is carried out 5 times when \( \alpha > \alpha_0 \) (otherwise it is carried out once). In the first outer iteration, the initial value \( e_{k}^{0} \) of the EM iteration (see (22) and (23)) is set to be 0, and in the following outer iterations, its value is given by its final estimate in the last outer iteration. The mean square identification error (MSIE) defined as

\[
\frac{1}{Q} \sum_{q=1}^{Q} \left( M^{-1} \sum_{m=0}^{M-1} h_{m,l}^q - h_{m,l}^q \right)^2,
\]

where \( h_{m,l}^q \) is the estimate of \( h_{m,l}^q \) in the \( q \)th simulation and \( Q \) is the number of Monte-Carlo simulations, is used to evaluate the performance of channel estimation, and the bit error rate (BER) is used to evaluate the system performance.

The MSIE performance and the BER performance of the proposed approach, the approach in [17], the approach in [18] (with approximate rule 1), and the approach in [19] (with CE2) are shown in Fig. 3. It can be seen that the proposed approach achieves considerable performance improvement. It can be seen from Fig. 3(b) that, considering the 0.4 dB power overhead due to the training sequence, the BER performance gap between the system with the proposed approach and the system with channel known is only about 1.5 dB. Note that, the performance gap may be narrowed with high order AR channel model but at the cost of increased complexity.

**V. Conclusion**

With a Gaussian assumption of the data symbols, we have developed a Gaussian message passing EM-based joint channel estimation and detection approach with logarithmic complexity by employing a frequency domain system model in the FFG design and introducing proper approximations in the message computations. The effectiveness of this approach has been demonstrated through simulations.

**References**


