

Code-Aided Estimation and Detection on Time-Varying Correlated MIMO Channels: A Factor Graph Approach

Frederik Simoens and Marc Moeneclaey

DIGCOM Research Group, Department of Telecommunications and Information Processing, Ghent University, Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium

Received 27 May 2005; Revised 20 March 2006; Accepted 7 April 2006

This paper concerns channel tracking in a multiantenna context for correlated flat-fading channels obeying a Gauss-Markov model. It is known that data-aided tracking of fast-fading channels requires a lot of pilot symbols in order to achieve sufficient accuracy, and hence decreases the spectral efficiency. To overcome this problem, we design a *code-aided* estimation scheme which exploits information from both the pilot symbols and the unknown *coded* data symbols. The algorithm is derived based on a factor graph representation of the system and application of the sum-product algorithm. The sum-product algorithm reveals how soft information from the decoder should be exploited for the purpose of estimation and how the information bits can be detected. Simulation results illustrate the effectiveness of our approach.

Copyright © 2006 Hindawi Publishing Corporation. All rights reserved.

1. INTRODUCTION

Communication over time-varying fading channels has been studied intensively during the last decade [1–3]. The introduction of turbo coding and channel interleaving gave rise to astounding performance results. In particular, channel interleaving [2–4] combined with coding can combat the adverse conditions, originating from the time varying nature of the channel, by spreading channel errors, caused by deep fades, over the full length of the frame. When further applying multiple transmit and receive antennas (resulting in a so-called MIMO transmission), high data rates and high diversity gains can be achieved simultaneously. However, in order to fully exploit these advantages, accurate knowledge of the channel state is required. Although a lot of research effort has been focused on this subject [5–11], estimation and tracking of fading channels remains a major challenge.

The Kalman filter/smoothing [12] is a powerful tool to obtain the minimum mean-squared error (MMSE) estimate of a parameter varying according to a discrete-time linear model. This technique is particularly convenient for *pilot-assisted* estimation of a time-varying channel [7–9]. However, estimating a time-varying channel in the presence of *unknown data* symbols is not possible by straightforward Kalman filtering/smoothing. This has led to the introduction of several modified approaches for the estimation of a time-varying channel (see [7, 10, 11] and references therein). The problem related to the unknown symbols was circum-

vented by introducing an iterative decision-directed structure.

Several years ago, it has been recognized that Kalman filtering can be interpreted as a message-passing algorithm (the sum-product (SP) algorithm) on a factor graph [13]. Ever since, the SP algorithm has been applied to a various number of estimation problems [14–17], capitalizing on the concepts from [18, 19]: the algorithm iterates between decoding and estimation, whereby the estimator accepts information from the decoder about the unknown data symbols. In [14] the estimation of a linear dynamical noise process is considered. In [15], the authors consider the tracking of a time-varying complex gain for single-input single-output (SISO) channels. A similar problem is considered in [16, 17], namely, phase noise estimation. As elaborated upon, the SP algorithm runs into practical difficulties in the presence of unknown data symbols. The problems are alleviated by representing and computing messages in an efficient fashion.

In this paper, we apply these ideas to the factor graph of a flat-fading *correlated* multiple-input multiple-output (MIMO) system with bit-interleaved coded modulation (BICM). The temporal behavior of our channel is modeled as a first-order autoregressive model [11, 20], whereas the spatial correlation abides by the findings from [21, 22]. As we will show, the complexity of the SP algorithm, in its exact form, is exponential in the block length. To overcome this problem, we introduce a suitable approximation. The resulting code-aided estimator exploits information about

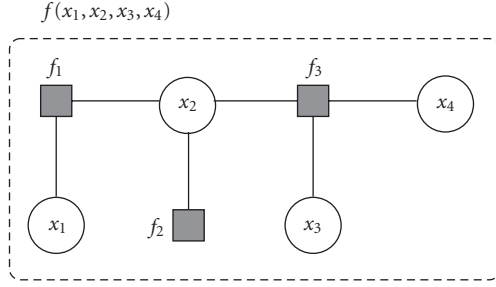


FIGURE 1: Example factor graph.

the received signal as well as soft information from the decoder in a systematic manner.

This paper is organized as follows. A short introduction on factor graphs is given in Section 2. The system model is described in Section 3. This is followed by a factor graph representation of the receiver and derivation of the SP algorithm on this graph. In Section 5 the practical estimation algorithm is derived. Before conclusions are drawn, the performance of the proposed algorithm is illustrated in Section 6.

2. FACTOR GRAPHS AND THE SUM-PRODUCT ALGORITHM

In this section, we briefly outline the basic ideas behind factor graphs and the sum-product algorithm. We refer to [13, 23] for a more profound analysis.

Factor graph

A factor graph is an elegant method to express the factorization of a function depending on many variables. As an example, consider the factor graph depicted in Figure 1. The graph represents the factorization of the following function:

$$f(x_1, x_2, x_3, x_4) = f_1(x_1, x_2) f_2(x_2) f_3(x_2, x_3, x_4). \quad (1)$$

We observe two types of nodes: function nodes (indicated by squares) and variable nodes (indicated by circles). When a function depends on some variable, there is a connection between the corresponding function node and variable node.

It is interesting to note that any type of function is suitable for a factor graph representation, however, throughout this paper, we will only consider the factorization of *probability density functions*.

Sum-product algorithm

In addition to visualizing the factorization of a (complicated) function, factor graphs also allow us to compute the marginals of that function in a systematic manner. The marginal of a function $f(x_1, \dots, x_N)$ with respect to the variable x_i is defined as

$$g_i(x_i) = \sum_{\sim\{x_i\}} f(x_1, \dots, x_N), \quad (2)$$

where $\sim\{x_i\}$ represents the set containing all variables, except x_i . If (some of the) variables are continuous, the summations with respect to these variables in (2) should be replaced by integrals.

The SP algorithm is a message-passing algorithm, that provides an efficient way to compute the marginals (2). Messages are computed in the different nodes based on the incoming messages on these nodes. Depending on the type of node, function node or variable node, the outgoing messages are computed according to

$$\text{variable node: } \mu_{x_i \rightarrow f_m}(x_i) = \prod_{n \neq m} \mu_{f_n \rightarrow x_i}(x_i), \quad (3)$$

$$\text{function node: } \mu_{f_m \rightarrow x_i}(x_i) = \sum_{\sim\{x_i\}} f_m(X_m) \prod_{j \neq i} \mu_{x_j \rightarrow f_m}(x_j). \quad (4)$$

The message-passing algorithm is initiated at nodes of degree 1, that is, nodes which are connected to one neighboring node only. Messages travel on the graph until all ingoing and outgoing messages of all nodes have been computed. If the graph contains no cycles, the algorithm is assured to converge, and the marginal with respect to a certain variable is obtained as the product of a pair of in- and outgoing messages on the corresponding variable node:

$$g_i(x_i) = \mu_{x_i \rightarrow f_m}(x_i) \times \mu_{f_m \rightarrow x_i}(x_i). \quad (5)$$

If the graph does contain cycles, the algorithm becomes iterative and the computed marginals are no longer assured to be exact. The larger the cycles are, the more accurately the computed marginals will approximate the true marginals.

3. SYSTEM MODEL

We consider a flat-fading MIMO channel with N_T transmit and N_R receive antennas. The transmitter, based on BICM (as illustrated in Figure 2), encodes and interleaves a sequence of L information bits $\mathbf{b} = [b_1, \dots, b_L]$. The resulting coded bits are mapped to a sequence of K coded symbol vectors \mathbf{a}_k , $k = 1, \dots, K$, each of dimension $N_T \times 1$. The n th entry of \mathbf{a}_k denotes the coded symbol, transmitted by the n th antenna at instant k . The mapping is described by a bijective mapping function $\mathcal{M} : \{0, 1\}^{MN_T} \rightarrow \Omega^{N_T}$, where Ω denotes a 2^M -ary signal set, that is,

$$\mathbf{a}_k = \mathcal{M}(a_k[1], \dots, a_k[MN_T]), \quad (6)$$

with $\{a_k[m], m = 1, \dots, MN_T\}$ denoting the MN_T coded bits that are contained in the symbol vector \mathbf{a}_k . Irrespective of the type of mapping function, whether it concerns a single- or multidimensional [24] mapping, we can generally state that each symbol vector \mathbf{a}_k depends on MN_T bits.

Note that inserting a bit interleaver between the encoder and the modulator spreads the burst errors, introduced by the time-selective fading channel. This way, the channel appears to be uncorrelated from the decoder's point of view and the time diversity provided by the fading channel is fully exploited.

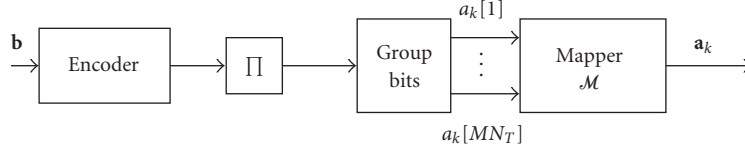


FIGURE 2: Transmitter structure.

Assuming a flat-fading channel, the received signal after matched filtering can be captured in the following discrete-time model:

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{a}_k + \mathbf{w}_k, \quad (7)$$

where \mathbf{y}_k is a $N_R \times 1$ vector of received signal samples at time instant k , \mathbf{H}_k denotes the $N_R \times N_T$ channel matrix, \mathbf{a}_k denotes the $N_T \times 1$ transmitted symbol vector, with an average energy per symbol equal to E_s , and \mathbf{w}_k is a $N_R \times 1$ vector of independent white complex Gaussian noise samples with independent real and imaginary parts each with a variance equal to $N_0/2$. We introduce the matrix of received samples $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_K]$.

In practice, the channel coefficients corresponding to the links between the different transmit and receive antennas will not be (totally) uncorrelated. The impact of this spatial correlation can be modeled by decomposing the channel matrix at each time instant as follows [21, 22]:

$$\mathbf{H}_k = \Sigma_R^{1/2} \mathbf{N} \Sigma_T^{1/2}, \quad (8)$$

where Σ_T and Σ_R denote the transmit and receive array correlation matrices and where \mathbf{N} denotes a $N_R \times N_T$ matrix containing i.i.d. zero-mean, unit-variance complex Gaussian elements. Various models have been proposed to characterize the temporal behavior of fading channels. Capitalizing on the information-theoretic results from [25], we adopt a first-order autoregressive model or Gauss-Markov model in this paper. Accordingly, our fading channel can be modeled as

$$\mathbf{H}_k = \alpha \mathbf{H}_{k-1} + \sqrt{1 - \alpha^2} \Sigma_R^{1/2} \mathbf{N}_k \Sigma_T^{1/2}, \quad (9)$$

where \mathbf{N}_k represents a $N_R \times N_T$ matrix containing i.i.d. zero-mean, unit-variance complex Gaussian elements. We further assume that the channel retains the steady state statistics given by (8) at instant $k = 1$. Thus, \mathbf{H}_k will be a stationary process with the following properties, for all time instants k :

$$\begin{aligned} E \left[\left(\mathbf{H}_k^{(n,m)} \right)^* \mathbf{H}_k^{(n',m')} \right] &= \Sigma_R^{(n,n')} \Sigma_T^{(m,m')}, \\ E \left[\left(\mathbf{H}_{k-1}^{(n,m)} \right)^* \mathbf{H}_k^{(n',m')} \right] &= \alpha \Sigma_R^{(n,n')} \Sigma_T^{(m,m')}, \end{aligned} \quad (10)$$

where $\mathbf{X}^{(n,m)}$ denotes the (n, m) th entry of the matrix \mathbf{X} . The coefficient α (with $|\alpha| < 1$) is related to the Doppler spread f_d according to the first-order approximation of Jakes' channel model [26]:

$$\alpha = J_0(2\pi f_d T), \quad (11)$$

where T is the symbol period and $J_0(\cdot)$ denotes the zeroth-order Bessel function of the first kind. The closer α to 1, the smaller the Doppler spread and the slower the fading. Channel model (9) is general and permits both temporal and spatial correlations. Note that a similar channel model was adopted in [20] for single-input multiple-output (SIMO) channels. Several other channel models can be considered as special cases of our model. The quasi-static correlated fading model from [21, 22] is obtained by setting $\alpha = 1$. The fast-fading model from [11] with uncorrelated antennas can be cast into this general model by setting $\Sigma_T = \mathbf{I}$ and $\Sigma_R = \mathbf{I}$.

To facilitate the analysis in the remainder of the paper, we introduce a vector notation of the channel matrix $\mathbf{h} = \text{vec}(\mathbf{H}^T)$, where the different rows of \mathbf{H} are transposed and stacked in the $N_T N_R \times 1$ column vector \mathbf{h} . Based on this new notation, we can rewrite the channel state (9) in the following manner:

$$\mathbf{h}_k = \alpha \mathbf{h}_{k-1} + \sqrt{1 - \alpha^2} \Sigma^{1/2} \mathbf{n}_k, \quad (12)$$

where we introduced the array correlation matrix $\Sigma \doteq \Sigma_R \otimes \Sigma_T$ (with \otimes denoting the Kronecker product) and where the $N_T N_R \times 1$ vector \mathbf{n}_k contains i.i.d. zero-mean, unit-variance Gaussian elements.

4. DETECTION AND ESTIMATION USING THE SP ALGORITHM

The main objective of the receiver in digital communication systems is to detect the transmitted information bits. In order to do so, the receiver requires an accurate estimate of the channel matrix (at each time instant). In this section, we adopt the concepts introduced in Section 2 to the detection and estimation problem at hand. The resulting algorithm yields channel estimates and reveals how these estimates should be applied in order to detect the information bits. The theoretical derivations from this section are transformed into a practical algorithm in Section 5.

4.1. Factor graph

Considering the information bits \mathbf{b} , the data symbol matrix $\mathbf{A} = \{\mathbf{a}_k\}_{k=1, \dots, K}$, and the set of all channel gain matrices $\tilde{\mathbf{H}} = \{\mathbf{H}_k\}_{k=1, \dots, K}$ as variables, we can write their joint a posteriori distribution as

$$p(\mathbf{b}, \mathbf{A}, \tilde{\mathbf{H}} | \mathbf{Y}) \propto p(\mathbf{b}) p(\mathbf{A} | \mathbf{b}) p(\tilde{\mathbf{H}}) p(\mathbf{Y} | \mathbf{A}, \tilde{\mathbf{H}}), \quad (13)$$

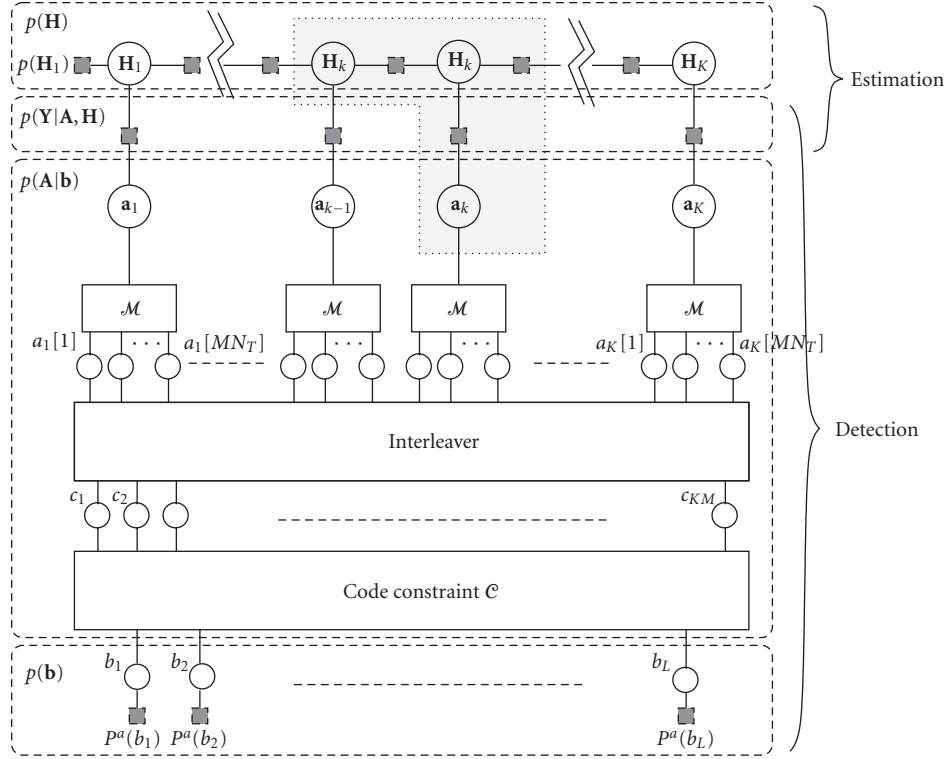


FIGURE 3: Factor graph representation of $p(\mathbf{b}, \mathbf{A}, \mathbf{H} | \mathbf{Y})$, up to a multiplicative constant. The grey area is shown in detail in Figure 4.

where we assumed that the transmitted symbols are independent with respect to the channel. This is a reasonable assumption, since it is hard to obtain accurate channel knowledge at the transmitter side in fast-fading channels and it is therefore difficult to exploit channel knowledge for selecting optimal transmission strategies. Observing the Markov chain behavior of the channel (9), we can factor the joint probability of the channel matrices at different time instants $1, \dots, K$ as follows:

$$p(\bar{\mathbf{H}}) = p(\mathbf{H}_1) \prod_{k=2}^K p(\mathbf{H}_k | \mathbf{H}_{k-1}), \quad (14)$$

where $p(\mathbf{H}_k | \mathbf{H}_{k-1})$ is fully determined by (9)

$$p(\mathbf{H}_k | \mathbf{H}_{k-1}) \propto \exp \left(- \frac{1}{1 - \alpha^2} [\mathbf{h}_k - \alpha \mathbf{h}_{k-1}]^H \Sigma^{-1} [\mathbf{h}_k - \alpha \mathbf{h}_{k-1}] \right), \quad (15)$$

where $\mathbf{h}_k = \text{vec}(\mathbf{H}_k^T)$. The flat-fading channel model (7) further implies that

$$p(\mathbf{Y} | \mathbf{A}, \bar{\mathbf{H}}) = \prod_{k=1}^K p(y_k | \mathbf{a}_k, \mathbf{H}_k) \propto \prod_{k=1}^K \exp \left(- \frac{1}{N_0} \|\mathbf{y}_k - \mathbf{H}_k \mathbf{a}_k\|^2 \right), \quad (16)$$

where $\|\cdot\|$ denotes the Frobenius norm. Interpreting $p(\mathbf{b}, \mathbf{A}, \bar{\mathbf{H}} | \mathbf{Y})$ as a function of the variables \mathbf{b} , \mathbf{A} , and $\bar{\mathbf{H}}$ and taking the factorizations (13), (14), and (16) into account, we obtain the factor graph depicted in Figure 3; for more clarity, a detail of Figure 3 is presented in Figure 4. We assume that the information bits are independent. The node marked \mathcal{C} represents the constraint on the coded bits, enforced by the code. Together with the interleaver and the mapper nodes, this part of the graph represents the factorization of $p(\mathbf{A} | \mathbf{b})$.

4.2. Sum-product algorithm

The SP algorithm permits us to compute the marginals of $p(\mathbf{b}, \mathbf{A}, \bar{\mathbf{H}} | \mathbf{Y})$. The purpose of the receiver consists in detecting the information bits; hence, the only relevant marginals are the a posteriori probabilities of the information bits $p(b_l | \mathbf{Y})$ for all l . In order to recover these, we compute the corresponding messages on the factor graph.

Unfortunately, the graph from Figure 3 contains cycles. It is well known [13] that in this scenario (i) the SP algorithm produces approximations of the marginals, instead of the exact marginals, and (ii) the SP algorithm becomes iterative. Although suboptimal, the SP algorithm still produces good results, as long as the cycles are not too short, and sufficiently many iterations are performed [23].

We will distinguish two phases within the iterative algorithm: a *detection phase* and an *estimation phase*. Information about the coded symbols and the channel is exchanged between these two stages.

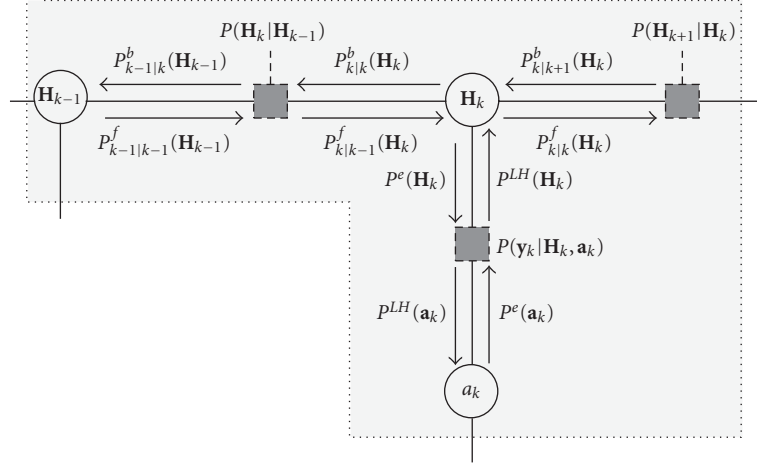


FIGURE 4: Details of the grey area from Figure 3, including messages.

4.2.1. Detection

The detector corresponds to the nodes $p(\mathbf{b})$, $p(\mathbf{A} | \mathbf{b})$, and $p(\mathbf{Y} | \mathbf{A}, \tilde{\mathbf{H}})$ in the factor graph from Figure 3. It has two main objectives.

- (1) To compute the extrinsic information of the coded symbol vectors $P^e(\mathbf{a}_k)$. This information is required for the channel estimation, as explained in Section 4.2.2.
- (2) To return the a posteriori probabilities of the information bits, after convergence of the SP algorithm.

A typical iterative detector operates according to the turbo principle by exchanging the so-called extrinsic information between the demapper and the decoder. Although a thorough investigation of these parts is not within the scope of the present paper, we provide a short overview of their interaction. The interested reader is referred to [4, 13, 24] for more details.

At the start of the detection phase, we receive channel information from the estimator by means of the messages¹ $P^e(\mathbf{H}_k)$ defined in Figure 4. Together with the information obtained from the observation \mathbf{y}_k , we compute the messages $P^{LH}(\mathbf{a}_k)$ according to the SP rule (4), that is,

$$P^{LH}(\mathbf{a}_k = \tilde{\mathbf{a}}) = \int_{\mathbf{H}_k} P^e(\mathbf{H}_k) p(\mathbf{y}_k | \mathbf{H}_k, \mathbf{a}_k = \tilde{\mathbf{a}}) d\mathbf{H}_k \quad (17)$$

$$\forall \tilde{\mathbf{a}} \in \Omega^{N_T}.$$

The message $P^{LH}(\mathbf{a}_k)$ can be interpreted as the likelihood (LH) of the observation \mathbf{y}_k given the transmitted symbol vector \mathbf{a}_k and the a priori distribution of the channel $P^e(\mathbf{H}_k)$. The operation referred to as demapping converts these symbol likelihoods into coded-bit likelihoods by accepting from the decoder extrinsic information on the coded

bits,

$$P^{LH}(a_k[m]) = \mathcal{F}_{M-D}(P^{LH}(\mathbf{a}_k); P^e(a_k[m']), \forall m' \neq m), \quad (18)$$

where $P^e(a_k[m'])$ denotes the extrinsic information with respect to the m' th bit of the k th symbol vector, provided by the decoder. A description of $\mathcal{F}_{M-D}(\cdot)$ can be found in [4, 24]. Similarly, the decoder accepts a deinterleaved version of the bit likelihoods $P^{LH}(a_k[m])$ and a priori information of the information bits $P^a(b_l)$ to update the extrinsic information $P^e(a_k[m])$:

$$P^e(a_k[m]) = \mathcal{F}_{D-M}(P^{LH}(a_k'[m']), \forall k', m'; P^a(b_l), \forall l). \quad (19)$$

For various codes, evaluation of $\mathcal{F}_{D-M}(\cdot)$ can be done in a computationally efficient manner [27–29]. Iterations between the demapper and decoder are performed until convergence. The detection phase ends by returning the extrinsic symbol vector probabilities $P^e(\mathbf{a}_k) = \prod_m P^e(a_k[m])$ to the estimator.

When the entire SP algorithm has converged, the decoder computes the extrinsic probabilities of the information bits in an efficient manner:

$$P^e(b_l) = \mathcal{F}_D(P^{LH}(a_k[m]), \forall k, m; P^a(b_{l'}), \forall l'). \quad (20)$$

The resulting a posteriori probabilities of the information bits are obtained as

$$p(b_l | \mathbf{Y}) \propto P^e(b_l) \times P^a(b_l). \quad (21)$$

Based on (21), final decisions with respect to the information bits are made. Algorithm 1 summarizes the operation of the detector.

4.2.2. Estimation

The estimation phase corresponds to the SP operation on the nodes $p(\tilde{\mathbf{H}})$ and $p(\mathbf{Y} | \mathbf{A}, \tilde{\mathbf{H}})$. At the beginning of the estimation phase, we have the extrinsic symbol vector probabilities

¹ Section 4.2.2 considers how to compute $P^e(\mathbf{H}_k)$.

```

(1) input:  $P^e(\mathbf{H}_k), \forall k$  (from estimator)
(2) compute  $P^{\text{LH}}(\mathbf{a}_k) = \int_{\mathbf{H}_k} P^e(\mathbf{H}_k) p(\mathbf{y}_k | \mathbf{H}_k, \mathbf{a}_k) d\mathbf{H}_k, \forall k$ 
(3) initialize  $P^e(a_k[m]) = 1/2, \forall k, m$ 
(4) for  $i = 1$  to  $I_{\text{MAX}}$  do
(5)   compute  $P^{\text{LH}}(a_k[m]) = \mathcal{F}_{M-D}(P^{\text{LH}}(\mathbf{a}_k); P^e(a_k[m']), \forall m' \neq m)$ 
(6)   compute  $P^e(a_k[m]) = \mathcal{F}_{D-M}(P^{\text{LH}}(a_{k'}[m']), \forall k', m'; P^a(b_l), \forall l)$ 
(7) end for
(8) return:  $P^e(\mathbf{a}_k) = \prod_m P^e(a_k[m]), \forall k$  (to estimator)
(9) if SP-algorithm converged then
(10)  compute  $P^e(b_l) = \mathcal{F}_D(P^{\text{LH}}(a_k[m]), \forall k, m; P^a(b_{l'}), \forall l')$ 
(11)  return: decisions on  $p(b_l | \mathbf{Y}) \propto P^e(b_l) \times P^a(b_l)$ 
(12) end if

```

ALGORITHM 1: Description of the detector operation.

$P^e(\mathbf{a}_k)$ at our disposal. The goal of the estimator is to update the extrinsic channel probabilities $P^e(\mathbf{H}_k)$ and feed these back to the detector. We distinguish two types of messages in the evaluation of the sum-product algorithm: forward and backward messages.

Forward message passing

In the forward message-passing phase, we compute the messages $P_{k|k-1}^f(\mathbf{H}_k)$, $P_{k|k}^f(\mathbf{H}_k)$, and $P^{\text{LH}}(\mathbf{H}_k)$ which are defined in Figure 4. The relation between these messages is found by a straightforward application of the sum-product rules (3) and (4). Based on (4), we deduce the following relations:

$$P_{k|k-1}^f(\mathbf{H}_k) = \int_{\mathbf{H}_{k-1}} P_{k-1|k-1}^f(\mathbf{H}_{k-1}) p(\mathbf{H}_k | \mathbf{H}_{k-1}) d\mathbf{H}_{k-1}, \quad (22)$$

$$P^{\text{LH}}(\mathbf{H}_k) = \sum_{\tilde{\mathbf{a}} \in \Omega^{N_T}} P^e(\mathbf{a}_k = \tilde{\mathbf{a}}) p(\mathbf{y}_k | \mathbf{H}_k, \mathbf{a}_k = \tilde{\mathbf{a}}). \quad (23)$$

From (3), we obtain

$$P_{k|k}^f(\mathbf{H}_k) = P^{\text{LH}}(\mathbf{H}_k) \times P_{k|k-1}^f(\mathbf{H}_k). \quad (24)$$

Combining (22), (23), and (24), we obtain a recursive relation between $P_{k|k}^f(\mathbf{H}_k)$ and $P_{k-1|k-1}^f(\mathbf{H}_k)$ of the form

$$\begin{aligned} P_{k|k}^f(\mathbf{H}_k) &= P^{\text{LH}}(\mathbf{H}_k) \int_{\mathbf{H}_{k-1}} P_{k-1|k-1}^f(\mathbf{H}_{k-1}) p(\mathbf{H}_k | \mathbf{H}_{k-1}) d\mathbf{H}_{k-1} \\ &\doteq \mathcal{F}_k^f(P_{k-1|k-1}^f(\mathbf{H}_{k-1})). \end{aligned} \quad (25)$$

Note that when a variable is defined over a continuous domain (i.e., $\mathbb{C}^{N_T \times N_R}$ in the case of \mathbf{H}_k), representation and computation of the messages is a major complexity issue in the SP algorithm. In Section 5, we will tackle this particular problem.

Backward message passing

Based on the SP rules, we can also compute the backward messages from Figure 4,

$$P_{k|k}^b(\mathbf{H}_k) = P^{\text{LH}}(\mathbf{H}_k) \times P_{k|k+1}^b(\mathbf{H}_k), \quad (26)$$

$$P_{k-1|k}^b(\mathbf{H}_{k-1}) = \int_{\mathbf{H}_k} P_{k|k}^b(\mathbf{H}_k) p(\mathbf{H}_k | \mathbf{H}_{k-1}) d\mathbf{H}_k. \quad (27)$$

Again, we obtain a backward recursive relation between these messages:

$$\begin{aligned} P_{k-1|k}^b(\mathbf{H}_{k-1}) &= \int_{\mathbf{H}_k} P^{\text{LH}}(\mathbf{H}_k) P_{k|k+1}^b(\mathbf{H}_k) p(\mathbf{H}_k | \mathbf{H}_{k-1}) d\mathbf{H}_k \\ &\doteq \mathcal{F}_k^b(P_{k|k+1}^b(\mathbf{H}_k)). \end{aligned} \quad (28)$$

Information to the detector

As readily seen from Figure 4, $P^e(\mathbf{H}_k)$ follows from (3),

$$P^e(\mathbf{H}_k) = P_{k|k+1}^b(\mathbf{H}_k) \times P_{k|k-1}^f(\mathbf{H}_k). \quad (29)$$

Finally, the estimator returns this extrinsic information about the channel matrix to the detector. The operation of the entire estimation is summarized in Algorithm 2.

Regarding complexity

An important issue with respect to factor graphs is how the messages are scheduled along the graph during the SP calculation. A proper scheduling of the messages can reduce the computational complexity of the receiver. As outlined in Section 4.2.1, the detector itself is iterative. Iterations occur between the demapper and decoder or within the decoder itself (e.g., turbo-like codes). To minimize the overhead caused by the estimation, we propose to embed the estimation into this iterative detection process. Our intent is to perform only a single demapping or decoding iteration within each detection stage and to maintain, rather than reset, state information at the beginning of the detection phase. More

(1) input: $P^e(\mathbf{a}_k)$, $\forall k$ (from detector)
 (2) initialize $P_{0|0}^f(\mathbf{H}_0)$ and $P_{K|K+1}^b(\mathbf{H}_K)$
 (3) for $k = 1$ to K do
 (4) compute $P_{k|k}^f(\mathbf{H}_k) = \mathcal{F}_k^f(P_{k-1|k-1}^f(\mathbf{H}_{k-1}))$
 (5) end for
 (6) for $k = K$ to 1 do
 (7) compute $P_{k-1|k}^b(\mathbf{H}_{k-1}) = \mathcal{F}_k^b(P_{k|k+1}^b(\mathbf{H}_k))$
 (8) end for
 (9) return: $P^e(\mathbf{H}_k) = P_{k|k+1}^b(\mathbf{H}_k) \times P_{k|k-1}^f(\mathbf{H}_k)$ (to detector)

ALGORITHM 2: Description of estimator operation.

specifically, the value I_{MAX} in Algorithm 1 is set equal to $I_{\text{MAX}} = 1$ and the initialization $P^e(a_k[m]) = 1/2$, for all k, m is ignored. Furthermore, when the decoding process itself is iterative, only one decoding iteration per detection iteration is performed.

5. PRACTICAL ESTIMATION ALGORITHM

In this section we derive a practical iterative estimation algorithm based on the results from the previous section. Before we evaluate the SP algorithm, we recall that representation and computation of the messages in the SP algorithm is not always straightforward. In particular, messages that operate on continuous variables are often difficult to represent or can lead to intractable update rules (e.g., an intractable integration in (22) or (27)). However, a few message types render a fairly easy representation. Gaussian probability density functions (pdfs), for example, are entirely defined by their mean and covariance matrices. This allows a very straightforward representation. As we observe from (23), $P^{\text{LH}}(\mathbf{H}_k)$, and also $P_{k|k}^f(\mathbf{H}_k)$ and $P_{k|k}^b(\mathbf{H}_k)$ are *no* Gaussian pdfs, but rather a mixture of Gaussian pdfs. Furthermore, the number of terms in this mixture grows exponentially with increasing time index k for $P_{k|k}^f(\mathbf{H}_k)$ and with decreasing k for $P_{k|k}^b(\mathbf{H}_k)$. Hence, the exact representation and computation of these messages becomes intractable. In order to solve this problem, we perform a well-chosen approximation. The idea is to approximate each of these messages, again, by a single-Gaussian pdf (instead of a mixture of Gaussian pdfs).

In order to do so, we approximate the distribution $P^{\text{LH}}(\mathbf{H}_k)$ with the following distribution:

$$\begin{aligned} P^{\text{LH}}(\mathbf{H}_k) &= \sum_{\tilde{\mathbf{a}} \in \Omega^{N_T}} P^e(\mathbf{a}_k = \tilde{\mathbf{a}}) p(\mathbf{y}_k | \mathbf{H}_k, \mathbf{a}_k = \tilde{\mathbf{a}}) \\ &\approx p(\mathbf{y}_k | \mathbf{H}_k, \hat{\mathbf{a}}_k) \\ &\propto \exp\left(-\frac{1}{N_0} \|\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{a}}_k\|^2\right), \end{aligned} \quad (30)$$

where $\hat{\mathbf{a}}_k$ is defined as the soft-symbol decision based on the extrinsic probabilities

$$\hat{\mathbf{a}}_k = \sum_{\tilde{\mathbf{a}} \in \Omega^{N_T}} \tilde{\mathbf{a}} \times P^e(\mathbf{a}_k = \tilde{\mathbf{a}}). \quad (31)$$

The error induced by this approximation is minor when the distribution $P^e(\mathbf{a}_k)$ has a pronounced peak, that is, when $P^e(\mathbf{a}_k = \tilde{\mathbf{a}}) \approx 1$ for a particular $\tilde{\mathbf{a}}$ and $P^e(\mathbf{a}_k = \mathbf{a}^*) \ll 1$ for $\mathbf{a}^* \neq \tilde{\mathbf{a}}$. Hence, as long as the detector provides reliable information, the approximation is accurate. We conjecture that the approximation is quite accurate in *any relevant* context, since, in general, code-aided estimation schemes only perform well when they have access to sufficiently reliable information about the unknown symbols.

The approximation in formula (30) allows us to represent $P^{\text{LH}}(\mathbf{H}_k)$ by a Gaussian pdf. Since the product of Gaussian pdfs (as in (24) and (26)) and marginalization of a Gaussian pdf (as in (22), (23), and (27)), results in a Gaussian pdf again, *all* forward and backward messages on the graph turn out to be Gaussian pdfs. Hence, all messages within the SP algorithm can easily be represented by their mean and covariance matrices.

In the next two paragraphs, we tackle the actual computation of these messages. We consider two scenarios: correlated receive antennas and uncorrelated receive antennas.

5.1. Correlated receive antennas $\Sigma_R \neq \mathbf{I}$

As shown in Figure 3, the estimation phase corresponds to the upper part of the factor graph. It is readily seen from (12) and (30) that this part of the factor graph represents the following state-space model:

$$\begin{aligned} \mathbf{h}_k &= \alpha \mathbf{h}_{k-1} + \sqrt{1 - \alpha^2} \Sigma^{1/2} \mathbf{n}_k, \\ \mathbf{y}_k &= \hat{\mathbf{A}}_k \mathbf{h}_k + \mathbf{w}_k, \end{aligned} \quad (32)$$

where we introduced the $N_R \times N_T N_R$ matrix

$$\hat{\mathbf{A}}_k = \begin{bmatrix} \hat{\mathbf{a}}_k^T & 0 & \dots & 0 \\ 0 & \hat{\mathbf{a}}_k^T & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \hat{\mathbf{a}}_k^T \end{bmatrix}. \quad (33)$$

The evaluation of the SP algorithm on a factor graph representing a state-space model similar to (32) has been considered in [13, 23]. The main conclusion was that the SP algorithm boils down to a straightforward Kalman smoother. As we elaborated upon, all messages on the factor graph are Gaussian pdfs. The recursive relations between these are obtained by evaluating (25) and (28) for Gaussian pdfs. This results in a Kalman smoother, which defines the relation between the mean and covariance matrices of these Gaussian pdfs. We refer to [12, 23] for the Kalman filter/smoothing update rules.

5.2. Uncorrelated receive antennas $\Sigma_R = \mathbf{I}$

When receive correlation is absent or ignored, the section of the factor graph corresponding to $p(\tilde{\mathbf{H}})$ turns out to be decoupled. We can factorize the nodes corresponding to

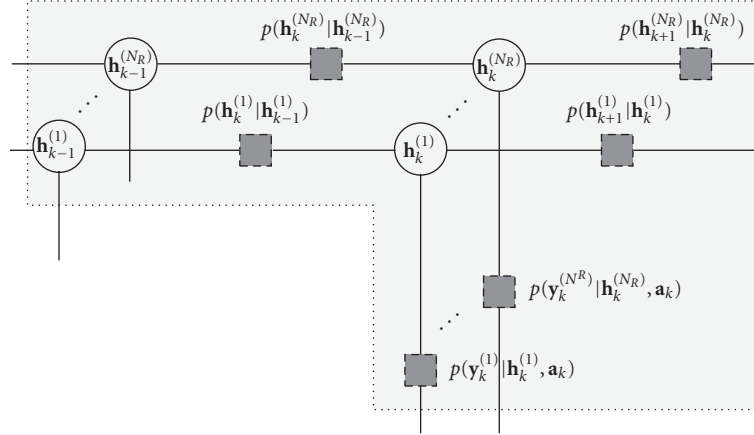


FIGURE 5: Details of the grey area from Figure 3, when receive antennas are uncorrelated ($\Sigma_R = \mathbf{I}$).

$p(\mathbf{H}_k | \mathbf{H}_{k-1})$ as follows:

$$\begin{aligned} p(\mathbf{H}_k | \mathbf{H}_{k-1}) &= \prod_{n=1}^{N_R} p(\mathbf{h}_k^{(n)} | \mathbf{h}_{k-1}^{(n)}) \\ &\propto \prod_{n=1}^{N_R} \exp\left(-\frac{1}{1-\alpha^2} [\mathbf{h}_k^{(n)} - \alpha \mathbf{h}_{k-1}^{(n)}]^H \Sigma_T^{-1} [\mathbf{h}_k^{(n)} - \alpha \mathbf{h}_{k-1}^{(n)}]\right), \end{aligned} \quad (34)$$

where $\mathbf{h}_k^{(n)}$ denotes the n th column of \mathbf{H}_k^T . Similarly, we can decouple the approximation for $P^{\text{LH}}(\mathbf{H}_k)$ in (30):

$$P^{\text{LH}}(\mathbf{H}_k) = \prod_{n=1}^{N_R} p(y_k^{(n)} | \hat{\mathbf{a}}_k, \mathbf{h}_k^{(n)}). \quad (35)$$

Note that the latter is valid for any Σ_R . We can easily take these factorizations into account by replacing the grey area in our original factor graph from Figure 3 with the grey area from Figure 5. The state-space equations that correspond to this part of the factor graph are now given by

$$\begin{aligned} \mathbf{h}_k^{(n)} &= \alpha \mathbf{h}_{k-1}^{(n)} + \sqrt{1-\alpha^2} \Sigma_T^{1/2} \mathbf{n}_k^{(n)}, \\ y_k^{(n)} &= \hat{\mathbf{a}}_k^T \mathbf{h}_k^{(n)} + \mathbf{w}_k^{(n)}, \end{aligned} \quad (36)$$

for $n = 1, \dots, N_R$. Again, evaluation of the SP algorithm boils down to Kalman smoothing. However, compared with the general case $\Sigma_R \neq \mathbf{I}$, the complexity has been reduced significantly. Instead of one large Kalman smoother, we encounter a bank of N_R parallel Kalman smoothers. Furthermore, the bulk of the required computations are common to all these Kalman smoothers. As seen from (36), only the observations $y_k^{(n)}$ differ among the state equations for different antennas. The other inputs remain the same and the Kalman smoothers share common covariance matrices (whereas the mean vectors differ). Breaking up the state equations according to (36) yields a reduction in the computational complexity proportional to N_R^2 .

5.3. Known data symbols: initialization

If all the transmitted symbols are known to the receiver, the message $P^e(\mathbf{a}_k = \mathbf{a})$ is 1 when \mathbf{a} equals the actual value of the k th transmitted symbol vector, and 0 otherwise. Thus, the resulting factor graph contains only the parts $p(\hat{\mathbf{H}})$ and $p(\mathbf{Y} | \mathbf{A}, \hat{\mathbf{H}})$ from Figure 3, along with the input messages $P^e(\mathbf{a}_k = \mathbf{a})$. This graph is cycle-free, hence, the a posteriori probability functions computed by the SP algorithm are *exact*. Naturally, this algorithm amounts to a standard data-aided Kalman smoother.

In practice, of course, we wish to transmit unknown coded symbols over the fading channel. Still, we periodically insert some known symbols to provide initial channel estimates and to prevent the algorithm from diverging. Divergence can occur due to the inherent ambiguities between the channel parameters and the unknown symbols (as mentioned in [5, 11]).

In the first iteration, no information is available about the unknown symbols and estimation is performed based on these pilot symbols only. More specifically, for instants k corresponding to unknown data symbol vectors, the messages $P^{\text{LH}}(\mathbf{H}_k)$ are ignored. This is equivalent to equating the soft symbols to zero in the state space (32) or (36). For each instant k that corresponds to a pilot symbol, $\hat{\mathbf{a}}_k$ is replaced by the actual value of the transmitted pilot symbol \mathbf{a}_k .

6. SIMULATION RESULTS

We present simulation results for a MIMO BICM scheme [24, 30] with $N_T = 2$ transmit antennas and $N_R = 2$ receive antennas. At the transmitter side, we assembled a rate 1/2 recursive convolutional code with octal polynomials $(37, 31)_8$, a random interleaver, and a BPSK symbol mapper. The channel is generated according to (9) and (11) for two different fading rates $f_d T = 0.02$ and $f_d T = 0.005$. The results shown in Figures 6 and 7 are for spatially uncorrelated channels ($\Sigma_T = \Sigma_R = \mathbf{I}$), whereas the impact of antenna correlation is considered in Figure 8. Frames consists of 1440 coded information bits, and a number of pilot symbols are periodically

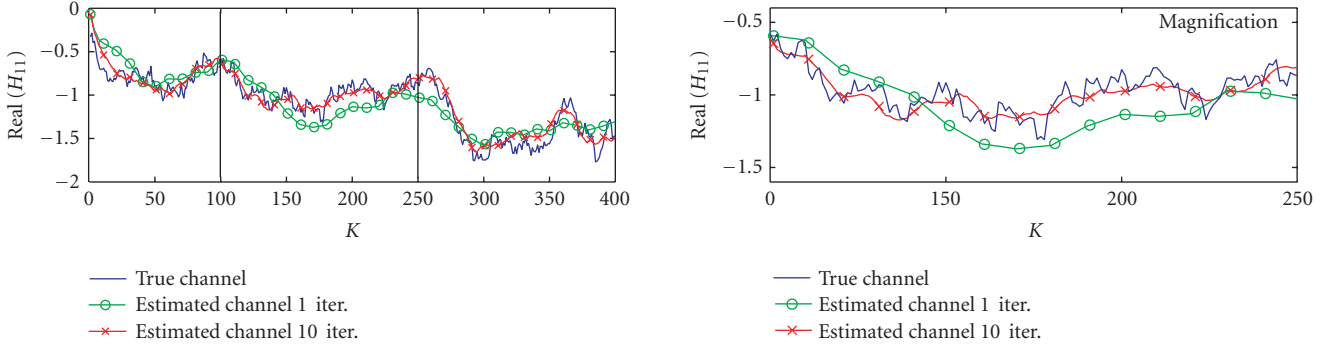


FIGURE 6: Comparison of the estimated channel and the true channel, in a convolutionally encoded system with $f_d T = 0.02$, $E_b/N_0 = 6$ dB, and 10% pilot symbols ($\Sigma_T = \Sigma_R = \mathbf{I}$).

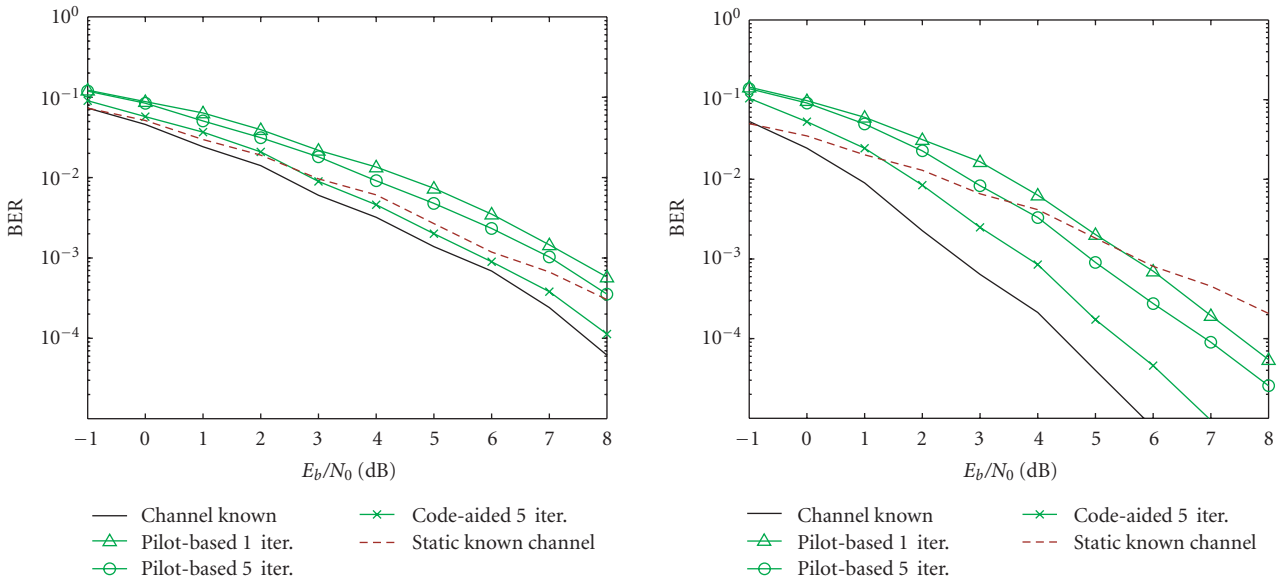


FIGURE 7: BER performance of convolutional code with $f_d T = 0.005$ and 5% pilot symbols (left) and $f_d T = 0.02$ and 10% pilot symbols (right).

inserted to provide initial channel estimates and to avoid divergence. Pilot symbol energy is set equal to the average data symbol energy. The bit energy to noise ratio (E_b/N_0) is computed without taking the energy required for pilot symbol transmission into account.

Figure 6 illustrates the channel-tracking performance, by comparing the mean value of the messages $P^e(\mathbf{H}_k)$ with the true channel \mathbf{H}_k . In the first iteration, only information about the pilot symbols is used, so that the algorithm corresponds to a pure data-aided Kalman smoother. As we observe, the ability to track the channel substantially improves after a few iterations. As expected, exploiting information from the decoder about the unknown coded symbols in the second and further iterations improves the channel estimation.

The curves in Figure 7 correspond to the BER performance exhibited on our MIMO time-varying fading channel ($f_d T = 0.005$ on the left and $f_d T = 0.02$ on the right,

both with no antenna correlation). We compare the performance of the iterative detector where the channel estimates are provided solely based on pilot symbols with the performance of an iterative code-aided estimation scheme, where the code-aided estimator is embedded in the iterative detector (as explained in Section 4.2.2). In Figure 7 (left), we inserted one pilot symbol (on each antenna) every 20 coded symbols, which correspond to a 5% pilot overhead. The performance of the iterative algorithm after convergence is close to the known-channel performance. Comparing the BER after the first iteration to the BER after convergence, we observe a 2 dB gain that results from iterating between the detection and the estimation. The code-aided estimator also yields more than 1 dB gain compared to the pilot-based estimator, after convergence. Figure 7 (right) illustrates the BER performance on a rapidly fading channel ($f_d T = 0.02$). First, observe the diversity gain of the time-varying fading channel compared to a static-fading channel ($f_d T = 0$ or $\alpha = 1$

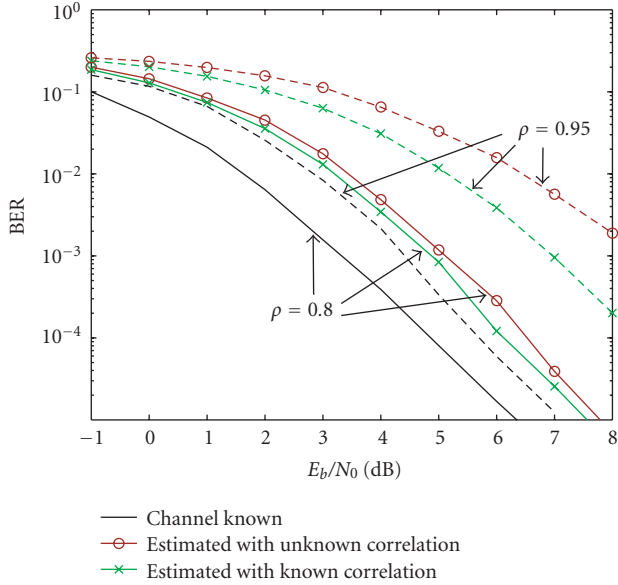


FIGURE 8: BER performance for correlated transmit antennas, with $f_d T = 0.02$ and 10% pilot symbols.

in (9)). This gain is obtained thanks to the interleaver, which spreads the error bursts, caused by occasionally deep fades, over the entire codeword. This property has been widely examined [2, 3] and emphasizes the benefit of using BICM for fading channels. Considering the estimation, we increased the number of pilots to 10% (insertion of 1 pilot symbol every 10 coded symbols) to avoid divergence of the iterative SP algorithm. The gain from exploiting the code is apparent again.

Finally, we consider the BER performance on a fading 2×2 MIMO channel with transmit antenna correlation. We assume that the correlation matrix is given by

$$\Sigma_T = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}. \quad (37)$$

Simulation results are shown for $\rho = 0.8$ and $\rho = 0.95$. We further consider two different scenarios: (i) the receiver knows the transmit correlation ρ and takes it into account in the SP computation; (ii) the receiver does not know the correlation and assumes $\rho = 0$. Figure 8 shows the BER performance after 3 iterations for a fading rate of $f_d T = 0.02$. For $\rho = 0.8$, the difference between the two scenarios is minor. Only for tightly coupled ($\rho = 0.95$) antennas, a significant performance gain is observed when taking the correlation into account. Observe also the well-known result that less correlated channels exhibit a better performance than more correlated channels.

7. CONCLUSIONS

By means of factor graph theory we have derived an iterative algorithm for joint code-aided estimation and detection on a time-varying flat-fading MIMO channel with spatial correlation. The tightly coupled estimation and detection algo-

gorithms exchange messages in accordance with the SP algorithm. The estimation algorithm boils down to a Kalman smoother that uses soft-symbol information provided by the decoder. Since MIMO detection often involves iterative decoding, we can limit the computational overhead caused by the estimation by embedding the estimation stages into the detection stages. When the receive antennas do not exhibit correlation, we can split the Kalman smoother into a bank of parallel Kalman smoothers, which significantly reduces the complexity.

Simulation results have shown that a significant performance improvement (in terms of BER) is obtained by exploiting information from the unknown transmitted symbols compared to estimation based on pilot symbols only. Also, ignoring the spatial correlation leads to a minor performance degradation, as long as the correlation is not too high.

ACKNOWLEDGMENTS

This work has been supported by the Interuniversity Attraction Poles Program P5/11-Belgian Science Policy and by the Network of Excellence in Wireless Communications (NEWCOM) funded by the European Commission. The first author also gratefully acknowledges the support from the Fund for Scientific Research in Flanders (FWO-Vlaanderen).

REFERENCES

- [1] E. Biglieri, J. Proakis, and S. Shamai, "Fading channels: information-theoretic and communications aspects," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2619–2692, 1998.
- [2] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 927–946, 1998.
- [3] E. K. Hall and S. G. Wilson, "Design and analysis of turbo codes on Rayleigh fading channels," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 2, pp. 160–174, 1998.
- [4] X. Li and J. A. Ritcey, "Trellis-coded modulation with bit interleaving and iterative decoding," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 4, pp. 715–724, 1999.
- [5] M. C. Valenti and B. D. Woerner, "Iterative channel estimation and decoding of pilot symbol assisted turbo codes over flat-fading channels," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 9, pp. 1697–1705, 2001.
- [6] C. Komninakis and R. D. Wesel, "Joint iterative channel estimation and decoding in flat correlated Rayleigh fading," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 9, pp. 1706–1717, 2001.
- [7] C. Komninakis, C. Fragouli, A. H. Sayed, and R. D. Wesel, "Multi-input multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Transactions on Signal Processing*, vol. 50, no. 5, pp. 1065–1076, 2002.
- [8] M. Dong, L. Tong, and B. M. Sadler, "Optimal insertion of pilot symbols for transmissions over time-varying flat fading channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 5, pp. 1403–1418, 2004.
- [9] D. Schafhuber, G. Matz, and F. Hlawatsch, "Kalman tracking of time-varying channels in wireless MIMO-OFDM systems,"

- in *Proceedings of the 37th IEEE Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1261–1265, Pacific Grove, Calif, USA, November 2003.
- [10] E. Baccarelli, R. Cusani, and S. Galli, “A novel adaptive receiver with enhanced channel tracking capability for TDMA-based mobile radio communications,” *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 9, pp. 1630–1639, 1998.
- [11] Z. Liu, X. Ma, and G. B. Giannakis, “Space-time coding and Kalman filtering for time-selective fading channels,” *IEEE Transactions on Communications*, vol. 50, no. 2, pp. 183–186, 2002.
- [12] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1979.
- [13] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.
- [14] T. Wadayama, “An iterative decoding algorithm for channels with additive linear dynamical noise,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E86-A, no. 10, pp. 2452–2460, 2003.
- [15] M. Shen, H. Niu, and H. Liu, “Iterative receiver design in Rayleigh fading using factor graph,” in *Proceedings of the 57th IEEE Vehicular Technology Conference (VTC '03)*, vol. 4, pp. 2604–2608, Jeju, South Korea, April 2003.
- [16] G. Colavolpe, A. Barbieri, G. Caire, and N. Bonneau, “Bayesian and nonBayesian methods for iterative joint decoding and detection in the presence of phase noise,” in *Proceedings of IEEE International Symposium on Information Theory (ISIT '04)*, p. 131, Chicago, Ill, USA, June-July 2004.
- [17] J. Dauwels and H.-A. Loeliger, “Phase estimation by message passing,” in *Proceedings of IEEE International Conference on Communications (ICC '04)*, vol. 1, pp. 523–527, Paris, France, June 2004.
- [18] A. P. Worthen and W. E. Stark, “Unified design of iterative receivers using factor graphs,” *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 843–849, 2001.
- [19] N. Wiberg, *Codes and decoding on general graphs*, Ph.D. thesis, Linköping University, Linköping, Sweden, 1996.
- [20] V. K. Nguyen, L. B. White, E. Jaffrot, M. Soamiadana, and I. Fijalkow, “Recursive receivers for diversity channels with correlated flat fading,” *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 5, pp. 754–764, 2003.
- [21] D. Gesbert, H. Bölcskei, D. A. Gore, and A. J. Paulraj, “Outdoor MIMO wireless channels: models and performance prediction,” *IEEE Transactions on Communications*, vol. 50, no. 12, pp. 1926–1934, 2002.
- [22] J. H. Kotecha and A. M. Sayeed, “Transmit signal design for optimal estimation of correlated MIMO channels,” *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 546–557, 2004.
- [23] H.-A. Loeliger, “An introduction to factor graphs,” *IEEE Signal Processing Magazine*, vol. 21, no. 1, pp. 28–41, 2004.
- [24] F. Simoens, H. Wymeersch, and M. Moeneclaey, “Spatial mapping for MIMO systems,” in *Proceedings of IEEE Information Theory Workshop (ITW '04)*, pp. 187–192, San Antonio, Tex, USA, October 2004.
- [25] H. S. Wang and P.-C. Chang, “On verifying the first-order Markovian assumption for a Rayleigh fading channel model,” *IEEE Transactions on Vehicular Technology*, vol. 45, no. 2, pp. 353–357, 1996.
- [26] W. C. Jakes, *Mobile Microwave Communication*, John Wiley & Sons, New York, NY, USA, 1974.
- [27] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Transactions on Information Theory*, vol. 20, no. 2, pp. 284–287, 1974.
- [28] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: turbo-codes. 1,” in *Proceedings of IEEE International Conference on Communications (ICC '93)*, pp. 1064–1070, Geneva, Switzerland, May 1993.
- [29] D. J. C. MacKay, “Good error-correcting codes based on very sparse matrices,” *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 399–431, 1999.
- [30] F. Simoens, H. Wymeersch, H. Steendam, and M. Moeneclaey, “Synchronization for MIMO systems,” in *Smart Antennas—State of the Art*, EURASIP Book Series on Signal Processing and Communications, chapter 6, Hindawi, New York, NY, USA, 2005.

Frederik Simoens received the M.S. degree in electrical engineering in 2003 from Ghent University, Belgium. In 2004 he was granted a Fund for Scientific Research in Flanders (FWO-Vlaanderen) scholarship to prepare a Ph.D., toward which he is currently working within the Department of Telecommunications and Information Processing (TELIN) of Ghent University. His main research interests include parameter estimation, modulation and coding for wireless digital communications.



Marc Moeneclaey received the Diploma and the Ph.D. degree, both in electrical engineering, from Ghent University, Gent, Belgium, in 1978 and 1983, respectively. He is currently a Professor in the Department of Telecommunications and Information Processing at Ghent University. His main research interests are in statistical communication theory, carrier and symbol synchronization, bandwidth-efficient modulation and coding, spread spectrum, as well as satellite and mobile communication. He is the author of about 250 scientific papers in international journals and conference proceedings. Together with H. Meyr (RWTH Aachen) and S. Fechtel (Siemens AG), he is the coauthor of the book *Digital Communication Receivers—Synchronization, Channel estimation, and Signal Processing* (New York: Wiley, 1998).

