Low Complexity MIMO Detection based on Belief Propagation over Pair-wise Graphs

Seokhyun Yoon, Member, IEEE and Chan-Byoung Chae, Senior Member, IEEE

Abstract

This paper considers belief propagation algorithm over pair-wise graphical models to develop low complexity, iterative multiple-input multiple-output (MIMO) detectors. The pair-wise graphical model is a bipartite graph where a pair of variable nodes are related by an observation node represented by the bivariate Gaussian function obtained by marginalizing the posterior joint probability density under the Gaussian input assumption. Specifically, we consider two types of pair-wise models, the fully-connected and ring-type. The pair-wise graphs are sparse, compared to the conventional graphical model in [18], insofar as the number of edges connected to an observation node (edge degree) is only two. Consequently the computations are much easier than those of maximum likelihood (ML) detection, which are similar to the belief propagation (BP) that is run over the fully connected bipartite graph. The link level performance for non-Gaussian input is evaluated via simulations, and the results show the validity of the proposed algorithms. We also customize the algorithm with Gaussian input assumption to obtain the Gaussian BP run over the two pair-wise graphical models and, for the ring-type, we prove its convergence in mean to the linear minimum mean square error (MMSE) estimates. Since the maximum a posterior (MAP) estimator for Gaussian input is equivalent to the linear MMSE estimator, it shows the optimality, in mean, of the scheme for Gaussian input.

Index Terms

Markov random field, low complexity MIMO detection, graph-based detection, belief propagation, sum-product algorithm, forward-backward recursion.

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A2038807) and partly by Dankook University Project for funding RICT 2011. This work was partially presented at VTC 2011 Spring.

S. Yoon is with the Department of Electronic Engineering, Dankook University, Yongin-si, Kyunggi-do, Korea, 448-160 (e-mail: syoon@dku.edu)

C.-B. Chae is with the School of Integrated Technology, Yonsei University, Korea (e-mail: cbchae@yonsei.ac.kr)
I. INTRODUCTION

Recent work on multi-input and multi-output (MIMO) detections has mainly been focused on so-called sphere decoding [1]–[6]. Sphere decoding is a two-stage detector in which the channel matrix is first converted into an upper triangular form and, utilizing this structure, a tree search is used for joint data detection. Since the full tree search has the same complexity as maximum likelihood (ML) detection, a sort of reduced search algorithm is applied by limiting the search space, e.g., the number of candidate symbols or radius at each tree search stage. One advantage of sphere decoding is that it can, by choosing an appropriate value of radius or list size, provide a tradeoff between performance and complexity. The performance of sphere decoding has been shown to be quite close to that of ML with a reasonable level of complexity [6]. To produce soft decisions required for channel decoding, however, the search space cannot be set too small.

Another type of MIMO detector, which has received little attention, is the channel truncation approach [7]–[10]. This approach is also a two-stage detector, where the channel is first converted into a bidiagonal or, more generally, a poly-diagonal form [9], [10] and, utilizing the effective channel structure, a trellis search, e.g., the Viterbi algorithm or the forward-backward algorithm [11], [12], is used for post-joint detection. The method is similar to the concatenated channel-shortening equalizer and maximum likelihood sequence estimator (MLSE) for the inter-symbol interference channel [13].

Another class of MIMO detection worthy of attention is graph based detection [14]–[20]. The approaches are based on the belief propagation (BP) algorithm [21], [22]. This algorithm has also been extensively studied for the decoding of channel codes, such as the turbo codes and low density parity check codes. In these approaches, the MIMO channel is modeled as a fully-connected bipartite graph, which consists of multiple $N$ observation nodes representing the received signal, multiple $M$ variable nodes representing the hidden data, and the edges connecting the observation nodes with the variable nodes. The resulting graph has the maximal edge degree, i.e., every observation node is connected to every variable node. When applying the BP algorithm [21] or the sum-product algorithm [22] to such graphs, the complexity is as high as the ML or MAP detector. This is mainly due to the metric computation and the marginalization operation required for the message update at the observation nodes.

To reduce the computational complexity, the Gaussian BP has been considered in [16] and [17], where the input data and messages are all assumed to be Gaussian so that the message and posterior probability can be represented by a pair of mean and variance, resulting in a very simple message update rule. As shown in [16] and [17], however, the algorithm converges (though not always) only to the linear minimum
mean squared error (LMMSE) solution, which is inferior to the ML detector for non-Gaussian input. On the other hand, [18] and [20] studied complexity reduction via model simplification. Particularly, in [18], it was suggested that to reduce the edge degree some edges in the fully connected bipartite graph should be pruned based on the strength of the channel coefficients. By doing so, not only is the number of messages reduced, but also the marginalization operation on observation nodes can be performed at a much lesser cost. Reduction in the marginalization cost is exponential with the edge-degree reduction, resulting in far less complexity than ML. The problem here, however, is that the performance loss can be more severe with the edge-degree reduction.

Other interesting graph-based approaches are those in [20], [23]–[25], based on the pair-wise Markov random field (MRF) [26]. In MRF, we have only one type of node representing the hidden data and the edges reflecting the local dependency among them. The local dependency is represented by potential functions and, specifically in pair-wise MRF, they are functions of one or two variables. In fact, as noticed in [20], [23] and [25] (also in [16] and [19]), a multivariate Gaussian function can be decomposed into a product of functions of one or two variables resulting in a fully connected pair-wise MRF. On the other hand, in [20], noticing that BP may not work well for a loopy graph, the authors proposed a tree approximation on the basis of Kullback-Leibler distance (KLD) optimality criterion. In [24], the same authors have proposed using the potential functions obtained by two dimensional projection and the expectation-maximization (EM) algorithm based post detection. Bit-based probabilistic data association is another approach to low complexity MIMO detection especially for higher order QAM. In [27], a matrix representation is introduced to represents symbol mapping, by which it can be considered as a linear processing and can be combined as part of MIMO channel giving us a room for complexity reduction for higher order QAM.

In this paper, we investigate a similar approach to the pair-wise MRF based MIMO detector, but with different formulation, i.e., instead of using the potential functions obtained from the direct decomposition of multivariate Gaussian function [20], [24] or from the two dimensional projection in [24], we propose using the functions obtained by marginalizing the posterior joint probability density under the Gaussian input assumption. As implicated in [23], [25], the corresponding bipartite graph has an edge degree of only two and the proposed scheme has much less complexity than that of ML/MAP. In addition to the fully connected pair-wise graph, we also consider the ring-type pair-wise graph. The proposed scheme can be regarded as an edge pruning technique, similar to the one in [18]. Unlike that of [18], however, the pruning is performed by a linear transformation and the performance degradation compared to the ML/MAP detector is shown to be reasonable with an edge degree of two.
This paper is organized as follows. In the next section, we briefly review the ML/MAP and the graph-based approach to MIMO detection. In Section III the proposed iterative detection algorithm is presented based on the fully-connected and ring-type pair-wise models, respectively, for non-Gaussian input. In Section IV, we customize the proposed algorithms under Gaussian input assumption (Gaussian BP), and discuss its convergence property. The performance is extensively evaluated and compared via link-level simulations in Section V and, finally, in Section VI, the concluding remarks are given.

II. SYSTEM MODEL, MAP AND GRAPH-BASED DETECTION

System Model: A Gaussian MIMO system with an $N \times M$ channel matrix $H(N \geq M)$ is modeled as

$$y = Hx + n = \sum_{k=1}^{M} h_{k} x_{k} + n$$

where $x$ is an $M \times 1$ transmitted data symbol vector, $n$ is an $N \times 1$ noise vector, $y$ is an $N \times 1$ received signal vector and $h_{m}$ is the $m$th column of $H$. A symbol $n$ is assumed to be complex Gaussian with mean 0 and covariance $\mathbb{E}[nn^{H}] = \sigma_{n}^{2}I$ and the transmitted data symbol vector $x$ is assumed to have mean 0 and covariance matrix $\mathbb{E}[xx^{H}] = I$, where $\mathbb{E}(\cdot)$ denotes expectation. In practice, each element of $x$ is usually a $2^{m}$-ary symbol drawn from a finite alphabet set $\Xi$ of size $2^{m}$ such as QPSK and 16-QAM.

MAP detection: The maximum a posteriori (MAP) detector selects $x$ that maximizes the a posteriori likelihood

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

where

$$p(y|x) = \mathcal{CN}(y; Hx, \sigma_{n}^{2}I)$$

$$p(x) = \prod_{j=1}^{M} p(x_{j})$$

with $\mathcal{CN}(y; \mu, C)$ representing a multivariate complex Gaussian probability density function (PDF) of mean $\mu$ and covariance $C$ defined as

$$\mathcal{CN}(y; \mu, C) \equiv \frac{1}{(\pi)^{M} \det C} \exp \left( -\frac{1}{2} (y - \mu)^{H} C^{-1} (y - \mu) \right)$$

where the superscript $H$ denotes Hermitian transpose. The search space of the MAP is an $M$-dimensional space, $\Xi^{M}$, and the complexity is $O(2^{mM})$. When using concatenated channel coding and MIMO, a MIMO detector is required to produce soft-decision values, i.e., log-likelihood ratio (LLR). Denoting the
\( j \)th data symbol as \( x_j(b_{j1}, b_{j2}, \cdots, b_{jm}), \) where \( b_{jk} \) is the \( k \)th bit contained in \( x_j. \) Then, LLR of \( b_{jk} \) can be obtained by first marginalizing \( p(\mathbf{x}|\mathbf{y}) \) over \( \mathbf{x} = (x_1, x_2, \cdots, x_{j-1}, x_{j+1}, \cdots, x_M) \) to get

\[
p(x_j = x|\mathbf{y}) = \sum_{\mathbf{x} \setminus x_j} p(x_1, x_2, \cdots, x_M|\mathbf{y})
\]

\[
= A \cdot \sum_{\mathbf{x} \setminus x_j \in \Xi^{M-1}} p(\mathbf{y}|x_1, x_2, \cdots, x_j = x, \cdots, x_M) \prod_{k \neq j} p(x_k)
\]

where \( A = p^{-1}(\mathbf{y}) \) is a normalizing constant, and \( x_j \)'s are assumed to be independent of each other. In (3), \( p(x_j) \) is the a priori probability of \( x_j, \) which is assumed to be uniformly distributed, i.e., \( p(x_j) = 1/2^m \) for a modulation size of \( 2^m. \) The LLR for each bit is then computed as

\[
\text{LLR}(b_{jk}) = \log \left( \frac{p(b_{jk} = 0|\mathbf{y})}{p(b_{jk} = 1|\mathbf{y})} \right) = \log \left( \frac{\sum_{\text{all } x_j: b_{jk} = 0} p(x_j|\mathbf{y})}{\sum_{\text{all } x_j: b_{jk} = 1} p(x_j|\mathbf{y})} \right).
\]

The prohibitive complexity, when \( m \) and \( M \) are large, comes from the marginalization operation in (3).

**Graph-based detection (BP over fully-connected bipartite graph):** The MAP detections in (3) is useful for turbo equalization [28], [29], where one can find a vast amount of literature showing the validity of iterative MIMO detection and channel decoding. Although turbo equalization is not our main focus in this paper, it is worthy of paying attention to the iterative detection as shown in [18], i.e., the BP over the fully-connected bipartite graph. In fact, the MAP detection in (3) can be regarded as a BP that is run over the singly connected factor graph as shown in Fig. 1(a), where each variable node, representing a data symbol, first passes a priori information to the observation node labeled by the received vector, \( \mathbf{y}. \) The observation node then provides each variable node with the corresponding a posteriori likelihood by computing the marginalization in (3). Since the graph is singly connected and all variable nodes are connected via one observation node, the BP over this graph will surely converge, in one iteration, to the correct a posteriori probability. The graph-based detection in [18], on the other hand, is a BP over the fully connected bipartite graph as shown in Fig. 1(b), where the marginalization is performed separately for each observation node and they are then combined to produce the belief and the extrinsic information on each data symbol. The algorithm in [18] can be summarized as follows.

**BP 1 over the fully-connected factor graph [18]**

For given a priori probability of \( x_j, \) which is typically assumed to be uniformly distributed, i.e., \( p(x_j) = 1/2^m \) for a modulation size of \( 2^m \)

1. Initialization:

\[
\lambda_{j \rightarrow i}(x_j) = p(x_j).
\]
(2) Observation node computation:
\[
\pi_{i \rightarrow j}(x_j) = A \cdot \sum_{x \in \mathbb{Z}^{M-1}} p(y_i|x_1, x_2, \ldots, x_M) \prod_{k \neq j} \lambda_{k \rightarrow i}(x_k).
\]

(3) Belief update:
\[
b(x_j) = \prod_{k=1}^{M} \pi_{k \rightarrow j}(x_j).
\]

(4) Variable node computation:
\[
\lambda_{j \rightarrow i}(x_j) = \prod_{k \neq j} \pi_{k \rightarrow j}(x_j) = \frac{b(x_j)}{\pi_{i \rightarrow j}(x_j)}.
\]

The message update (5)-(7) are repeated by a pre-defined number or until the belief does not change any more.

Note that, in (5), \( p(y_i|x_1, x_2, \ldots, x_j = x, \ldots, x_M) \) is given by
\[
p(y_i|x_1, x_2, \ldots, x_M) = \mathcal{N}\left(y_i; \sum_{j=1}^{M} h_{ij}x_j, \sigma^2\right)
\]
and, by combining (5) and (6), we see that, at the first iteration
\[
b(x_j) \propto \prod_{k=1}^{M} p(x_j|y_k)
\]
which is certainly different from \( p(x_j|y) \) in (3). That is, in BP algorithm, we first marginalize \( p(x|y_k) \) for each received signal \( y_k \) to obtain \( p(x_j|y_k) \) and, then, the belief is obtained by their product, while, in MAP, we just marginalize \( p(x|y) \), once and for all. Note also that since the marginalization in (5) is performed over \( M - 1 \) dimensional space and must be performed for the total number \( 2^m \) states of \( x_j \), the complexity for one iteration is the same as that of MAP and the total complexity is multiplied by the number of iteration resulting in far complex computation than that of MAP detection. Regardless of its complexity, however, it provides a base structure for the development of low complexity detector.

**Complexity Reduction via Edge Pruning:** To reduce the computational burden of the marginalization in (5) for non-Gaussian input, [18] proposed pruning some edges of which the corresponding variable and observation nodes are weakly coupled together, e.g., those variable-observation node pairs with small values of \(|h_{jk}|\). By using only (edge degree) \( d_f \) edges per observation node (i.e., pruning \( M - d_f \) edges), the complexity is reduced by a factor of \( 1/2^{m(M-d_f)} \) relative to the ML/MAP or the BP 1 of complexity \( O(2^{mM}) \). Here \( d_f \) is the edge degree. The problem with this scheme is that \( d_f \) must be large enough to ensure a reasonable performance, as shown in [18].
III. DETECTION ALGORITHM BASED ON PAIR-WISE GRAPHICAL MODELS

In this section, we develop low complexity iterative MIMO detection algorithms based on the pair-wise graphical models. We consider two types, namely, the fully-connected and the ring-type, and derive the corresponding BP algorithms that work for non-Gaussian input. As will be shown below, BP over the ring-type pair-wise graph is, with a slight difference, effectively equivalent to the one in [10].

A. BP based on pair-wise Markov Random Field

Our starting point is the BP algorithm based on pair-wise Markov random field (MRF) in [19], [20] and [25]. MRF is an undirected graph that describes local dependencies among a set of random variables. In MRF, the joint PDF of all random variables involved can be represented by a product of the joint PDF of each clique.\footnote{A clique in a graph is defined by a set of nodes having full-connection to each other.} The pair-wise MRF means that a joint PDF (of all variables involved) is represented by a product of joint PDFs with only two variables corresponding to an edge connecting any two neighbors. Let $V = \{1, 2, \cdots, M\}$ be the set of nodes in the MRF corresponding to the random variables $x_1, x_2, \cdots, x_M$, respectively, and let $E$ be the set of all edges connecting these nodes. For a compact expression, we also denote the edge connecting nodes $j$ and $k$ as $e(j,k)$ and the set of neighbors of the $j$th node as $V(j)$. In pair-wise MRFs, the \textit{a posteriori} joint function $p(x_1, x_2, \cdots, x_M | y)$ is modeled by a product of pair-wise potential functions [17], [26], e.g.,

$$
\hat{p}(x_1, x_2, \cdots, x_M | y) = A \prod_{i \in V} \psi_i(x_i) \prod_{(i,j):e(i,j) \in E} \phi_{ij}(x_i, x_j),
$$

where $\psi(x_i)$ is self-potential assigned to each node and $\phi(x_i, x_j)$ is the edge potential assigned to each edge. Such modeling based on a pair-wise MRF can also facilitate the marginalization to finally obtain the marginal distribution for each random variable. Denoting the (incoming) message from the $i$th to the $j$th node as $\pi_{i \rightarrow j}(x_j)$, the BP through the pair-wise MRF can be described as [17]

$$
\pi_{i \rightarrow j}(x_j) = \alpha \sum_{x_i \in \Xi} \psi_i(x_i) \phi_{ij}(x_i, x_j) \prod_{k \in V(i) \setminus j} \pi_{k \rightarrow i}(x_i)
$$

where, $\alpha$ is the normalizing constant, $V(i) \setminus j$ is the set of neighbors of node $i$ excluding node $j$. Note that we follow the convention in [17], [19], and [20] to describe the message passing over a MRF, where only one type of node, say the variable nodes, exist and the message flies between these variable nodes. When we use a bipartite graph as shown in Fig. 1(b), we need to define two types of messages, i.e., one from variable node to observation node and the other from observation node to variable node, which can be
easily obtained by dividing (9) into two separate steps, i.e., \( \lambda_{i \rightarrow j}(x_i) = \prod_{k \in N(i) \cup j} \pi_{k \rightarrow i}(x_i) \) (variable-to-observation node message) and \( \pi_{i \rightarrow j}(x_j) = \alpha \sum_{x_i \in \Xi} \psi_i(x_i) \phi_{ij}(x_i, x_j) \cdot \lambda_{i \rightarrow j}(x_i) \) (observation-to-variable node message).

In (9), the incoming messages are combined first to produce the extrinsic information, \( \prod_{k \in V(i) \cup j} \pi_{k \rightarrow i}(x_i) \), and they are then “translated” by the potential function, \( \psi_i(x_i) \phi_{ij}(x_i, x_j) \). The belief on the variable, \( x_j \), is given by

\[
b(x_j) = \prod_{k \in V(i)} \pi_{k \rightarrow j}(x_j). \tag{10}\]

The potential functions in (8) is given by a factorization of the joint a posteriori probability. Specifically, in [17], [27], the potential function is obtained by decomposition of multivariate Gaussian function, i.e.,

\[
\phi_{i,j}(x_i, x_j) = A_{ij} \exp \left( -\frac{1}{\sigma^2} \text{Re}[x_i^* R_{ij} x_j] \right) \\
\psi_i(x_i) = A_i \exp \left( -\frac{1}{\sigma^2} \text{Re}[x_{ij}^* y_j' - R_{ii}|x_i|^2] \right) \tag{11}\]

where \( R_{ij} = h_i^T h_j \) and \( y_j' = h_i^T y \), and * denotes complex conjugate. In fact, such decomposition gives us a fully connected pair-wise MRF and is exact in the sense that (9) with the functions in (11) is exactly the same as the joint Gaussian PDF. It has been shown in [16] and [17] that, with (11), the BP over the fully connected pair-wise MRF results in the MMSE solution if it converges (though the convergence is not always guaranteed for arbitrary channel matrices). Most of all, however, it does not work well for non-Gaussian input and the performance is shown to be inferior to the ML/MAP detector, especially for higher order modulation.

B. The proposed BP algorithm over pair-wise graphical models

In this paper, we propose using the following message passing rule.

\[
\pi_{i \rightarrow j}(x_j) = \alpha \sum_{x_i \in \Xi} \tilde{p}(x_j|x_i, y) \prod_{k \in V(i) \cup j} \pi_{k \rightarrow i}(x_i). \tag{12}\]

where \( \tilde{p}(x_j|x_i, y) \) is the conditional a posteriori probability derived under a Gaussian input assumption to be discussed shortly. Comparing with (9), the potential function in (9) is replaced with \( \tilde{p}(x_j|x_i, y) \). Note, however, that it is not a factor of the a posteriori probability in (1), unlike those in (11).

The trick here is to use \( \tilde{p}(x_j|x_i, y) \) obtained under Gaussian input assumption in order to approximate the marginal PDF of non-Gaussian data. Note also that although the translation function \( \tilde{p}(x_j|x_i, y) \) is obtained under the Gaussian assumption on the data symbol, the message itself, \( \pi_{i \rightarrow j}(x_j) \), will not be treated as Gaussian. The rationale of using \( \tilde{p}(x_j|x_i, y) \) is to reduce the computational complexity. Let
\( p(x_j|x_i, y) \) be the true conditional *a posteriori* probability without the Gaussian assumption. Further assume that, after many iterations, the extrinsic information \( \prod_{k \in V(i) \setminus j} \pi_{k \to i}(x_i) \) for the \( i \)th node (a neighbor of the \( j \)th node) converges to its true *a posteriori* marginal distribution, \( p(x_i|y) \). Then, with an appropriate normalizing constant, we also have \( \pi_{i \to j}(x_j) \to p(x_j|y) \) for the \( j \)th node, which means, once converged, this translation function ensures that the final belief is given by the true marginal *a posteriori* distribution. This is actually a non-sense since, before we run the algorithm, we need first to compute \( p(x_j|x_i, y) \), which, however, has a complexity of ML detection. Hence, at this step, we assume \( x_j \)'s are all Gaussian to obtain \( \tilde{p}(x_j|x_i, y) \), of which the computation is much simpler as to be discussed shortly.

It is a simple trick to use \( \tilde{p}(x_j|x_i, y) \) obtained under Gaussian input assumption to approximate the true posterior marginal for non-Gaussian input (i.e., \( p(x_j|y) \)).

On the other hand, the conditional PDF, \( \tilde{p}(x_j|x_i, y) \), under Gaussian input assumption can be easily obtained from the following simple probability relations, i.e.,

\[
\tilde{p}(x_i, x_j|y)\tilde{p}(y) = \tilde{p}(y|x_i, x_j)\tilde{p}(x_i, x_j) = \tilde{p}(x_j|x_i, y)\tilde{p}(x_i, y)
\]

resulting in

\[
\tilde{p}(x_j|x_i, y) = \frac{\tilde{p}(x_i, x_j|y)\tilde{p}(x_j)}{\tilde{p}(y|x_i, y)} = \frac{\tilde{p}(y|x_i, x_j)\tilde{p}(x_j)}{\tilde{p}(y|x_i)}
\]

where

\[
\tilde{p}(y|x_i, x_j) = \mathcal{CN}(y; h_i x_i + h_j x_j, K_{\{j,i\}})
\]

\[
\tilde{p}(y|x_i) = \mathcal{CN}(y; h_i x_i, K_{\{i\}})
\]

\[
\tilde{p}(x_i) = \mathcal{CN}(x_i; 0, 1)
\]

with

\[
K_\Phi = \sigma^2 I + \sum_{k \notin \Phi} h_k h_k^H
\]

for \( \Phi = \{i,j\} \) or \( \{i\} \). In the second equality in (13), we used the independence assumptions on \( x_j \)'s.

Moreover, the Gaussian input assumption leads us to a much simpler form. First, define the conditional MMSE estimator for \( x_j \) given \( x_i \),

\[
c_{j|i} = K_{\{j,i\}}^{-1} h_j
\]

and \( y'_{j|i} = c_{j|i}^H y \) such that

\[
y'_{j|i} = c_{j|i}^H y = a_{j|i,j} x_j + a_{j|i,i} x_i + n'_{j|i}
\]
where

\[
\begin{align*}
  a_{j|i,k} &= c^H_{j|i} h_k = h^H_j K^{-1}_{(j,i)} h_k \quad \text{for } k = i \text{ or } j \\
  \mathbb{E}|n'_{j|i}|^2 &= c^H_{j|i} K_{(j,i)} c_{j|i} = h^H_j K^{-1}_{(j,i)} h_j \equiv \sigma^2_{j|i}. 
\end{align*}
\]

Then, (13) can be rewritten as

\[
\tilde{p}(x_j | x_i, y'_{j|i}) = \frac{\tilde{p}(y'_{j|i} | x_i, x_j) \tilde{p}(x_j)}{\tilde{p}(y'_{j|i} | x_i)}
\]

with

\[
\tilde{p}(y'_{j|i} | x_i, x_j) = \mathcal{CN}(y'_{j|i}; a_{j|i,j} x_j + a_{j|i,i} x_i, \sigma^2_{j|i})
\]

\[
\tilde{p}(y'_{j|i} | x_i) = \mathcal{CN}(y'_{j|i}; a_{j|i,i} x_i, \sigma^2_{j|i} + |a_{j|i,j}|^2).
\]

In (23), we used \( p(x_j) = \mathcal{CN}(x_j; 0, 1) \). Plugging (22) and (23) into (21) and by replacing \( p(x_j) \) with \( \mathcal{CN}(x_j; 0, 1) \), we have the simplified translation function from the derivation in the appendix.

\[
\tilde{p}(x_j | x_i, y'_{j|i}) = \mathcal{CN}
\left(x_j; \frac{a^{*}_{j|i,j}}{\sigma^2_{j|i} + |a_{j|i,j}|^2} (y'_{j|i} - a_{j|i,i} x_i), \frac{\sigma^2_{j|i}}{\sigma^2_{j|i} + |a_{j|i,j}|^2}\right)
\]

\[
= \mathcal{CN}
\left(x_j; \frac{1}{1 + \sigma^2_{j|i}} (y'_{j|i} - a_{j|i,i} x_i), \frac{1}{1 + \sigma^2_{j|i}}\right)
\]

where, in the last line, we used the fact that \( a_{j|i,j} \) is real valued and is equal to \( \sigma^2_{j|i} \). Note that in (24), the mean is the conditional MMSE estimate of \( x_j \) given \( x_i \).

Using (17) to (24), the proposed message passing rule can be summarized as follows.

BP 2 over the fully-connected pair-wise graph

Given the messages in the previous iteration, \( \pi_{k \rightarrow i}(x_i) \).

1. Compute the extrinsic information for all pairs \((i,j)\) with \( i \neq j \)

\[
\lambda_{i \rightarrow j}(x_i) = \prod_{k \in V(i) \setminus j} \pi_{k \rightarrow i}(x_i).
\]

2. Translate the message \( \lambda_{i \rightarrow j}(x_j) \) to \( \pi_{i \rightarrow j}(x_j) \)

\[
\pi_{i \rightarrow j}(x_j) = \alpha \sum_{x_i \in \Xi} \tilde{p}(x_j | x_i, y'_{j|i}) \cdot \lambda_{i \rightarrow j}(x_i).
\]

with \( \tilde{p}(x_j | x_i, y'_{j|i}) \) given by (24). The above message passing is computed for all edges in both directions, and they are repeated by a pre-defined number or until the messages do not change any more. The belief is finally obtained the same as that in (10).
Note that the above algorithm uses two types of message and can be efficiently described by a message passing over a bipartite graph in Fig. 2(a), where the observations used for the message translation from the \( j \)th node to the \( i \)th and its reverse is clearly denoted by \( y'_{ji} \) and \( y'_{ij} \), respectively. It is also interesting to note that the above algorithm is similar to the algorithms in [9] and [10] with two differences. One is in the underlying structure and the other in message translation. To clarify the similarity and difference, we consider the ring-type bipartite graph shown in Fig. 2(b). In this ring-type graph, each node has only two neighbors and, hence, in the computation of extrinsic information, the incoming message from one neighbor is simply passed to the other and the detection algorithm can be described more concisely and clearly as follows (even though it can be generally applicable to any pair-wise graphical model).

\textbf{BP 3 over the ring-type pair-wise graph (Forward-backward recursion)}

Given the messages in the previous iteration, \( \pi_{k \rightarrow i}(x_i) \),

1. Variable node to observation node message

\[
\lambda_{j \rightarrow (j \pm 1)_M}(x_j) = \pi_{(j \mp 1)_M \rightarrow j}(x_j) \quad \forall j. \tag{27}
\]

2. Observation node to variable node message

\[
\pi_{j \rightarrow (j \pm 1)_M}(x_{(j \pm 1)_M}) = \sum_{x_j \in \Xi} \tilde{p}(x_{(j \pm 1)_M} | x_j, y'_{(j \pm 1)_M}) \cdot \lambda_{j \rightarrow (j \pm 1)_M}(x_j) \quad \forall j. \tag{28}
\]

with \( \tilde{p}(x_j | x_i, y'_{ji}) \) given by (24). After a pre-defined number of iterations, the belief is finally obtained by

\[
b(x_j) = \pi_{(j+1)_M \rightarrow j}(x_j) \cdot \pi_{(j-1)_M \rightarrow j}(x_j). \tag{29}
\]

From (27) to (29), \((\cdot)_M\) denotes the 1-base modulo-\( M \) operation such that \((M+1)_M = 1\) and \((0)_M = M\). Later on, however, we will omit this for notational simplicity.

On the other hand, this message update rule is a forward-backward algorithm similar to those in [9], i.e., the message from the \((j-1)\)th node to the \(j\)th node corresponds to the forward message, and the one from the \((j+1)\)th node to the \(j\)th node corresponds to the backward message. The difference is in the message translation. In (28), the message translation from the \(j\)th node to the \(i\)th and its reverse utilize a different translation function, i.e.,

\[
y'_{ji} \neq y'_{ij} \Rightarrow p(x_j | x_i, y'_{ji}) = \frac{p(y'_{ji} | x_i, x_j)p(x_j)}{p(y'_{ji} | x_i)} \neq p(x_i | x_j, y'_{ij}).
\]

This means the branch metrics used for the forward and backward recursion are separately optimized to maximize their conditional SINR, as also proposed in [10]. The translation function is also different from...
the branch metric in [10], i.e., the mean and variance in (24) have a scaling factor of $a^*_j|i,j/(\sigma^2_j + |a^*_j|i,j|^2)$ and $1/(\sigma^2_j + |a^*_j|i,j|^2)$, respectively, instead of $a^*_j|i,j/|a^*_j|i,j|^2$ and $1/|a_j|i,j|^2$, though it has a minor impact on the error rate performances. The bipartite graphs corresponding to this algorithm is shown in Fig. 2(b), where the observation used for the message translation from the $j$th node to the $i$th and its reverse is clearly denoted by $y_j^\prime i$ and $y_i^\prime j$, respectively. Note that, for ring-type graph, we obtain different performance with a different antenna permutation, as also noted in [7], while, in the fully-connected one, we do not need antenna permutation, which is one possible advantage of the latter to the former.

Since the graphical models have short cycle(s) (especially the fully-connected pair-wise graph), it is quite questionable whether or not BP 2 and 3 will converge. In the literature, it was known that the convergence of BP over a loopy graph is not guaranteed, even though it does converge in most practical cases. Since the convergence proof for non-Gaussian input is not tractable, we will tackle this question in the next section by modifying them for Gaussian input.

C. Complexity

For complexity comparisons, we need to consider both the linear preprocessing and the post iterative detections. Consider first the computational complexity of the post iterative detection only. In the MAP detector, the distance metric $|y - Hx|^2$ is computed first for all combinations of $(x_1, x_2, \ldots, x_M) \in \Xi^M$ and, then, the marginalization in (3) is performed over all combinations of $x_j \in \Xi^{M-1}$ for each of $2^m$ alphabet, resulting in a complexity of $O(M^2 \cdot 2^m M)$. Comparing with the complexity of MAP detector, the computational burden in the BP2 for the fully-connected pair-wise graph in Fig. 2(a) for $\nu$ iterations is $O(\nu \cdot M(M - 1) \cdot 2^m M)$ since the marginalization for each $M$ node is performed separately for its $(M - 1)$ neighbors and repeated $\nu$ times. Although some additional computation is required for the linear processing in (16)-(20), it is typically much smaller than $2^m(M - 1)$, resulting in considerable computational reduction, which certainly comes from modeling through the pair-wise graphical model. On the other hand, the computational complexity for the ring-type pair-wise graph in Fig. 2(b) is $O(\nu \cdot M \cdot 2^m)$, which is even less than that of the fully-connected one.

To evaluate approximate number of operations, we assume:

1) The marginalization in (3) for the MAP and the computation in (26) and (28) for the BP 2 and 3, respectively, are performed in log-domain, where multiplications and additions in these equations are replaced with addition and max-operation, respectively, and, in (2) and (24), we only need to compute its exponent.
2) A multiplication of a \((p \times q)\) matrix with a \((q \times r)\) matrix requires \(pqr\) times of multiplications and additions (of complex numbers).

3) An inversion of a \((p \times p)\) square matrix approximately requires \(2p^3 - 2p^2\) times of multiplications and \(2p^3 - p^2\) times of additions (of complex numbers).

4) Division of complex numbers requires one complex multiplication and two real divisions.

5) A complex addition requires two real additions and complex multiplication requires four real multiplications and two real additions.

6) Real addition and multiplication are assumed to have the same complexity of one (operation), while real division to have 8 (operations).

With these assumptions, we can count the number of operations required to generate the symbol likelihoods, i.e., the \textit{a posteriori} likelihood in (3) for the MAP and the final beliefs in the BP2 and BP3. We do not count the generation of LLR for each bit from the symbol likelihood since it is the same for all detectors. The results are summarized in Table.1, where we also show two examples, one with \(M = 6\), \(m = 2\), \(\nu_1 = 4\), \(\nu_2 = 6\) and the other with \(M = 4\), \(m = 4\), \(\nu_1 = 4\), \(\nu_2 = 6\), where \(\nu_1\) and \(\nu_2\) are the number of iterations for the BP2 and BP3, respectively.

It will be interesting to compare the complexity of the proposed schemes with the one in [27] (Table I). As analyzed for BP2 and BP3, the complexity can be considered separately for the preprocessing and the post decoding. For the latter, the complexity of the one in [27] should be the same as that of the BP2, though it would be more complex than that of the BP3 in our proposal. The main difference is in the preprocessing stage. Certainly, the complexity of the preprocessing in [27] is much less than that of the proposed preprocessing since it consists of only two matrix multiplications, i.e., \(HHH\) and \(HHR\), which requires \(M^3 + M^2\) of complex multiplications and the same number of complex additions.

IV. MESSAGE PASSING WITH GAUSSIAN INPUT

In Section III, we developed BP algorithms run over the pair-wise bipartite graphs for non-Gaussian messages. The Gaussian assumption on \(x_j\)’s was employed first to obtain the translation function in (24). While, we used the exact marginalization in the message translation step. In this section, we further simplify the message passing rule by extending the Gaussian assumption to the message translation step, as was done in [16], [17], [19], to obtain the Gaussian BP over the two graphical models under consideration.

\textit{ML detection with Gaussian input:} With independent and identically distributed Gaussian input, \(p(x) = \)
\[ \prod_{j=1}^{M} \mathcal{N}(x_j; 0, 1), \] the MAP detector in (3) becomes
\[ p(x_j|y) = A \cdot \int \cdots \int \mathcal{N}(y; Hx, \sigma^2) \prod_{k \neq j} \mathcal{N}(x; 0, 1) dx \mid x_j = \mathcal{N}(x_j; h_j^H K^{-1} y, 1 - h_j^H K^{-1} h_j) \] where we appropriately select a normalization constant \( A \), while the covariance matrix \( K \), is given by \( K = (H H^H + \sigma^2 I) \). Noting that, in (30), the mean is the linear MMSE estimates of \( x_j \) and the variance is the corresponding minimum MSE, i.e.,
\[ \hat{x}_j = h_j^H K^{-1} y \]
\[ \text{MMSE}_j = 1 - h_j^H K^{-1} h_j. \]
This means that linear MMSE estimation is optimum for the Gaussian input, while it does not hold for non-Gaussian input.

**A. Gaussian BP over the proposed pair-wise graphs**

Assuming that \( x_j \)'s are Gaussian and the distributions \( \pi_{i \rightarrow j}(x_j) \), and \( b(x_i) \) are all Gaussian PDFs, they can be characterized by their mean and variance only. This means the messages \( \pi_{i \rightarrow j}(x_j) \) and the belief, \( b(x_i) \), in the BP 2 and 3 can be replaced with the update rule for the mean and variance pair. Since the Gaussian BP corresponding to the BP 1 over the fully connected pair-wise graph in (5)-(7) has already been discussed in [16], we consider here only the BP 2 and 3 over the two pair-wise graphical models.

Let us denote the mean and the variance pair of the complex Gaussian PDFs, \( \pi_{i \rightarrow j}(x_j) \), and \( b(x_i) \) as \( (\mu_{\pi, i \rightarrow j}, \sigma_{\pi, i \rightarrow j}^2) \) and \( (\mu_i, \sigma_i^2) \). Then, the BP 2 and 3 under the Gaussian input assumption can be rewritten as follows (Detailed derivations are shown in the appendix):

**Gaussian BP 2G over the fully-connected pair-wise graph**

Given the messages in the previous iteration (or the initial messages), \( \pi_{i \rightarrow j}(x_i) = (\mu_{\pi, i \rightarrow j}, \sigma_{\pi, i \rightarrow j}^2) \ \forall (i, j) : i \neq j \), they are recursively updated by
\[ \sigma_{\pi, i \rightarrow j}^2 = \frac{1}{1 + \sigma_{j i}^2} + \frac{|a_{j i}|^2}{(1 + \sigma_{j i}^2)^2} \sum_{k \in V(i) \setminus j} \sigma_{\pi, k \rightarrow i}^{-2} \]
\[ \mu_{\pi, i \rightarrow j} = \frac{y_{j i}}{1 + \sigma_{j i}^2} - \frac{a_{j i}}{1 + \sigma_{j i}^2} \sum_{k \in V(i) \setminus j} \sigma_{\pi, k \rightarrow i}^{-2} \sigma_{\pi, k \rightarrow i}^{-1}. \]
After a number of iterations of the above, the final belief on \( x_i \) is obtained by
\[ \sigma_i^{-2} = \sum_{k \in V(i)} \sigma_{\pi, k \rightarrow i}^{-2} \]
\[ \mu_i = \frac{\sum_{k \in V(i)} \sigma_{\pi, k \rightarrow i} \sigma_{\pi, k \rightarrow i}^{-1} \mu_{\pi, k \rightarrow i}}{\sum_{k \in V(i)} \sigma_{\pi, k \rightarrow i}^{-2}}. \]
Gaussian BP 3G over the ring-type MRF (Gaussian forward-backward recursion)

Given the messages in the previous iteration, \( \pi_{i \rightarrow i \pm 1} (x_i) = (\mu_{\pi, i \rightarrow i \pm 1}, \sigma^2_{\pi, i \rightarrow i \pm 1}) \) \( \forall i \), they are recursively updated by

\[
\sigma^2_{\pi, i \rightarrow i \pm 1} = \frac{1}{1 + \sigma^2_{i \pm 1 | i}} + \frac{|a_{i \pm 1 | i,i}|^2}{(1 + \sigma^2_{i \pm 1 | i})^2} \cdot \sigma^{-2}_{\pi, i \mp 1 \rightarrow i} \tag{37}
\]

\[
\mu_{\pi, i \rightarrow i \pm 1} = \frac{1}{1 + \sigma^2_{i \pm 1 | i}} y_{i \pm 1 | i} - \frac{a_{i \pm 1 | i,i}}{1 + \sigma^2_{i \pm 1 | i}} \cdot \mu_{\pi, i \mp 1 \rightarrow i}. \tag{38}
\]

After a number of iterations of the above, the final belief on \( x_i \) is obtained by

\[
\sigma^{-2}_i = \sigma^{-2}_{\pi, i+1 \rightarrow i} + \sigma^{-2}_{\pi, i-1 \rightarrow i} \tag{39}
\]

\[
\mu_i = \frac{\sigma^{-2}_{\pi, i+1 \rightarrow i} \mu_{\pi, i+1 \rightarrow i} + \sigma^{-2}_{\pi, i-1 \rightarrow i} \mu_{\pi, i-1 \rightarrow i}}{\sigma^{-2}_{\pi, i+1 \rightarrow i} + \sigma^{-2}_{\pi, i-1 \rightarrow i}}. \tag{40}
\]

Particularly, in the Gaussian BP 3G, we observe the following:

1) The variance and mean are updated separately (except in the final belief).
2) In (37) and (38), there are two separate message flows; one is the forward from \( i \) to \( i+1 \) and the other is the backward from \( i \) to \( i-1 \).
3) Eq. (38) can be rewritten as

\[
\text{Forward recursion: } \mu_{\pi, i \rightarrow i+1} = F_i \circ \mu_{\pi, i-1 \rightarrow i} \tag{41}
\]

\[
\text{Backward recursion: } \mu_{\pi, i \rightarrow i-1} = B_i \circ \mu_{\pi, i+1 \rightarrow i} \tag{42}
\]

where the operations, \( F_i \) and \( B_i \), are first order elementary function defined as

\[
F_i \circ \mu \equiv u_{i+1,i} + v_{i+1,i} \cdot \mu \tag{43}
\]

\[
B_i \circ \mu \equiv u_{i-1,i} + v_{i-1,i} \cdot \mu \tag{44}
\]

with

\[
u_{j,i} = \frac{y_{j|i}}{1 + \sigma^2_{j|i}} = \frac{h^H_{j} K^{-1}_{\{j,i\}} y}{1 + h^H_{j} K^{-1}_{\{j,i\}} h_{j}} = h^H_{j} K^{-1}_{\{i\}} y \tag{45}
\]

\[
u_{j,i} = -\frac{a_{j|i,i}}{1 + \sigma^2_{j|i}} = -\frac{h^H_{j} K^{-1}_{\{j,i\}} h_{i}}{1 + h^H_{j} K^{-1}_{\{j,i\}} h_{j}} = -h^H_{j} K^{-1}_{\{i\}} h_{i} \tag{46}
\]

Here, we used (17)-(20) and, in the last, the matrix inversion lemma

\[(A + BB^H)^{-1} = A^{-1} - A^{-1}B(I + B^HA^{-1}B)^{-1}B^HA^{-1}.\]
4) Similar to the means, (37) can also be rewritten as

\[
\text{Forward recursion: } \sigma^2_{\pi,i \rightarrow i+1} = F'_{i} \circ \sigma^2_{\pi,i \rightarrow i+1} \tag{47}
\]

\[
\text{Backward recursion: } \sigma^2_{\pi,i \rightarrow i-1} = B'_{i} \circ \sigma^2_{\pi,i \rightarrow i-1} \tag{48}
\]

where

\[
F'_{i} \circ \mu \equiv u'_{i+1,i} + v'_{i+1,i} \cdot \mu \tag{49}
\]

\[
B'_{i} \circ \mu \equiv u'_{i-1,i} + v'_{i-1,i} \cdot \mu \tag{50}
\]

with

\[
u'_{ji,i} = \frac{1}{1 + \sigma^2_{ji,i}} = \frac{1}{1 + h_j^H K_{(j,i)}^{-1} h_j} \tag{51}
\]

\[
v'_{ji,i} = \frac{|a_{ji,i}|^2}{(1 + \sigma^2_{ji,i})^2} = \frac{h_j^H K_{(j,i)}^{-1} h_i}{1 + h_j^H K_{(j,i)}^{-1} h_j} = \left| h_j^H K_{(j,i)}^{-1} h_i \right|^2. \tag{52}
\]

B. Convergence of Gaussian BP

Regarding the convergence of Gaussian BP, it was previously shown in [30] that Gaussian BP for arbitrary topology converges to the correct mean (see also [31]). It was shown in [16] that the Gaussian BP over the factor graph in Fig. 1(b) converges to the linear MMSE solution, even though its convergence is not assured. Based on these findings, we can conjecture that, for both the Gaussian BP of rules 2G and 3G, the mean converges to the linear MMSE solution, as also verified by simulations in the next section.

One way to prove the convergence would be to use the idea of the “unwrapped tree” presented in [30]. In our case, however, this would be a tedious derivation. Therefore, we try an alternative approach that works for GBP 3G, but not for GBP 2G. Note, however, that the derivation here differs from [16], [17] in the underlying graphical model and the translation function used. The objective in this subsection is to prove the following theorem.

**Theorem 1:** In the Gaussian BP 3G over the ring-type pair-wise graph, the mean converges to the linear MMSE estimate (31) for non-zero noise power as the number of iterations approaches infinity.

The proof is based on the following Lemmas.

**Lemma 2:** For an arbitrary initial value \( \mu(0) \), both the forward and backward recursions for the mean in (38) converge respectively to a unique, fixed point.

**Proof:** Define one iteration as one complete turn of a message passing along the ring and consider, without loss of generality, the message at Node 1. Based on observations 1) through 3) in the previous
subsection, we obtain the recursive relations for Node 1, i.e., using an arbitrary initial value \( \mu(0) \), we have

\[
\mu_{\pi,1 \rightarrow 2}(n) = (F_M \circ \cdots \circ F_3 \circ F_2 \circ F_1 \circ) \mu_{\pi,1 \rightarrow 2}(n - 1)
\]  
(53)

\[
= (F_M \circ \cdots \circ F_3 \circ F_2 \circ F_1 \circ)^n \mu(0)
\]

\[
\mu_{\pi,1 \rightarrow M}(n) = (B_2 \circ B_3 \circ \cdots \circ B_M \circ B_1 \circ) \mu_{\pi,1 \rightarrow M}(n - 1)
\]
(54)

\[
= (B_2 \circ B_3 \circ \cdots \circ B_M \circ B_1 \circ)^n \mu(0)
\]

where \( n \) is the iteration number and the collective operations for one iteration of the forward/backward recursion are given, respectively, by

\[
F_{1,T} \circ \mu = F_M \circ \cdots \circ F_3 \circ F_2 \circ F_1 \circ \mu = f_{1,U} + f_{1,V} \mu
\]
(55)

\[
B_{1,T} \circ \mu = B_2 \circ B_3 \circ \cdots \circ B_M \circ B_1 \circ \mu = b_{1,U} + b_{1,V} \mu
\]
(56)

for some constants, \( f_{1,U}, f_{1,V}, b_{1,U}, \) and \( b_{1,V} \), which, in turn, are monomials of \( u_{j,i} \) and \( v_{j,i} \) in (43) and (44). For example, we have for \( M = 4 \)

\[
F_4 \circ F_3 \circ F_2 \circ F_1 \circ \mu = (u_{1,4} + v_{1,4}u_{4,3} + v_{1,4}v_{4,3}u_{3,2} + v_{1,4}v_{4,3}v_{3,2}v_{2,1}) + (v_{1,4}v_{4,3}v_{3,2}v_{2,1}) \cdot \mu
\]

\[
B_2 \circ B_3 \circ B_4 \circ B_1 \circ \mu = (u_{1,2} + v_{1,2}u_{2,3} + v_{1,2}v_{2,3}u_{3,4} + v_{1,2}v_{2,3}v_{3,4}u_{4,1}) + (v_{1,2}v_{2,3}v_{3,4}u_{4,1}) \cdot \mu
\]

for which

\[
f_{1,U} = u_{1,4} + v_{1,4}u_{4,3} + v_{1,4}v_{4,3}u_{3,2} + v_{1,4}v_{4,3}v_{3,2}u_{2,1}
\]

\[
f_{1,V} = v_{1,4}v_{4,3}v_{3,2}v_{2,1}
\]

\[
b_{1,U} = u_{1,2} + v_{1,2}u_{2,3} + v_{1,2}v_{2,3}u_{3,4} + v_{1,2}v_{2,3}v_{3,4}u_{4,1}
\]

\[
b_{1,V} = v_{1,2}v_{2,3}v_{3,4}v_{4,1}.
\]

Here, we can show that \( f_{1,V} \) and \( b_{1,V} \) are given, respectively, by

\[
f_{1,V} = \prod_{j=1}^{M} v_{j,j-1} \quad \text{and} \quad b_{1,V} = \prod_{j=1}^{M} v_{j,j+1}.
\]
(57)

On the other hand, using (55) and (56), (53) and (54) become

\[
\mu_{\pi,1 \rightarrow 2}(n) = (F_{1,T} \circ)^n \mu(0) = f_{1,U} \cdot \sum_{k=0}^{n-1} f_{1,V}^k \mu(0)
\]
(58)

\[
\mu_{\pi,1 \rightarrow M}(n) = (B_{1,T} \circ)^n \mu(0) = b_{1,U} \cdot \sum_{k=0}^{n-1} b_{1,V}^k \mu(0)
\]
(59)
where, from the fact to be proved in the next Lemma that $|f_{i,V}| < 1$ and $|b_{i,V}| < 1$, we have

$$f_{i,V}^n \cdot \mu(0) \to 0, \ b_{i,V}^n \cdot \mu(0) \to 0 \text{ as } n \to \infty.$$  

Therefore, the unique fixed point of the mean in GBP 3G is given by

$$\lim_{n \to \infty} \mu_{\pi,1 \to 2}(n) \to f_{i,U} \cdot \sum_{k=0}^{\infty} f_{i,V}^k = \frac{f_{i,U}}{1 - f_{i,V}} \quad \text{ (60)}$$

$$\lim_{n \to \infty} \mu_{\pi,1 \to M}(n) \to b_{i,U} \cdot \sum_{k=0}^{\infty} b_{i,V}^k = \frac{b_{i,U}}{1 - b_{i,V}} \quad \text{ (61)}$$

**Lemma 3:** $|f_{i,V}| = |\prod_{j=1}^{M} v_{j,j-1}| < 1$ and $|b_{i,V}| = |\prod_{j=1}^{M} v_{j,j+1}| < 1$ for all $i$.

**Proof:** By plugging into (46) into (57), we have for all $i$.

$$|f_{i,V}| = \left| \prod_{j=1}^{M} h_j^H K_{\{j,j-1\}}^{-1} h_j \right| = \left| \prod_{j=1}^{M} h_j^H K_{\{j-1\}}^{-1} h_j \right| = \left| h_j^H K_{\{j\}}^{-1} h_j \right|$$

$$\leq \left| \prod_{j=1}^{M} \text{tr}(h_j h_j^H K_{\{j-1\}}^{-1}) \right| = \left| \prod_{j=1}^{M} h_j^H K_{\{j-1\}}^{-1} h_j \right|$$

$$\leq \prod_{j=1}^{M} \frac{h_j^H K_{\{j-1\}}^{-1} h_j}{1 + h_j^H K_{\{j-1\}}^{-1} h_j} = \prod_{j=1}^{M} \frac{\sigma_{j,j-1}^2}{1 + \sigma_{j,j-1}^2} < 1$$

where, (a) follows by the fact that $a^H b = \text{tr}(ba^H)$ for arbitrary vectors $a$ and $b$, and (b) results from $\text{tr}(AB) \leq \text{tr}(A)\text{tr}(B)$ for arbitrary non-negative definite matrices $A$ and $B$. Also, (c) follows by the following matrix inversion Lemma

$$h_j^H K_{\{j,j+1\}}^{-1} = h_j^H \left( K_{\{j,j+1\}} + h_j h_j^H \right)^{-1} = h_j^H \left( K_{\{j,j+1\}}^{-1} - h_j (1 + h_j h_j^H K_{\{j,j+1\}}^{-1} h_j)^{-1} h_j h_j^H K_{\{j,j+1\}}^{-1} \right)$$

$$= \left( 1 - \frac{h_j h_j^H K_{\{j,j+1\}}^{-1} h_j}{1 + h_j h_j^H K_{\{j,j+1\}}^{-1} h_j} \right) h_j^H K_{\{j,j+1\}}^{-1} = \left( \frac{1}{1 + h_j h_j^H K_{\{j,j+1\}}^{-1} h_j} \right) h_j^H K_{\{j,j+1\}}^{-1}.$$  

For the backward recursion, $|b_{i,V}| < 1$ can also be proved in a similar way.

In (58) and (59), we see that the convergence rate depends on

$$\left| \prod_{j=1}^{M} h_j^H K_{\{j-1\}}^{-1} h_j \right| \leq \prod_{j=1}^{M} \frac{\sigma_{j,j-1}^2}{1 + \sigma_{j,j-1}^2} < 1$$

which is similar to the result in [16]. Note that $h_i^H K_{\{i-1\}}^{-1} h_{i-1}$ reflects the channel correlation between neighboring antennas.  

May 10, 2014
On the other hand, the operations, $F_i$ and $B_i$, are not permutable, such that $F_i \circ F_j \circ \mu$ and $F_j \circ F_i \circ \mu$ may be different, and so are $F_{j,T} \circ \mu$ and $F_{i,T} \circ \mu$ for $j \neq i$. That is, the fixed point for each node may differ from one another.

The following two Lemmas show that the fixed points in (60) and (61) are both equal to the MMSE estimate in (31).

**Lemma 4:** In the forward recursion, $\mu_{\pi,i \rightarrow i+1}(n)$ is the linear MMSE estimates of $x_{i+1}$ provided that the previous message, $\mu_{\pi,i-1 \rightarrow i}(n)$, is the linear MMSE estimates of $x_i$. Likewise, in the backward recursion, $\mu_{\pi,i \rightarrow i-1}(n)$, is the linear MMSE estimates of $x_{i-1}$ provided that $\mu_{\pi,i+1 \rightarrow i}(n)$ is the linear MMSE estimates of $x_i$.

**Proof:** With $c_j = K^{-1}h_j$, the linear MMSE estimate of $x_i$ is given by $h_i^H K^{-1}y$. And, hence, the proof is to show from (41) and (42) that

$$h_{i+1}^H K^{-1}y = F_i \circ (h_i^H K^{-1}y) = u_{i+1,i} + \nu_{i+1,i} \cdot (h_i^H K^{-1}y)$$

$$h_{i-1}^H K^{-1}y = B_i \circ (h_i^H K^{-1}y) = u_{i-1,i} + \nu_{i-1,i} \cdot (h_i^H K^{-1}y)$$

where $u_{j,i}$ and $\nu_{j,i}$ are given by (45) and (46). Plugging these into the right hand side of (62) for the forward recursion, we finally have

$$u_{i+1,i} + \nu_{i+1,i} \cdot (h_i^H K^{-1}y) = h_{i+1}^H K^{-1}y - h_{i+1}^H K_{(i)}^{-1} h_i^H K^{-1} y = h_{i+1}^H \left( K_{(i)}^{-1} - K_{(i)}^{-1} h_i^H K^{-1} \right) y$$

$$= h_{i+1}^H \left( K_{(i)}^{-1} - K_{(i)}^{-1} \frac{h_i^H K_{(i)}^{-1} h_i}{1 + h_i^H K_{(i)}^{-1} h_i} \right) y = h_{i+1}^H K^{-1}y.$$

Similarly, for the backward recursion, we obtain

$$u_{i-1,i} + \nu_{i-1,i} \cdot (h_i^H K^{-1}y) = h_{i-1}^H K^{-1}y.$$

**Lemma 5:** Both the fixed points, $\frac{f_{i,v}}{1-f_{j,v}}$ and $\frac{b_{i,v}}{1-b_{j,v}}$, are equal to the MMSE estimate of $x_j$, i.e., $h_j^H K^{-1}y$.

**Proof:** Without loss of generality, let us consider the first data symbol, $x_1$. Starting from $h_2^H K^{-1}y = F_1 \circ (h_1^H K^{-1}y)$, we can successively apply the operations, $F_2 \circ, F_3 \circ, ..., F_M \circ$ to finally obtain

$$F_M \circ F_{M-1} \circ \cdots \circ F_2 \circ F_1 \circ (h_1^H K^{-1}y) = F_{1,T} \circ (h_1^H K^{-1}y)$$

$$= f_{1,U} + f_{1,V} \cdot (h_1^H K^{-1}y)$$

$$= (h_1^H K^{-1}y)$$
where the first and second equality are obtained by the definition of $F_{1,T}$ and $(f_{1,U}, f_{1,V})$, respectively, and the last equality is from Lemma 4, i.e., $H_j^T K^{-1} y = F_M \circ (h_j^T K^{-1} y)$. From the last equality, we obtain $\frac{f_{j,U}}{f_{j,V}} = h_j^T K^{-1} y$ and, in a similar way, we also can prove that $\frac{b_{j,U}}{b_{j,V}} = h_j^T K^{-1} y$.

Proof: The proof of Theorem 1 is now obvious from the above lemmas, i.e., from Lemma 2 the mean in the Gaussian BP over the ring-type graphical model converges to a unique fixed point and Lemma 5 shows that the fixed point of the mean is equal to the linear MMSE estimates in (31).

Note that the Theorem 1 holds for any channel matrices if noise variance is not zero since, for $\sigma^2 > 0$, the covariance matrices, $K_{(i,j)}$'s in (16) are always invertible so that there certainly exist the MMSE estimator in (17) and the translation functions in (24) for all pair of $(i,j)$.

Since the message-update rule for the variance in (47) and (48) have the same form as in (41) and (42), we can also prove the convergence of the variance in GBP 3G, which can be summarized by the following Lemma.

Lemma 6: For an arbitrary initial value $\sigma^2(0)$, both the forward and backward recursions for the variance in (33) and (35) converge, respectively, to a unique fixed point.

The proof is similar to that of Lemma 3. Unfortunately, however, the fixed point is not necessarily correct. That is, it may not equal the MMSE in (32), as also confirmed in [30]. In [29], the convergence property of the BP over such ring-type graph was shown to be optimal for binary input. For Gaussian input, however, it is optimal only in the mean, i.e., the mean converges to the fixed point that is equal to the MMSE estimates, $\hat{x}_j$ in (31), and we cannot say so in a strict sense since the fixed point of the variance is not equal to the MSE in (32), the MAP estimates on the variance.

It will also be worth comparing GBP 2G and 3G proposed in this paper and the Gaussian BP in [17], [19], and [16], all of which are based on the direct decomposition of Gaussian PDF, and, as noticed in [19], are the same algorithm. The comparison can be made in several aspects, i.e., in complexity and convergence. In complexity, the Gaussian BP in [17], [19], and [16] is much simpler than GBP 2G and 3G proposed here. Note that (1) Gaussian BP in [17], [19], and [16] does not require preprocessing while GBP 2G and 3G in this paper do and (2) the complexity of the post iteration for the former is obviously the same as that of GBP 2G since they utilize the same graphical model, even though the post iteration of GBP 3G is a little bit less complex than GBP 2G. Based on this, the overall complexity of the proposed GBP 2G and 3G is certainly more complex than those in [17], [19], and [16]. Now, let us consider their convergence. Basically, GBP 3G proposed in this paper and the Gaussian BP in [17], [19], and [16] results in an MMSE solution (in mean) if they converge, as proved here for GBP 3G and in [16] for Gaussian BP with the direct decomposition. This means that, once converged, they will perform
the same. Unfortunately, the convergence of the Gaussian BP in [17], [19], and [16] seems not to be assured while GBP 3G surely converges.

V. SIMULATION RESULTS

In this section, we present simulation results for the iterative algorithms with and without channel coding. For channel coding, we used DVB-S2 LDPC code of rates 3/4 and length 64800 [32]. The performances of ML, MMSE, and the bi-diagonalization approach in [9] are also evaluated as references.

In the transmitter, a block (48600 bits) of random information bits are generated first and then coded using the LDPC encoder and then interleaved with a random interleaver and modulated into a sequence of $2^m$-ary symbols. The symbol sequence is then divided into sub-blocks of $M$ symbols, each of which is fed to a transmit antenna, where $M$ corresponds to the number of transmit antennas. At the receiver, the sequence of received vectors is passed to a MIMO detector, which generates the estimates of symbol likelihoods and LLRs for each coded bit. The LLR is then de-interleaved and decoded by using a generic LDPC decoder.$^2$ Note that no 'turbo principle' is applied since it is not our focus in this paper. This means that the LDPC decoding begins only after the inner iteration in MIMO detector is finished. Regarding the MIMO channel, we generated, for each transmitted data vector, an independent and identically distributed (i.i.d.) MIMO channel matrix, of which each element is also an i.i.d. complex Gaussian random variable with mean 0 and variance 1. The resulting channel can be regarded as a fully interleaved frequency selective MIMO channel that can be seen on top of the orthogonal frequency division multiplexing (OFDM), especially for those channels where the transmission bandwidth is much larger than the channel coherence bandwidth.

Fig. 3 shows a comparison of bit error rate performance as a function of signal-to-noise ratio (SNR) ($1/\sigma^2$) for ML, BP1 in [18], MMSE, the bi-diagonalization approach in [9], and the proposed BP-based detector with the fully-connected and ring type pair-wise model. We use a $4 \times 4$ antenna configuration and QPSK modulation. We could confirm from Fig. 3 that the pair-wise MRF-based detector performs as well as the ML with soft decisions (i.e., using (3) and (4)). The SNR gap between the proposed scheme and the ML is shown to be around 0.1 and 0.3, respectively.

Fig. 4 shows a comparison of bit error rate performance without channel coding as a function of signal-to-noise ratio (SNR) ($1/\sigma^2$) for ML, MMSE, and the proposed detector of fully-connected and ring type

$^2$In the transmitter and the receiver, the interleaving/de-interleaving and channel coding/decoding is used if channel coding is applied.
model. We used the same antenna configuration and modulation size, but without channel coding. As shown in the figure, the tendency in the relative performance looks similar to that with channel coding.

It is also worth comparing the performance of BP2 and BP3 with the one in [27] (Table-I), where the pair-wise MRF obtained by the direct decomposition of Gaussian PDF is used. The performance comparison is shown in Fig. 5 for BPSK modulation and the same LDPC coding. As shown in the figure, the performance of the one in [27] is almost the same as that of BP2. We set the number of iterations to 4 for BP2 and BP3 and 6 for the one in [27]. We also tried to obtain the results for 4PAM. Unfortunately, however, the algorithm in [27] failed to work for 4-PAM and this is one of the advantages of using the proposed scheme over the existing (fully-connected) MRF based MIMO detection.

Fig. 6 shows the BER performance for a $6 \times 6$ antenna configuration with the same modulation and channel coding. The SNR gap between the proposed scheme and the ML is now approximately 0.75 dB for the fully connected pair-wise graph and 1 dB for the ring-type one, respectively. Although the performance degradation compared to the ML is larger than for a $4 \times 4$ antenna configuration, the SNR gain over the MMSE detector is around 3.5 dB.

In Fig. 7, BER performance with higher modulation order (16-QAM) is shown. We used a $4 \times 4$ antenna configuration and the same channel coding. Here, the SNR gap between the proposed method and the ML is shown to be around 1 dB for the BP2 over the fully connected pair-wise graph and 0.7 dB for BP3 over the ring-type, respectively. Note that the performance of BP2 over the fully connected pair-wise graph is now worse than that of BP3 over the ring-type. Here, we set the number of iterations of BP2 and BP3 to four and six. One possible reason for why the fully-connected graph perform worse than the ring-type for higher order QAM can be inferred from the convergence behavior as shown in Fig.9, where it is shown that the convergence for fully-connected pair-wise graph is stuck at three or four iterations and the BER is increased sharply with more number of iterations, while, for the ring-type, it converges steadily.

In Figs. 3 to 7, the number of iterations was set based on the simulation results in Figs. 8 and 9, which we performed with different number of antennas and modulation size, to give insights into how many iterations are required for a satisfactory performance. As shown in the simulation results, the number of iterations required for convergence depends on the modulation sizes, but not much on the number

3 The reason we consider here only one-dimensional constellation like BPSK or 4-PAM is that the algorithm in [27] is applicable only to those real constellation and we just wanted to use the algorithm as is since any modification may cause unexpected results.
of antennas. Specifically, for BP3, we can say that we need more number of iterations for the eventual convergence with higher modulation size. For BP2, the convergence behavior with different number of antennas looks similar to that of BP3 (specifically for QPSK), while it is quite different from that of BP3 with different modulation sizes. Specifically speaking, the performance of BP2 over the fully connected graph does not get better with more than 3 or 4 iterations. Rather, it is degraded especially for higher order modulation. In BP3 over the ring-type graph, however, no degradation has been observed with more iterations. As mentioned previously, the condition for sure convergence in loopy graph is still an open problem. And the difference in the convergence behavior of BP2 and BP3 can only be explained by the note in [21], i.e., in densely connected graph, the messages may circulate along the short loops preventing the eventual convergence. Fig. 2(a) of the fully connected pair-wise graph is more densely connected than Fig. 2(b) of the ring-type pair-wise graph. Although the message will propagate faster in densely connected graph than in sparsely connected graph, resulting in faster convergence, the message circulation may prevents the eventual convergence with more iterations.

Another point we need to note is that, in BP3, one can allow a slight performance degradation for a large computational saving. Certainly, as shown in Fig. 9 and implicated in Fig. 7 at least 10 iterations is needed for eventual convergence for 16QAM. However, comparing the required SNR for, say, $10^{-4}$ BER, the difference between 6 and 12 iterations is less than 0.1 dB while, in computational burden, 12 iterations is twice that of 6.

Fig. 10 shows the convergence behavior of the Gaussian BP discussed in Section IV. We plotted the bit error rate performance of the Gaussian BP over the fully-connected and ring-type pair-wise graph, respectively, with various numbers of iterations. As can be seen in the figure, both GBP 2G and 3G converge to the performance of linear MMSE detector, though it requires many more iterations than those of BP2 and BP3. The only difference between GBP 2G and 3G is the rate of convergence. On the other hand, in the high SNR region, the performance appears to worsen with higher SNR. However, it should be noted that with higher SNR eventual convergence simply requires more iterations.

VI. CONCLUSIONS

In this paper, low complexity, iterative MIMO detection algorithms were derived as a message passing over the pair-wise bipartite graphs with the translation functions that are obtained by marginalizing the posterior joint probability density under the Gaussian input assumption. We investigated two models, the fully-connected and ring-type pair-wise graph. The latter is shown to be an extension of the previous work in [9], [10]. The two pair-wise graphical models are rather sparse in the sense that the number of
edges connected to an observation node, i.e., edge degree, is only two and, thus, the message passing becomes much easier than that over the fully connected bipartite graph.

We also investigated the proposed algorithm under Gaussian input assumption. It was shown that, for the Gaussian BP over the ring-type pair-wise graph, the mean converges to the linear MMSE estimates, even though the variance converges to a different value from the MMSE obtained by MAP estimation. These results are in line with those in [16], [17], [30], [31]. Gaussian BP over the fully-connected pair-wise graph shows a faster convergence rate than Gaussian BP over the ring-type graph.

As proved in this paper, the convergence of the Gaussian BP over the ring-type graph is guaranteed. This does not, however, appear to be the case for non-Gaussian message. The performance of BP 2 for non-Gaussian case degrade with more than four iterations. This phenomenon might stem from the short cycles in their graphical model and may be avoided by utilizing “global iteration” between MIMO detection and channel decoding. That is, by employing an appropriate channel code and interleaver, message circulation along local short cycles can be broken up not only for steady convergence but also for better performance. We leave this for our future work.

APPENDIX

DETAILED DERIVATIONS OF (44) AND THE GAUSSIAN BP

To derive (24) and the Gaussian BP rule, (33)-(40), we use the properties of the Gaussian PDF in [21], some of which are as follows

1) $CN(x; \mu, \sigma^2) = CN(\mu; x, \sigma^2) = CN(x - \mu; 0, \sigma^2) = CN(\mu - x; 0, \sigma^2)$

2) $CN(ax + b; \mu, \sigma^2) = CN(x; \frac{\mu - b}{a}, |a|^2)$

3) $CN(x; \mu_1, \sigma_1^2) \cdot CN(x; \mu_2, \sigma_2^2) = CN\left(x; \frac{\sigma_1^{-2}\mu_1 + \sigma_2^{-2}\mu_2}{\sigma_1^{-2} + \sigma_2^{-2}}, \frac{1}{\sigma_1^{-2} + \sigma_2^{-2}}\right) \cdot CN(\mu_1; \mu_2, \sigma_1^2 + \sigma_2^2)$

4) $\int CN(x; \mu_1, \sigma_1^2) \cdot CN(x; \mu_2, \sigma_2^2) \cdot dx = CN(\mu_1; \mu_2, \sigma_1^2 + \sigma_2^2)$.

Using these, (24) is obtained by direct computation as follows.

$$
\tilde{p}(x_j|x_i, y_{ji}^{'}) = \frac{CN\left(y_{ji}^{'}; a_{ji|i,i}x_i + a_{ji|i,j}x_j, \sigma_{ji}^2\right) \cdot CN(x_j; 0, 1)}{CN\left(y_{ji}^{'}; a_{ji|i,i}x_i, \sigma_{ji}^2 + |a_{ji|i,j}|^2\right)}
\quad \frac{CN\left(x_j; 0, 1\right)}{CN\left(y_{ji}^{'}; a_{ji|i,i}x_i, \sigma_{ji}^2 + |a_{ji|i,j}|^2\right)}
\quad \frac{CN\left(y_{ji}^{'}; a_{ji|i,i}x_i + a_{ji|i,j}x_j, \sigma_{ji}^2\right)}{CN\left(y_{ji}^{'}; a_{ji|i,i}x_i, \sigma_{ji}^2 + |a_{ji|i,j}|^2\right)}
$$
For the message translation, we first rewrite (63) as

\[
\tilde{p}(x_j | x_i, y'_j | x_i) = \mathcal{CN} \left( x_j; \frac{a^*_{j|i,j}}{\sigma_{j|i,j}^2 + |a_{j|i,j}|^2}, \frac{\sigma^2_{j|i,j}}{\sigma_{j|i,j}^2 + |a_{j|i,j}|^2} \right) \cdot \mathcal{CN} \left( y'_j | x_i, y'_j; 0, |a_{j|i,j}|^2 + \sigma^2_{j|i,j} \right)
\]

\[
= \mathcal{CN} \left( x_j; \frac{a^*_{j|i,j}}{\sigma_{j|i,j}^2 + |a_{j|i,j}|^2}, \frac{\sigma^2_{j|i,j}}{\sigma_{j|i,j}^2 + |a_{j|i,j}|^2} \right) \cdot \mathcal{CN} \left( y'_j | x_i; 0, |a_{j|i,j}|^2 + \sigma^2_{j|i,j} \right)
\]

Now, we derive the message update rule of the Gaussian BP. To this end, we divide the message update rule in BP2, (26), into two steps, i.e., the extrinsic information computation, \( \lambda_{i\rightarrow j}(x_i) = \prod_{k \in V(i) \setminus j} \pi_{k\rightarrow i}(x_i) \), and the message translation step, \( \pi_{i\rightarrow j}(x_j) = \alpha \sum_{x_i \in \mathcal{X}} \tilde{p}(x_j | x_i, y) \lambda_{i\rightarrow j}(x_i) \). Assuming the Gaussian messages, \( \pi_{k\rightarrow i}(x_i) = \mathcal{CN}(x_i; \mu_{\pi,k\rightarrow i}, \sigma^2_{\pi,k\rightarrow i}) \), the former is given by

\[
\lambda_{i\rightarrow j}(x_i) = \prod_{k \in V(i) \setminus j} \pi_{k\rightarrow i}(x_i) = \prod_{k \in V(i) \setminus j} \mathcal{CN}(x_i; \mu_{\pi,k\rightarrow i}, \sigma^2_{\pi,k\rightarrow i})
\]

\[
\propto \mathcal{CN} \left( x_i; \frac{\sum_{k \in V(i) \setminus j} \sigma^{-2}_{\pi,k\rightarrow i} \mu_{\pi,k\rightarrow i}}{\sum_{k \in V(i) \setminus j} \sigma^{-2}_{\pi,k\rightarrow i}}, \left( \sum_{k \in V(i) \setminus j} \sigma^{-2}_{\pi,k\rightarrow i} \right)^{-1} \right) = \mathcal{CN} \left( x_i; \mu_{\lambda_{i\rightarrow j}}, \sigma^2_{\lambda_{i\rightarrow j}} \right) \quad (64)
\]

For the message translation, we first rewrite (63) as

\[
\mathcal{CN} \left( x_j, \frac{a^*_{j|i,j}}{\sigma_{j|i,j}^2 + |a_{j|i,j}|^2}, \frac{\sigma^2_{j|i,j}}{\sigma_{j|i,j}^2 + |a_{j|i,j}|^2} \right) \cdot \mathcal{CN} \left( y'_j | x_i, a_{j|i,j}^2, \sigma^2_{j|i,j} \right)
\]

\[
= \mathcal{CN} \left( x_j; \frac{a^*_{j|i,j}}{\sigma_{j|i,j}^2 + |a_{j|i,j}|^2}, \frac{\sigma^2_{j|i,j}}{\sigma_{j|i,j}^2 + |a_{j|i,j}|^2} \right) \cdot \mathcal{CN} \left( y'_j | x_i; 0, |a_{j|i,j}|^2 + \sigma^2_{j|i,j} \right)
\]
Then, by plugging (64) and (65) into (26) and changing the summation into integral\(^4\) we have

\[
\pi_{i \rightarrow j}(x_j) = \int x_i \, p(x_j | x_i, y_{ji}) \cdot \lambda_{i \rightarrow j}(x_i) \cdot dx_i
\]

\[
= \int x_i \mathcal{N} \left( x_i; \frac{1}{\alpha_{j,i}} \left( y_{ji} - \frac{\sigma_{j|i}^2 + |a_{j|i,j}|^2}{\alpha_{j|i,j}^2} x_j \right), \frac{\sigma_{j|i}^2 + |a_{j|i,j}|^2}{|a_{j|i,j}|^2} \right) \cdot \mathcal{N} \left( x_i; \mu_{\lambda,i \rightarrow j}, \sigma_{\lambda,i \rightarrow j}^2 \right) \cdot dx_i
\]

\[
= \mathcal{N} \left( y_{ji} - \frac{\sigma_{j|i}^2 + |a_{j|i,j}|^2}{\alpha_{j|i,j}^2} x_j - a_{j|i,j} \mu_{\lambda,i \rightarrow j} \right) \cdot \mathcal{N} \left( x_j; \frac{\sigma_{j|i}^2 + |a_{j|i,j}|^2}{\alpha_{j|i,j}^2} \right) \cdot \mathcal{N} \left( x_j; \mu_{\pi,i \rightarrow j}, \sigma_{\pi,i \rightarrow j}^2 \right) \cdot \frac{\sigma_{j|i}^2 + |a_{j|i,j}|^2}{|a_{j|i,j}|^2} \cdot \frac{\sigma_{\lambda,i \rightarrow j}^2}{|a_{j|i,j}|^2}
\]  

(66)

\[
= \mathcal{N} \left( x_j; \mu_{\pi,i \rightarrow j}, \sigma_{\pi,i \rightarrow j}^2 \right) \cdot \frac{\sigma_{j|i}^2 + |a_{j|i,j}|^2}{|a_{j|i,j}|^2} \cdot \frac{\sigma_{\lambda,i \rightarrow j}^2}{|a_{j|i,j}|^2}
\]  

(67)

By comparing the mean and variance in (66) and (67), we obtain the message passing rules of (33) and (34), respectively. The belief in (33) and (36) can be obtained similarly to the derivation in (64).

REFERENCES

[1] U. Fincke and M. Poste, “Improved methods for calculating vectors of short length in a lattice, including a complexity analysis,” Math. Comput., vol. 44, pp. 463–471, April 1985.
[2] H. Vikalo and B. Hassibi, “Modified Fincke-Poste algorithm for low-complexity iterative decoding over multiple antenna channel,” in Proc. IEEE Int. Symp. Info. Th., July 2002, p. 390.
[3] B. M. Hochwald and S. ten Brink, “Achieving near-capacity on multiple-antenna channel,” IEEE Trans. Comm., vol. 51, no. 3, pp. 389–399, March 2003.
[4] D. Wubben, R. Boehnke, V. Kuehn, and K. Kameyer, “MMSE extension of V-BLAST based on sorted QR decomposition,” in Proc. IEEE Veh. Technol. Conf., Oct. 2003.
[5] C. Studer, A. Burg, and H. Boleskei, “Soft-output sphere decoding: Algorithms and VLSI implementation,” IEEE Jour. Select. Areas in Comm., vol. 26, no. 2, pp. 290–300, Feb. 2008.
[6] K. Higuchi, H. Kawai, H. Taoka, N. Maeda, and M. Sawahashi, “Adaptive selection of surviving symbol replica candidates for quasi-maximum likelihood detection using M-algorithm with QR-decomposition for OFDM MIMO multiplexing,” IEICE Trans. Comm., vol. e92-b, 2009.
[7] W. Jiang, X. Yu, and Y. Li, “Bi-truncation for simplified MIMO signal detection,” in Proc. IEEE Glob. Telecom. Conf., Nov. 29–Dec. 3 2004, pp. 401–405.
[8] W. Jiang, Y. Li, and X. Yu, “Truncation for low-complexity MIMO signal detection,” IEEE Trans. Info. Th., vol. 53, no. 4, pp. 1564–1571, April 2007.
[9] S. Yoon and S. Lee, “A detection algorithm for multi-input multi-output (MIMO) transmission using poly-diagonalization and trellis decoding,” IEEE Jour. Select. Areas in Comm., vol. 26, no. 8, pp. 993–1002, Aug. 2008.
[10] S. Yoon, “Asymmetrically optimized poly-diagonalization for low complexity MIMO detection,” IET Elec. Letters, vol. 46, no. 17, pp. 1226–1228, Aug. 2010.
[11] G. D. Forney, “The forward-backward algorithm,” in Proc. of Allerton Conf. on Comm. Control and Comp., Sep. 1996.
[12] J. B. Anderson and S. M. Hladik, “Tailbiting MAP decoders,” IEEE Jour. Select. Areas in Comm., vol. 16, no. 2, pp. 297–302, Feb. 1998.
[13] D. D. Falconer and F. R. Magee, “Adaptive channel memory truncation for maximum likelihood sequence estimation,” The Bell Sys. Tech. Jour., vol. 52, no. 9, pp. 1541–1562, Sep. 1973.
[14] Y. Kabashima, “A CDMA multiuser detection algorithm on the basis of belief propagation,” J. Phys. A., pp. 11111–11121, 2003.

\(^4\)Since the input is now continuous Gaussian random variable, we need to change the summation in (24) into integral.
### Table I

An approximate number of operations required for detection in a single $M \times M$ MIMO channel (modulation size of $2^m$).

| Detector | Linear preprocessing | Post detection | $M=6$, $m=2$, $\nu_1=4$, $\nu_2=6$ | $M=4$, $m=4$, $\nu_1=4$, $\nu_2=6$ |
|----------|----------------------|----------------|-----------------------------------|-----------------------------------|
| MMSE     | $24M^3 + 18M^2 + 2M$ | $6M \cdot 2^m$ | 5,988                             | 2,216                             |
| ML       | 0                    | $2^m M \cdot (8M^3 + 9M)$ | 1,454,080                        | 11,337,728                        |
| BP2      | $16M^4 + 60M^3 + 437M^2 - 486M$ | $[2^m \cdot (2\nu_1 + 21) + 2^m \nu_1] \cdot M (M - 1)$ | 61,032                           | 102,648                           |
| BP3      | $56M^5 + 113M^4 + 914M^2$ | $2^m \cdot (2\nu_2 + 21) \cdot 2M$ | 27,984                           | 76,632                            |

Parameters $\nu_1$ and $\nu_2$ are the number of iterations for the BP 2 and BP 3, respectively.

---

[15] T. Tanaka and M. Okada, “Approximate belief propagation, density evolution, and neurodynamics for CDMA multiuser detection,” *IEEE Trans. Info. Th.*, vol. 51, no. 2, pp. 700–706, Feb. 2005.

[16] A. Montanari, B. Prabhakar, and D. Tse, “Belief propagation based multi-user detection,” in *Proc. of Allerton Conf. on Comm. Control and Comp.*, Sep. 2005.

[17] D. Bickson, O. Shental, P. H. Siegel, J. K. Wolf, and D. Dolev, “Linear detection via belief propagation,” in *Proc. of Allerton Conf. on Comm. Control and Comp.*, Sep. 2007.

[18] J. Hu and T. M. Duman, “Graph-based detector for BLAST architecture,” in *Proc. IEEE Int. Conf. on Comm.*, June 2007, pp. 1018–1023.

[19] D. Bickson, O. Shental, P. H. Siegel, J. K. Wolf, and D. Dolev, “Gaussian belief propagation based multiuser detection,” in *Proc. IEEE Int. Symp. Info. Th.*, July 2008, pp. 1878–1882.

[20] J. Goldberger and A. Leshem, “A Gaussian tree approximation for integer least-squares,” in *Neural Info. Proc. Systems (NIPS)*, 2009.

[21] J. Pearl, *Probabilistic reasoning in intelligent systems: Networks of plausible inference*, Morgan Kaufmann, 1987.

[22] F. Kschischang, B. Frey, and H. A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Trans. Info. Th.*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[23] J. Goldberger and A. Leshem, “MIMO decoding based on stochastic reconstruction from multiple projections,” in *Proc. IEEE Int. Conf. Acoust.*, *Speech and Sig. Proc.*, Apr. 2009, pp. 2457–2460.

[24] M. Suneel, P. Som, A. Chockalingam, and B. S. Rajan, “Belief propagation based decoding of large non-orthogonal STBCs,” in *Proc. IEEE Int. Symp. Info. Th.*, June 28–July 3 2009, pp. 2003–2007.

[25] P. Som, T. Datta, A. Chockalingam, and B. S. Rajan, “Improved large-MIMO detection based on damped belief propagation,” in *Proc. of IEEE Info. Th. Workshop*, Jan. 2010.

[26] J. S. Yedidia, W. T. Freeman, and Y. Weiss, “Understanding belief propagation and its generalizations,” in *Proc. Int. Joint. Conf. on Artificial Intelligence*, 2001.

[27] S. Yang, T. Lv, R. G. Mauner, and L. Hanzo, “Unified bit-based probabilistic data association aided mimo detection for high-order qam constellations,” *IEEE Trans. on Veh. Technol.*, vol. 60, no. 3, pp. 981–991, Mar. 2011.

[28] R. Koetter, A. C. Singer, and M. Tchler, “Turbo equalization,” *IEEE Sig. Proc. Mag.*, vol. 21, no. 1, pp. 67–80, Jan. 2004.

[29] S. M. Aji, G. B. Horn, and R. J. McEliece, “Iterative decoding on graphs with a single cycle,” in *Proc. IEEE Int. Symp. Info. Th.*, Aug. 1998, p. 276.

[30] Y. Weiss and W. T. Freeman, “Correctness of belief propagation in gaussian graphical models of arbitrary topology,” *Report No UCB/CSRD-99-1046, UC Berkeley*, 1999.

[31] P. Rusmevichientong and B. V. Roy, “An analysis of belief propagation on the turbo decoding graph with Gaussian densities,” *IEEE Trans. Info. Th.*, vol. 47, no. 2, pp. 745–765, Feb. 2001.

[32] Digital Video Broadcasting (DVB), “Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications (DVB-S2),” *ETSI EN 302207*, April 2009, ver. 1.2.1.
Fig. 1. Bipartite graphs for a $4 \times 4$ MIMO channel. The circles are variable nodes corresponding to a data symbol and the boxes labeled by $y$ and $y_j$ are observation nodes corresponding to the received signal.

Fig. 2. The bipartite graph for (a) fully-connected pair-wise model and (b) ring-type pair-wise model, respectively, for a $4 \times N$ MIMO channel.

Fig. 3. A comparison of bit error rate performance of MMSE, MAP and the proposed detectors as a function of SNR ($1/\sigma^2$); $4 \times 4$ antenna configuration, QPSK modulation with DVB-S2 LDPC code of rate 3/4 (length 64800).
Fig. 4. A comparison of bit error rate performance of MMSE, MAP, and the proposed detectors as a function of SNR \((1/\sigma^2)\); 4 × 4 antenna configuration, QPSK modulation, no channel coding.

Fig. 5. A comparison of bit error rate performances of MMSE, the proposed detectors and the detector in [25] with a damping factor 0.45.
Fig. 6. A comparison of bit error rate performance of MMSE, MAP and the proposed detectors as a function of SNR \((1/\sigma^2)\); 6 × 6 antenna configuration, QPSK modulation with DVB-S2 LDPC code of rate 3/4 (length 64800).

Fig. 7. A comparison of bit error rate performance of MMSE, MAP, and the proposed detectors as a function of SNR \((1/\sigma^2)\); 4 × 4 antenna configuration, 16QAM modulation with DVB-S2 LDPC code of rate 3/4 (length 64800).
Fig. 8. Convergence property of the proposed algorithm with the number of antennas $M = 4, 6$ and $8$; QPSK modulation, DVB-S2 LDPC code of rate $3/4$ (length 64800).

Fig. 9. Convergence property of the proposed algorithm with modulation size of $2^m$ for $m = 2, 3, 4$ and $5$; $4 \times 4$ antenna configuration, DVB-S2 LDPC code of rate $3/4$ (length 64800)
Fig. 10. Bit error rate performance of the Gaussian BP over the fully-connected and ring-type pair-wise graph, respectively; 4 × 4 antenna configuration, QPSK modulation, no channel coding.