Semiblind space–time decoding in wireless communications: a maximum likelihood approach

Joaquín Míguez*1, Luis Castedo

Departamento de Electrónica e Sistemas, Universidade da Coruña, Spain

Received 29 May 2000; received in revised form 12 September 2001; accepted 12 September 2001

Abstract

It has been recently shown that deploying multiple transmitting and receiving antennae can substantially improve the capacity of multipath wireless channels if the rich time-scattering is properly exploited. Space–time coding (STC) is a novel proposal that combines channel coding techniques suitable for multiple transmitting elements with signal processing algorithms that exploit the spatial and temporal diversity at the receiver. In this paper, we focus on the signal processing perspective and propose a novel space–time semiblind decoding scheme that performs maximum likelihood (ML) based channel estimation and data detection. The multiple input multiple output (MIMO) time-scattering channel is estimated using a block iterative expectation-maximization (EM) algorithm that fully exploits the statistical features of the transmitted signal together with the knowledge of a small number of transmitted symbols, hence the term semiblind. Data detection is efficiently carried out using the Viterbi algorithm. In order to reduce the computational load of the receiver, a modification of the EM algorithm with a potentially lower complexity is also suggested. Computer simulations show that the proposed semiblind decoder clearly outperforms conventional receivers that estimate the channel parameters exclusively from the a priori known transmitted symbols. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Maximum likelihood; Space–time processing; Space–time decoding; Channel estimation; Sequence detection; Multiuser detection; Interference suppression; EM algorithm; Semiblind receivers

1. Introduction

The development of third generation mobile communication systems has brought wireless digital transmission to the focus of research in the last few years. The wireless channel is a hostile medium where communication signals are severely distorted due to multipath propagation, additive white Gaussian noise (AWGN) and inter-symbol interference (ISI) caused by time scattering. Therefore, it is hard to achieve high data rates and it is expected that future mobile communication systems will incorporate sophisticated coding and signal processing capabilities in order to accommodate a wide range of new wideband services including multimedia transmission [5].

The key to overcome the limitations imposed by the wireless channel and achieve high transmission rates is to introduce some form of diversity. We will use this term, hereinafter, to indicate the availability

* Corresponding author. Department of Electrical and Computer Engineering, State University of New York at Stony Brook (SUNY), Light Engineering Building, Stony Brook, NY 11794, USA. Tel.: +34-981-167150; fax: +34-981-167160.
1 Facultade de Informática, Campus de Elviña s/n, 15071 A Coruña, Spain.
E-mail address: jmiguez@udc.es (J. Míguez), luis@des. fi.udc.es (L. Castedo).
of different replicas of a signal of interest that can be combined to enhance performance in some sense [35,51]. Two interesting examples are:

- **Temporal diversity**: multipath propagation in wireless channels causes several replicas of the transmitted signal to arrive at the receiver with different delays. Channel coding together with interleaving also provides temporal redundancy that may be understood as a form of diversity [24,35].

- **Spatial diversity**: spatially separated antennae can be used at the receiver to obtain different replicas of the communication signal. Spatial transmit diversity has also recently been considered as a means to increase capacity [25,52].

In particular, it has been recently demonstrated that deploying multiple antennae both at the transmitter and the receiver leads to a significant increase in the system spectral efficiency. Quantitatively, it was first shown by G.J. Foschini that the system capacity grows linearly with the number of transmitting antennae \(^2\) [7,8]. Further combination with temporal diversity has led to a novel proposal termed space–time coding (STC) that brings together channel coding techniques suitable for multiple transmitting antennae with signal processing at the receiver [7,8,10,15,26,29,32–35,50].

Fig. 1 shows the basic building blocks of a wireless communication system with space–time (ST) coding capabilities [10]. The bit stream to be transmitted, \(b(l)\), is fed into a temporal coding stage. In the simplest case, this processing block converts the bit stream into a sequence of independent symbols. In a more general setup, a channel code is used to produce a temporally correlated symbol sequence with desirable properties that enable error detection and correction at the receiver. This sequence, \(s(l)\), is the input of a serial-to-parallel converter that produces \(N\) symbol substreams, \(^3\) \(s_1(n), \ldots, s_N(n)\). Parallel synchronous transmission of the \(N\) substreams is carried out using a bank of \(N\) conventional radio-frequency modulators.

\(^2\) As long as this is less than or equal to the number of receiving antennae.

\(^3\) In some cases, such as the V-BLAST scheme proposed in [50], the different data substreams are encoded independently and it is easier to think of the space–time encoder as a serial-to-parallel converter followed by a bank of channel encoders.
and transmitting antennae. Due to the combination of channel coding and demultiplexing, the resulting communication signal may present both temporal and spatial correlation. Multipath propagation occurs between each transmitting and receiving element, resulting in a multiple input multiple output (MIMO) channel. A bank of $L \geq N$ coherent demodulators is employed at the receiver to obtain observations, $x_1(n), \ldots, x_L(n)$, that are processed by the ST decoder in order to compute estimates of the source bits, $\hat{b}(l)$.

The design of ST codes for flat fading MIMO channels has been studied in [10,32–35]. Although the time-scattering effect of typical wireless multipath channels is viewed by these authors as an impairment, they show that ISI does not reduce the performance gain provided by adequately designed ST codes. Indeed, the analytical results in [29] indicate that further improvements in the data rate can be achieved when time-scattering is properly exploited. An optimal STC structure that allows to achieve the capacity bound is also proposed in [29]. Unfortunately, its implementation requires to have full knowledge of the MIMO time-scattering channel, which is not available in practice.

In this paper, we focus on the signal processing aspect of STC systems. From this point of view, a number of techniques previously developed for channel estimation and data detection in MIMO systems are also relevant for STC. When a sufficiently large number of transmitted data are known a priori by the receiver (i.e., training sequences are employed), conventional signal processing [11,36] can be applied for supervised estimation of both the channel and the data. Transmission of training sequences, however, is not always possible or desirable and, therefore, several blind methods have been investigated during the last decade. Blind techniques rely on the statistical and structural knowledge of the communication signal in order to estimate the transmitted data without using training sequences and without any a priori knowledge of the channel [3,18,37]. Many blind algorithms for blind source separation (BSS) [3,4,17,26,41], blind channel identification [9,37,40,42,43] and joint channel/data estimation [30,31,53] can be either directly applied or straightforwardly extended for their use in STC systems. It is also widely recognized [33] that ST decoding resembles the multi-user detection (MUD) problem [44] encountered in code division multiple access (CDMA) systems. In CDMA, diversity is introduced by spreading the spectrum of the communication signals using distinct signature waveforms. Many blind MUD algorithms have been proposed [1,2,21,22,38,39,44–48] but most of them cannot be directly used for ST decoding because they rely on the knowledge of the users’ spreading codes. In the STC context, this requirement is equivalent to having some a priori structural knowledge of the MIMO channel, which is not usually available.

Nevertheless, blind techniques also pose a number of problems, which may include misconvergence due to inadequate choice of initial conditions [30,31,53], the need of very large data records to attain adequate results [3,4,41] and poor performance in the low signal-to-noise ratio (SNR) region [46]. Semi-blind techniques try to alleviate these drawbacks by combining statistical/structural criteria and the transmission of short training sequences. Most decision directed algorithms for joint channel/data estimation can be classified as semiblind when a training sequence is used for computing initial estimates [30,31,53]. Several blind subspace approaches, such as the sub-channel matching method in [16] and the projection approach in [23], also lend themselves to a semiblind implementation [14].

---

4 Either channel state information or soft data estimates must be obtained before decoding.

5 A training sequence is short if it is not enough for adequate supervised-only channel estimation.
combinations of a supervised criterion, typically least squares (LS), and a blind one, such as constant modulus (CM), have also been claimed to exhibit enhanced performance [12,13].

In the present paper, we propose a novel approach to ST decoding that consists of the estimation of the MIMO time-scattering channel coefficients and the detection of the transmitted data in the maximum likelihood (ML) sense. Channel estimation is carried out using the expectation-maximization (EM) algorithm [19] whereas joint symbol detection is efficiently performed with a dynamic programming algorithm (e.g., the Viterbi algorithm [28]) that makes use of side information obtained during the channel estimation stage. Notice that this approach is optimum when a trellis channel code and/or a modulation format with memory are used at the ST encoder. Although we show that the MIMO channel can be estimated in a totally blind way (i.e., without any knowledge of the transmitted symbols), we focus on a semiblind scheme that assumes that a small part of the transmitted symbols are known a priori at the receiver. Since the proposed semiblind ST decoder exploits the statistical knowledge of the transmitted data to a great extent, a very small number of a priori known symbols are enough to attain practically optimum performance. In fact, computer simulations show that the novel semiblind receiver is clearly more efficient than conventional decoders because much shorter training sequences are enough to attain the same bit error rate (BER).

The main drawback of the proposed semiblind decoding scheme is the computational complexity of the resulting algorithms. It can be easily shown that the complexity of ML-based channel estimation grows exponentially with the number of transmitting antennas and the channel memory size. To avoid this limitation, we also introduce a modification of the channel estimation EM algorithm that uses suboptimum detection schemes in order to reduce the computational load of channel estimation. As a result, a decision driven semiblind receiver is obtained.

The remaining of this paper is organized as follows. Section 2 presents the signal model under consideration. In Section 3 we introduce the optimization problem that leads to the estimation of the MIMO time-scattering channel in the ML sense, together with iterative algorithms that solve it. The convergence of the iterative EM algorithm for channel estimation is discussed in Section 4. The optimum detection procedure is described in Section 5. In Section 6, computer simulation results are presented that illustrate the performance of the proposed ST decoding structures. Finally, Section 7 is devoted to the conclusions.

2. Signal model

Let us consider the STC system depicted in Fig. 1. The source bit stream, \( b(l) \), is encoded into \( N \) complex symbol substreams, \( s_i(n), \) \( i = 1, \ldots, N \). This process consists, in general, of a channel coding operation that introduces temporal correlation and a serial to parallel converter that allocates the coded symbols into \( N \) different transmitters, thus creating the spatial dependence structure. In its simplest form (no channel coding), the ST encoder maps the source bits into a sequence of independent symbols and then demultiplexes this sequence into \( N \) branches. During the \( n \)th symbol period, the symbol substreams, \( s_1(n), \ldots, s_N(n) \), are fed into the bank of transmitters. \( N \) different baseband communication signals (1 per symbol substream) are generated, upconverted by a radio frequency carrier and synchronously transmitted through the wireless multipath link. At the receiving end, the signals are coherently demodulated back to baseband at each receiving element to yield an \( L \times 1 \) observation vector, \( \mathbf{x}(n) = [x_1(n), \ldots, x_L(n)]^T \). The observation drawn from the \( i \)th receiving element, \( x_i(n) \), can be written in terms of the contributions from the \( N \) transmitted symbol substreams as

\[
x_i(n) = \sum_{j=1}^{N} \sum_{l=0}^{m-1} h_{ij}^*(l)x_j(n-l) + g_i(n),
\]

where \( g_i(n) \) is a zero-mean complex Gaussian noise component with variance \( \sigma_g^2 \) and \( h_{ij}^*(l) \), \( l = 0, \ldots, m-1 \) are the discrete-time channel coefficients resulting from the multipath wireless propagation from the \( j \)th transmitting antenna to the \( i \)th receiving antenna (superindex * denotes complex conjugation). The quantity \( m \) will be referred to as the channel memory size and depends on the time-scattering introduced by the channel. Using (1), the \( n \)th observation vector can be written as

\[
\mathbf{x}(n) = \sum_{l=0}^{m-1} \mathbf{H}^*(l)\mathbf{s}(n-l) + \mathbf{g}(n)
\]
where superindex $^H$ denotes Hermitian transposition, $g(n) = [g_1(n), \ldots, g_L(n)]^T$ is an additive white Gaussian noise (AWGN) vector, $\hat{s}(n-l) = [s_1(n-l), \ldots, s_N(n-l)]^T$ is the $(n-l)$th transmitted symbol vector and

$$
\hat{H}(l) = \begin{bmatrix}
h_{11}(l) & h_{21}(l) & \cdots & h_{L1}(l) \\
h_{12}(l) & h_{22}(l) & \cdots & h_{L2}(l) \\
\vdots & \vdots & \ddots & \vdots \\
h_{1N}(l) & h_{2N}(l) & \cdots & h_{LN}(l)
\end{bmatrix}
$$

(3)

is the $N \times L$ channel matrix corresponding to the $(n-l)$th symbol vector during the $n$th symbol period. The ST channel can finally be expressed as

$$
x(n) = H^H s(n) + g(n),
$$

(4)

where

$$
H = \begin{bmatrix}
H(m-1) \\
H(m-2) \\
\vdots \\
H(0)
\end{bmatrix}
$$

and

$$
s(n) = \begin{bmatrix}
\hat{s}(n-m+1) \\
\hat{s}(n-m+2) \\
\vdots \\
\hat{s}(n)
\end{bmatrix}
$$

(5)

are the $Nm \times L$ MIMO channel matrix and the $n$th received $Nm \times 1$ symbol vector, respectively.

A channel use consists of the transmission of $K$ complex symbols in each substream, including a training sequence of $M$ symbols. The information bearing observation frame, or simply observation frame, is built with the observations that involve unknown symbols and can be expressed as

$$
x = H^H s_p + g,
$$

(6)

where $g = [g^T(M) \cdots g^T(K-1)]^T$ is the $(K-M)L \times 1$ AWGN vector,

$$
x = \begin{bmatrix}
x(M) \\
x(M+1) \\
\vdots \\
x(K-1)
\end{bmatrix}
$$

(7)

is the $(K-M)L \times 1$ observation frame,

$$
\hat{H} = \begin{bmatrix}
H(m-1) & 0 & \cdots & 0 \\
H(m-2) & H(m-1) & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & H(0)
\end{bmatrix}
$$

(8)

is the $N(K-M+m-1) \times (K-M)L$ overall MIMO channel matrix and

$$
s_p = \begin{bmatrix}
\hat{s}(M-m+1) \\
\vdots \\
\hat{s}(M) \\
\vdots \\
\hat{s}(K-1)
\end{bmatrix}
$$

(9)

Vector $s_p$ can be decomposed as $s_p = [\hat{s}^T(M-m+1) \cdots \hat{s}^T(K-1)]^T$ where $\hat{s} = [\hat{s}^T(M) \cdots \hat{s}^T(K-1)]^T$ is the symbol frame or ST codeword that contains the symbols to be detected. Thus, $s_p$ will be termed the padded ST codeword.

Finally, note that for the ST decoder to be able to recover the $N$ symbol substreams and decode them into an output bit stream without ambiguity, the overall channel matrix $H$ must have full row rank [29]. This is a realistic assumption in urban scenarios where severe multipath propagation and time-scattering effects are typically observed [20,50].

3. Semiblind channel estimation

We propose to estimate the MIMO channel matrix $H$ in the maximum likelihood (ML) sense by solving the optimization problem

$$
\hat{H} = \arg\max_{\hat{H}} \{ E_x \log(f_{x,H}(x)) \},
$$

(10)
where \( f_{x,H}(x) \) is the probability density function (p.d.f.) of a single observation vector \( x \), that depends on matrix \( H \), \( \log(\cdot) \) is the natural logarithm, \( E_x \) denotes statistical expectation with respect to (w.r.t.) the received vector \( x \) and \( \hat{H} \) is the resulting MIMO channel matrix estimate. Assuming that the AWGN is statistically independent of the transmitted symbols, the p.d.f. \( f_{x,H}(\cdot) \) reduces to (See Appendix A)

\[
f_{x,H}(x) = \frac{1}{\pi^L \sigma_y^2} E_x e^{-(1/\sigma_y^2) ||x-Hs||^2}, \tag{11}
\]

where \( E_x \) denotes statistical expectation w.r.t. the received symbol vector, \( s \), and \( ||v||^2 = v^H v \). Substituting (11) into (10) and neglecting constant terms yields the equivalent optimization problem

\[
\hat{H} = \arg \max_H \{ \mathcal{L}(H) = E_x \log(e^{-(1/\sigma_y^2) ||x-Hs||^2}) \}, \tag{12}
\]

where the cost function \( \mathcal{L}(H) \) will be termed the expected log-likelihood of the MIMO channel. The expectation \( E_x \) can be analytically obtained because in digital communications the symbols are usually modelled as discrete random variables with known finite alphabet p.d.f. As a consequence, given an arbitrary function \( \phi(s) \), the expectation w.r.t. \( s \) reduces to a simple summation

\[
E_x \phi(s) = \sum_j f_{s}(s_j) \phi(s_j), \tag{13}
\]

where \( f_{s}(\cdot) \) is the joint p.d.f. of the received symbols and \( s_j \) is the \( j \)th possible realization of the received symbol vector. When the received symbols are statistically independent and identically distributed (i.i.d.) and the modulation format has \( M \) levels, there are \( M^{Nm} \) different realizations of \( s \) and \( f_{s}(s_j) = M^{-Nm} \forall j \). In general, however, \( f_{s}(\cdot) \) may take more complicated forms because it depends on the spatio-temporal correlation induced by the ST encoder. The expectation \( E_x \), on the other hand, cannot be analytically derived and must be estimated from the actually received observation vectors. Thus, the channel estimate \( \hat{H} \) is obtained in practice as

\[
\hat{H} = \arg \max_H \left\{ \mathcal{L}(H) = \sum_{n=0}^{K-1} \log(e^{-(1/\sigma_y^2) ||x(n)-H^H s(n)||^2}) \right\}, \tag{14}
\]

where \( \mathcal{L}(H) \) is an estimate of the expected log-likelihood function. Notice that when the MIMO channel is not time-scattering and the transmitted symbols are i.i.d., \( \hat{L}(\cdot) \) is also the true likelihood of the MIMO channel matrix. Problem (14) leads to a totally blind estimation method because we have not exploited the availability of the training sequence, \( S = [s(0) \cdots s(M-1)] \), yet. When this information is taken into account, we arrive at the semiblind estimator

\[
\hat{H} = \arg \max_H \left\{ \mathcal{L}(H) = \sum_{n=0}^{K-1} \log(e^{-(1/\sigma_y^2) ||x(n)-H^H s(n)||^2}) \right\} \tag{15}
\]

where notation \( s(n) | S \) denotes conditioning of the \( n \)th received symbol vector to the known symbols. Notice that, for \( M = 0 \), problem (15) reduces to the blind approach (14) whereas, for \( M = K \), (15) is simply the LS estimator.

3.1. The EM algorithm

Problem (15) does not have a closed form solution and, therefore, some iterative optimization procedure is required to compute \( \hat{H} \). The EM algorithm [19] is a suitable iterative method for ML estimation when direct maximization of thelikelihood is not feasible. Together with the available observations, the EM paradigm postulates the existence of some unobserved data that, if known, would simplify the parameter estimation problem. Let us define the complete data set as the union of the observed and unobserved data sets. The algorithm consists of a two-step iteration:

- use the observed data and the current parameter estimate to compute sufficient statistics of the complete data (E step) and
- re-estimate the parameter using the computed complete data sufficient statistics (M step).

In our problem, the observed data are the observation vectors \( x(0), \ldots, x(K-1) \) and the complete data are the extended observations \( x_c(n) = [s^T(n) \ x^T(n)]^T \), \( n = 0, \ldots, K-1 \). Using this notation, the log-likelihood of a single observation vector, \( x(n) \), can be
written as [19]
\[
\log(f_{x|H}(x(n))) = U(H, \hat{H}, n) - V(H, \hat{H}, n),
\]
where
\[
U(H, \hat{H}, n) = E_{x(n)|x(n); \hat{H}} \log(f_{x|H}(x(n))),
\]
\[
V(H, \hat{H}, n) = E_{x(n)|x(n); \hat{H}} \log(f_{x|H}(x(n))).
\]
Using (16) the estimate of the expected log-likelihood can be rewritten as
\[
\hat{\mathcal{L}}(H) = \sum_{n=0}^{K-1} U(H, \hat{H}, n) - V(H, \hat{H}, n)
\]
\[
- K \log \left( \frac{1}{\pi^2 \sigma_y^2} \right)
\]
\[
= \mathcal{A}(H, \hat{H}) - \mathcal{J}(H, \hat{H}) - K \log \left( \frac{1}{\pi^2 \sigma_y^2} \right)
\]
where \( \mathcal{A}(H, \hat{H}) = \sum_{n=0}^{K-1} U(H, \hat{H}, n) \) and \( \mathcal{J}(H, \hat{H}) = \sum_{n=0}^{K-1} V(H, \hat{H}, n) \). Applying Jensen’s inequality and the concavity of the logarithmic function it can be shown that \( V(H, \hat{H}, n) \leq V(H, \hat{H}, n) \) \( \forall H \) [19] and, subsequently,
\[
\mathcal{J}(H, \hat{H}) \leq \mathcal{J}(\hat{H}, \hat{H}).
\]
The above relationship allows to prove that the sequence of channel estimates
\[
\hat{H}_{t+1} = \arg \max_H \{ \mathcal{A}(H, \hat{H}) \}
\]
is monotonically non decreasing in the expected log-likelihood. Function \( \mathcal{A}(H, \hat{H}) \) constitutes a set of complete data sufficient statistics and Eq. (21) comprises the two steps of the EM algorithm into a single recursive rule.

The complete data p.d.f., \( f_{x,H}(\cdot) \), can be written as (see Appendix A)
\[
f_{x,H}(x(n)) = f_s(s(n)) \frac{1}{\pi^2 \sigma_y^2} e^{-\left(1/\sigma_y^2\right)||x(n)-H^s(s(n))||^2}.
\]
Substituting (22) into \( \mathcal{A}(H, \hat{H}) \) and conditioning the statistical expectation to the training sequence \( \hat{S} \) yields the equivalent recursive rule
\[
\hat{H}_{t+1} = \arg \min_H \left\{ \sum_{n=0}^{K-1} E_{s(n)|x(n); \hat{S}} \mathcal{A}(H) | x(n) - H^s(s(n)) |^2 \right\}
\]
where we have neglected constant terms and taken into account that the only random part in \( x_s(n) | x(n) \) is the received symbol vector \( s(n) \). At this point, we have cast problem (15), which does not have a closed form solution, into a sequence of quadratic problems that can be analytically solved as
\[
\hat{H}_{t+1} = \left( \sum_{n=0}^{K-1} E_{s(n)|x(n); \hat{S}} \hat{H} s(n)^{T}(n) \right)^{-1}
\]
\[
\times \left( \sum_{n=0}^{K-1} E_{s(n)|x(n); \hat{S}} \hat{H} s(n) x^{T}(n) \right).
\]
To compute the conditioned expectations in the previous formula the p.d.f. \( f_{s|x(n)}(\cdot) \) is required. This function can be expressed in terms of \( f_{x|H}(\cdot) \), \( f_{s|x(n)}(\cdot) \), \( \hat{H} \) and \( \hat{S} \) using the Bayes theorem. Notice that \( f_{s|x(n)}(\cdot) \) can be easily obtained from (11) by simply dropping the expectation \( E_s \). As a result, we arrive at the following relationship:
\[
E_{s(n)|x(n); \hat{H}} \phi(s(n))
\]
\[
= \frac{E_{s(n)} e^{-\left(1/\sigma_y^2\right)||x(n)-\hat{H}^s(s(n))||^2} \phi(s(n))}{E_{s(n)} e^{-\left(1/\sigma_y^2\right)||x(n)-\hat{H}^s(s(n))||^2}}.
\]
Finally, let us observe that the above expression drastically simplifies for \( n=0, \ldots, M-1 \) due to the knowledge of the training sequence \( \hat{S} \)
\[
\sum_{n=0}^{M-1} E_{s(n)|x(n); \hat{S}} \phi(s(n))
\]
\[
= \sum_{n=0}^{M-1} \phi(s(n)), \quad n = 0, \ldots, M-1
\]
thus reducing the number of operations required to compute \( \hat{H}_{t+1} \).
3.2. The decision-directed EM algorithm

The EM algorithm computational complexity grows exponentially with the number of transmitting antennae, \( N \), and the channel memory size, \( m \), due to the need of computing statistical expectations w.r.t. the received symbol vector \( s(n) \). An important simplification, however, can be achieved if we realize that the conditioned expectation \( E_{s(n)|x(n)}[s(n)] \) is actually the nonlinear mean squared (NMS) [36] estimate of \( s(n) \) obtained from the corresponding observation vector, \( x(n) \), and the current channel estimate, \( \hat{\mathbf{H}}_i \). Therefore, the channel estimation algorithm will still work if we substitute \( E_{s(n)|x(n)}[s(n)] \) and \( E_{s(n)|x(n)}[s(n)s^H(n)] \) by \( \hat{s}_i(n) \) and \( \hat{s}_i(n)\hat{s}_i^H(n) \), respectively, where \( \hat{s}_i(n) \) is a conditional estimate of \( s(n) \) computed using \( \hat{\mathbf{H}}_i \) and the available observations. In this way, we arrive at the decision-directed expectation-maximization (DDEM) algorithm

\[
\hat{s}_i(n) = \text{Detector}\{x(0), \ldots, x(K-1); \hat{\mathbf{H}}_i\}
\]

\[
\hat{\mathbf{H}}_{i+1} = \left( \sum_{n=0}^{M-1} s(n)s^H(n) + \sum_{n=M}^{K-1} \hat{s}_i(n)\hat{s}_i^H(n) \right)^{-1}
\times \left( \sum_{n=0}^{M-1} s(n)s^H(n) + \sum_{n=M}^{K-1} \hat{s}_i(n)\hat{s}_i^H(n) \right).
\]

(27)

This algorithm employs the last update of the channel estimate to detect the transmitted symbols and, afterwards, uses the detected symbols to improve the channel estimate. The initial channel estimate, \( \hat{\mathbf{H}}_0 \), can be easily obtained from the training sequence \( \mathbf{S} \).

As any other decision directed procedure, the DDEM algorithm is sensitive to the choice of the initial conditions and, therefore, we can expect that the longer training sequences are, the better performance will be attained. On the other hand, if the computation of \( \hat{s}_i(n) \) is carried out in an inexpensive way (e.g., linear detection) the complexity of the DDEM algorithm reduces drastically w.r.t. the original EM algorithm. In addition, the algorithm combines channel estimation and data detection into a single stage because symbol estimates, \( \hat{s}_i(n) \), are iteratively computed as the algorithm proceeds. Thus, the method may lend itself to the implementation of low-complexity semiblind ST decoders.

In Section 6, we will consider linear minimum mean square error (LMMSE) filtering, followed by projection onto the symbol alphabet, for the detection part of the DDEM algorithm. Assuming that the transmitted symbols are zero-mean, the LMMSE detector computes soft estimates as

\[
\hat{s}_{\text{LMMSE}} = \arg \min \{ E_{\mathbf{x}}[\mathbf{x} - \hat{\mathbf{H}}_p\mathbf{s}]^2 \}
\]

\[
= [(\hat{\mathbf{H}}_p\mathbf{R}_p\hat{\mathbf{H}} + \sigma^2_I R_{L,M})^{-1}\hat{\mathbf{H}}_p^H \mathbf{R}_{sp}]^H \mathbf{x}
\]

(28)

where \( \hat{\mathbf{H}}_i \) is an estimate of the overall MIMO channel matrix built from the channel estimate \( \hat{\mathbf{H}} \), \( \mathbf{R}_p = E_{\mathbf{s}\mathbf{s}_p^H} \) is the padded ST codeword autocorrelation matrix, \( \mathbf{R}_{sp} = E_{\mathbf{s}\mathbf{s}_p^H} \) is the cross correlation matrix between the padded and non-padded ST codewords and \( I_{L(M-K)} \) denotes the \( L(M-K) \times L(M-K) \) identity matrix. When the transmitted symbols are uncorrelated and they have equal power, the above correlation matrices reduce to

\[
\mathbf{R}_p = \begin{bmatrix} \sigma^2_s I_{N(M-1)} & 0_{N(M-1)\times N(K-M)} \\ 0_{N(K-M)\times N(M-1)} & \sigma^2_s I_{N(K-M)} \end{bmatrix}
\]

and

\[
\mathbf{R}_{sp} = \begin{bmatrix} 0_{N(M-1)\times N(K-M)} \\ \sigma^2_s I_{N(K-M)} \end{bmatrix}.
\]

(29)

Notation \( 0_{r\times c} \) denotes the zero matrix of the indicated dimensions, \( I_r \) is the \( r \times r \) square matrix with all elements equal to 1 and \( \sigma^2_s = E[s_i(n)^2] \forall i, n \) is the transmitted symbol power.

Detector (28) is an extension of the well-known Wiener filter [13] to the ST signal model under consideration. The main difficulty in the derivation of (28) arises from the (possible) existence of statistical dependence among the symbols in \( \mathbf{s} \), which may render the exact computation of \( \mathbf{R}_p \) and \( \mathbf{R}_{sp} \) somehow involved.

3.3. Deterministic ML methods

In order to derive the EM channel estimation algorithm (24), we have taken a stochastic ML approach where the transmitted symbols are dealt with as random variables. In this way, channel estimation and
data detection are decoupled. It is also possible to take a deterministic ML approach in order to jointly estimate the channel coefficients and the transmitted codeword. Solving this joint problem via the space alternating generalized EM (SAGE) algorithm [6,53] leads to the decision directed procedure termed iterative least squares with enumeration (ILSE) [31]. The ILSE algorithm proceeds in two alternating steps: (1) ML estimation of the transmitted symbols, via enumeration, using the current channel estimate and (2) LS channel reestimation using the symbol estimates.

For reasons of computational complexity, the enumeration procedure in the data detection step can be substituted either by a tree search method [49] (when transmission is synchronous and the MIMO channel is non-dispersive) or by the Viterbi algorithm (when transmission is asynchronous or the channel introduces ISI). The latter choice leads to the extension for MIMO systems of the generalized ML sequence detection and channel estimation (GMLSDE) iterative scheme introduced in [53] for single input and single output (SISO) channels.

Although derived in a different way, the DDEM algorithm proposed in this paper can be considered as an extension of the ILSP algorithm in which: (a) ISI is specifically taken into account and (b) the data detection step is not constrained to any particular method, so it can be matched to the system as illustrated in Section 6. Note that the DDEM method reduces to the ILSP algorithm when considering no ISI ($m = 1$) and the LS criterion, followed by projection, is used for the detection step in (27).

4. Convergence of the EM algorithm

There is an inherent ambiguity in the totally blind estimation of the MIMO channel matrix that may cause an arbitrary permutation of the rows in $\hat{H}$ w.r.t. the true channel matrix, $H$. Indeed, the received observation vectors can be written either as in Eq. (4) or, alternatively, as

$$x(n) = (PH)^{H}(P^{-1}s(n)) + g(n).$$

where $P$ is an arbitrary, and unknown, $Nm \times Nm$ permutation matrix. As a consequence, $H$ cannot be distinguished from $PH$ using the observations $x(n)$ alone and the estimate $\hat{H}$ may correspond to $PH$ instead of $H$. In this case, detection of the transmitted symbols becomes considerably involved. In addition, it is well known that, if the likelihood function presents undesired local maxima, the EM algorithm may suffer from local misconvergence problems [19] that make blind estimation very dependent on the choice of the initial condition, $\hat{H}_0$.

Although a rigorous proof would involve the analytical maximization of $\hat{S}(H)$, which is not possible, the computer simulation results presented in Section 6 show that the semiblind estimator incorporating short training sequences is able to practically avoid the permutation and phase-rotation ambiguity and is also very robust to local misconvergence. This is also illustrated in this section by means of a graphical analysis of the expected log-likelihood function for a two-coefficient SISO channel ($N = 1$, $L = 1$, $m = 2$) and a numerical study of a simple MIMO channel ($N = 2$, $L = 2$, $m = 1$).

4.1. Expected log-likelihood surface for a two-coefficient SISO channel

Let us consider a simple system with one transmitting antenna ($N = 1$), one receiving antenna ($L = 1$), BPSK modulation and small ISI ($m = 2$). The resulting channel matrix, $H$, reduces in this case to a two-coefficient vector, $h = [h_1 \ h_2]$. In order to allow graphical representation, it is necessary to assume that both $h_1$ and $h_2$ are real-valued. In particular, let us set $h_1 = 0.1$ and $h_2 = 0.4$. Data are transmitted in frames of $K = 150$ binary symbols, including a training sequence of length $M$ at the beginning of the data burst. The average signal-to-noise ratio is set to 10 dB.

\[^{6}\text{P presents exactly one non-zero element in each row and column. This non-zero element has unit amplitude and an unknown phase rotation, i.e., it is of the form } e^{i\theta}, \text{ where } \theta \text{ is unknown.}\]
Fig. 2(a) plots the likelihood function $\hat{L}([0.1\ 0.4])$ when $K = M = 150$. In this case, the ML criterion reduces to LS estimation and we observe a purely quadratic surface with a single maximum (labeled with $\circ$) that corresponds to a very accurate estimate of the true channel (labeled with $\otimes$).

Fig. 2(b) depicts the quadratic surface obtained when the training sequence length is $M = 15$ and conventional LS estimation of the channel coefficients is carried out (i.e., only the $K = M = 15$ observations from the training sequence are used). It can be seen how the maximum of the surface, that corresponds to the channel estimate, is shifted from the true channel location.

Fig. 2(c) represents the likelihood function for the totally blind case, when $M = 0$. The resulting surface is a considerably more complicated one, with one local minimum (labeled with $\times$), eight local maxima and four saddle points. The local maxima correspond to all possible permutations and sign changes of $\mathbf{h}$ and they are due to the ambiguity inherent to blind channel estimation, as explained at the beginning of this section. The EM algorithm may also get trapped at a saddle point, but it can be shown that a small random perturbation of the channel estimate would cause the sequence of iterations to diverge from that point [19].

Finally, Fig. 2(d) depicts the likelihood function for the semiblind case, i.e., $K = 150$ and $M = 15$. It can be seen that the local maxima corresponding to permutations and/or sign changes of the true channel are suppressed. The resulting surface presents a single maximum that corresponds to a very accurate estimate of the true channel. Moreover, the properties of the EM algorithm guarantee the convergence to this maximum [19]. We must remark that the suppression of local maxima is not ensured for a particular value of $M$, since it depends on the other system parameters and the channel itself, but it is always achieved as $M$ increases.

4.2. Expected log-likelihood surface for a four-coefficient MIMO channel

As a second example, we tackle the numerical study of the expected log-likelihood in a MIMO channel with two transmitting and receiving antennae ($N = L = 2$), BPSK modulation and no ISI ($m = 1$). Hence, the channel is represented by a $2 \times 2$ matrix that we set as $\mathbf{H}_t = \begin{bmatrix} 0.90 & -0.30 \\ 0.40 & -0.80 \end{bmatrix}$. As before, the frame size is $K = 150$, including $M$ known symbols per
transmitting antenna, and the average signal-to-noise ratio is 10 dB. Due to the increased dimensionality of the system, it is not possible to graphically represent $\mathcal{L}(\hat{H})$ as in the previous subsection. Instead, we show the numerical values (with a precision of $\pm 0.01$) of the points of interest in the four-dimensional space, together with the corresponding magnitude and slope of the expected log-likelihood function. In summary, the results are analogous to the SISO case.

Table 1 shows the single maximum of $\mathcal{L}(\hat{H})$ when $K = M = 150$. As in the SISO case, the ML criterion reduces to the LS method and a very accurate estimate of $\hat{H}_r$ is obtained. The performance loss that occurs in practice when a LS estimate of the channel is computed using only the available training sequence is illustrated by Table 2. The maximum of $\mathcal{L}(\hat{H})$ when only the initial $M = 15$ observation vectors are processed by the channel estimator is indicated.

In the totally blind case ($K = 150$ and $M = 0$) there are eight local maxima corresponding to the different permutations and sign changes of $\hat{H}_r$. They are shown, together with the associated value of the expected log-likelihood and its gradient, in Table 3. Notice, however, the improved accuracy w.r.t. the LS estimator with $M = 15$. This is because the ML estimator makes use of the whole frame of available data.

Finally, we study the semiblind case, i.e., $K = 150$ and $M = 15$. Table 4 shows the gradient $\nabla_H \mathcal{L}$ evaluated at each one of the eight former local maxima. It is clearly observed that the slope of the expected log-likelihood function is non negligible except for the case $\mathbf{P}^T = \mathbf{I}_2$ (i.e., no permutation), where the gradient practically vanishes. This is analogous to the SISO example, and shows that the transmission of the short training sequence is enough to remove any local maxima associated to permutations or sign inversions of the coefficients in $\hat{H}_r$. Again, we must remark that, although the elimination of the non-desired local maxima does not depend on the choice of $M$ alone, it is always achieved as this parameter is increased.

5. Maximum likelihood ST decoding

The problem of ST decoding can be formulated as the detection of the transmitted ST codeword. Assuming that some sort of ST trellis code is employed, optimum decoding is achieved by ML detection of the codeword, $\mathbf{s}$ [34]. The decoder can be implemented with a dynamic programming algorithm (e.g., the Viterbi algorithm [28]) that searches the path with the least cost in a trellis structure that accounts for both the channel and the encoder memory.

Given a MIMO channel estimate $\hat{H}$, the ML decoder is

$$\hat{\mathbf{s}}_{ML} = \arg \max \{ \log(f_{\mathbf{x}|\mathbf{s}}(\mathbf{x})) \}$$

$$= \arg \min \{ \theta_{ML}(\mathbf{s}) = ||\mathbf{x} - \hat{H}^H \mathbf{s}_p||^2 \},$$

(31)
Table 3
Maximum of the expected log-likelihood function in the \([N=L=2, m=1]\) MIMO channel with \(K=150\) and \(M=0\) (blind ML estimation)

<table>
<thead>
<tr>
<th>Channel permutation</th>
<th>Channel estimate</th>
<th>Log-likelihood</th>
<th>Gradient</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\mathbf{P}^T) = [1 0 0 1]</td>
<td>(\hat{\mathbf{H}} = \begin{bmatrix} 0.91 &amp; -0.32 \ 0.40 &amp; -0.83 \end{bmatrix})</td>
<td>(\frac{1}{\lambda} \mathbf{L}^\prime(\hat{\mathbf{H}}) = -4.56)</td>
<td>(\frac{1}{\lambda} \nabla_{\mathbf{H}} \mathbf{L}^\prime(\hat{\mathbf{H}}) = \begin{bmatrix} 0.02 &amp; 0.02 \ 0.02 &amp; 0.01 \end{bmatrix})</td>
</tr>
<tr>
<td>(\mathbf{P}^T) = [0 1 1 0]</td>
<td>(\hat{\mathbf{H}} = \begin{bmatrix} 0.40 &amp; -0.83 \ 0.91 &amp; -0.32 \end{bmatrix})</td>
<td>(\frac{1}{\lambda} \mathbf{L}^\prime(\hat{\mathbf{H}}) = -4.56)</td>
<td>(\frac{1}{\lambda} \nabla_{\mathbf{H}} \mathbf{L}^\prime(\hat{\mathbf{H}}) = \begin{bmatrix} 0.02 &amp; -0.02 \ 0.01 &amp; -0.02 \end{bmatrix})</td>
</tr>
<tr>
<td>(\mathbf{P}^T) = [1 0 0 1]</td>
<td>(\hat{\mathbf{H}} = \begin{bmatrix} 0.91 &amp; -0.32 \ -0.40 &amp; 0.83 \end{bmatrix})</td>
<td>(\frac{1}{\lambda} \mathbf{L}^\prime(\hat{\mathbf{H}}) = -4.56)</td>
<td>(\frac{1}{\lambda} \nabla_{\mathbf{H}} \mathbf{L}^\prime(\hat{\mathbf{H}}) = \begin{bmatrix} 0.02 &amp; 0.02 \ 0.02 &amp; 0.02 \end{bmatrix})</td>
</tr>
<tr>
<td>(\mathbf{P}^T) = [0 1 1 0]</td>
<td>(\hat{\mathbf{H}} = \begin{bmatrix} -0.40 &amp; 0.83 \ 0.91 &amp; -0.32 \end{bmatrix})</td>
<td>(\frac{1}{\lambda} \mathbf{L}^\prime(\hat{\mathbf{H}}) = -4.56)</td>
<td>(\frac{1}{\lambda} \nabla_{\mathbf{H}} \mathbf{L}^\prime(\hat{\mathbf{H}}) = \begin{bmatrix} -0.02 &amp; 0.02 \ -0.02 &amp; 0.01 \end{bmatrix})</td>
</tr>
<tr>
<td>(\mathbf{P}^T) = [1 0 0 1]</td>
<td>(\hat{\mathbf{H}} = \begin{bmatrix} -0.91 &amp; 0.32 \ 0.40 &amp; -0.83 \end{bmatrix})</td>
<td>(\frac{1}{\lambda} \mathbf{L}^\prime(\hat{\mathbf{H}}) = -4.56)</td>
<td>(\frac{1}{\lambda} \nabla_{\mathbf{H}} \mathbf{L}^\prime(\hat{\mathbf{H}}) = \begin{bmatrix} 0.02 &amp; -0.02 \ 0.01 &amp; 0.02 \end{bmatrix})</td>
</tr>
<tr>
<td>(\mathbf{P}^T) = [0 1 1 0]</td>
<td>(\hat{\mathbf{H}} = \begin{bmatrix} 0.40 &amp; -0.83 \ -0.91 &amp; 0.32 \end{bmatrix})</td>
<td>(\frac{1}{\lambda} \mathbf{L}^\prime(\hat{\mathbf{H}}) = -4.56)</td>
<td>(\frac{1}{\lambda} \nabla_{\mathbf{H}} \mathbf{L}^\prime(\hat{\mathbf{H}}) = \begin{bmatrix} 0.02 &amp; -0.02 \ 0.01 &amp; 0.02 \end{bmatrix})</td>
</tr>
<tr>
<td>(\mathbf{P}^T) = [1 0 0 1]</td>
<td>(\hat{\mathbf{H}} = \begin{bmatrix} -0.91 &amp; 0.32 \ -0.40 &amp; 0.83 \end{bmatrix})</td>
<td>(\frac{1}{\lambda} \mathbf{L}^\prime(\hat{\mathbf{H}}) = -4.56)</td>
<td>(\frac{1}{\lambda} \nabla_{\mathbf{H}} \mathbf{L}^\prime(\hat{\mathbf{H}}) = \begin{bmatrix} -0.02 &amp; -0.02 \ -0.02 &amp; -0.01 \end{bmatrix})</td>
</tr>
<tr>
<td>(\mathbf{P}^T) = [0 1 1 0]</td>
<td>(\hat{\mathbf{H}} = \begin{bmatrix} 0.40 &amp; 0.83 \ -0.91 &amp; 0.32 \end{bmatrix})</td>
<td>(\frac{1}{\lambda} \mathbf{L}^\prime(\hat{\mathbf{H}}) = -4.56)</td>
<td>(\frac{1}{\lambda} \nabla_{\mathbf{H}} \mathbf{L}^\prime(\hat{\mathbf{H}}) = \begin{bmatrix} -0.02 &amp; -0.02 \ -0.01 &amp; -0.02 \end{bmatrix})</td>
</tr>
</tbody>
</table>

where

\[
f_{n|\mathbf{H}^\prime|\mathbf{x}}(\mathbf{x}) = \frac{1}{\pi(L-K-M)\delta^2(K-M)} e^{-(1/\delta^2)|\mathbf{x} - \hat{\mathbf{H}}^\prime \mathbf{s}_p|^2} \quad (32)
\]

is the conditional p.d.f. of the received data frame, \(\mathbf{x}\), that depends on the overall channel matrix, \(\hat{\mathbf{H}}\). The quadratic cost function \(\theta_{ML}(\mathbf{s})\) results from neglecting the terms and factors in \(\log(f_{n|\mathbf{H}^\prime|\mathbf{x}}(\mathbf{x}))\) that do not depend on \(\mathbf{s}\). It admits an additive decomposition

\[
\theta_{ML}(\mathbf{s}) = \|\mathbf{x} - \hat{\mathbf{H}}^\prime \mathbf{s}_p\|^2 = \sum_{n=M}^{K-1} \|\mathbf{x}(n) - \hat{\mathbf{H}}^\prime \mathbf{s}(n)\|^2 \quad (33)
\]

that yields the branch metrics for the Viterbi algorithm,

\[
\hat{\lambda}_n = \|\mathbf{x}(n) - \hat{\mathbf{H}}^\prime \mathbf{s}(n)\|^2 \geq 0.
\]

Note that, when \(\hat{\mathbf{H}}\) has been calculated via the EM algorithm (24), the computational load required to solve (31) is alleviated because the branch metrics \(\hat{\lambda}_n\) have already been computed in the last iteration of (24). Nevertheless, the complexity of the Viterbi algorithm grows exponentially with the number of transmitting antennae, \(N\), and the channel memory size, \(m\).

6. Computer simulations

In this section we present computer simulation results that illustrate the performance of the proposed semiblind decoding schemes. Let us consider a STC system with \(N=3\) and \(L=3\) transmitting and receiving antennae, respectively. The symbol stream, \(s(l)\), is fed into the serial-to-parallel converter that allocates the symbols into the \(N\) radio-frequency modulators in a round-robin fashion, i.e.,

\[
\begin{align*}
\mathbf{s}_1(0) &= s(0) & \mathbf{s}_1(1) &= s(4) & \mathbf{s}_1(2) &= s(7) & \cdots \\
\mathbf{s}_2(0) &= s(1) & \mathbf{s}_2(1) &= s(5) & \mathbf{s}_2(2) &= s(8) & \cdots \\
\mathbf{s}_3(0) &= s(2) & \mathbf{s}_3(1) &= s(6) & \mathbf{s}_3(2) &= s(9) & \cdots
\end{align*}
\]

Frames of \(K=150\) symbols are transmitted through each antenna. This frame includes a training sequence...
The communication signals propagate through a multipath wireless link that introduces ISI between consecutive symbols (i.e., the channel memory size is \( m = 2 \)). The signal-to-noise ratio (SNR) at the receiver is defined as

\[
\text{SNR} = \frac{\sigma_s^2 \text{Trace}(H^H H)}{L \sigma_y^2},
\]

where \( \sigma_s^2 = E|s(t)|^2 \) is the symbol power and Trace[·] denotes the matrix trace operator.

The performance of the STC system described above is illustrated when the minimum shift keying (MSK) modulation format is employed [28]. The MSK encoder can be interpreted as a trellis stage, that converts the source bit stream into a sequence of correlated symbols, followed by a radio-frequency modulator and, therefore, it fits the system structure depicted in Fig. 1. Symbol correlation has to be incorporated into the EM channel estimation algorithm when computing the statistical expectations w.r.t. the received symbol vectors, \( s(n) \). In the DDEM algorithm, projection of the soft symbol estimates onto the finite symbol alphabet is carried out as

\[
\hat{\phi} = \arg \min_{\phi \in \Pi} \min_{0 \leq i < 2N} \{ |\hat{s}_i(n) - e^{j\phi} \tilde{s}_i(n-1)|^2 \} \quad (35)
\]

\[
\hat{s}_i(n) = e^{j\phi} \tilde{s}_i(n-1)
\]

where \( \tilde{s}_i(n) \) is the soft estimate for the \( n \)th symbol transmitted from the \( i \)th antenna and \( \hat{s}_i(n) \) is the corresponding hard estimate. Finally, the ST decoder has to take into account both the modulation and the channel memory to simultaneously detect the symbols in the \( N \) transmission streams. It is implemented using the Viterbi algorithm to search for the most likely ST codeword in a trellis structure with \( 2^{N m} \) states and \( 2^N \) incoming and outgoing transitions associated to each state.

The results presented below have been averaged over 60 independent random realizations of the MIMO channel matrix, \( \mathbf{H}^{(1)}, \ldots, \mathbf{H}^{(60)} \). To obtain these

<table>
<thead>
<tr>
<th>Channel permutation</th>
<th>Channel estimate</th>
<th>Expected log-likelihood</th>
<th>Gradient</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mathbf{P}^T = \begin{bmatrix} 1 &amp; 0 \ 0 &amp; 1 \end{bmatrix} )</td>
<td>( \hat{\mathbf{H}} = \begin{bmatrix} 0.91 &amp; -0.32 \ 0.40 &amp; -0.83 \end{bmatrix} )</td>
<td>( \frac{1}{\lambda} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = -4.28 )</td>
<td>( \frac{1}{\lambda} \nabla_{\mathbf{H}} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = \begin{bmatrix} 0.02 &amp; 0.02 \ 0.02 &amp; 0.01 \end{bmatrix} )</td>
</tr>
<tr>
<td>( \mathbf{P}^T = \begin{bmatrix} 0 &amp; 1 \ 1 &amp; 0 \end{bmatrix} )</td>
<td>( \hat{\mathbf{H}} = \begin{bmatrix} 0.40 &amp; -0.83 \ 0.91 &amp; -0.32 \end{bmatrix} )</td>
<td>( \frac{1}{\lambda} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = -5.07 )</td>
<td>( \frac{1}{\lambda} \nabla_{\mathbf{H}} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = \begin{bmatrix} 0.34 &amp; -0.30 \ 0.46 &amp; -0.42 \end{bmatrix} )</td>
</tr>
<tr>
<td>( \mathbf{P}^T = \begin{bmatrix} 1 &amp; 0 \ 0 &amp; -1 \end{bmatrix} )</td>
<td>( \hat{\mathbf{H}} = \begin{bmatrix} 0.91 &amp; -0.32 \ -0.40 &amp; 0.83 \end{bmatrix} )</td>
<td>( \frac{1}{\lambda} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = -7.85 )</td>
<td>( \frac{1}{\lambda} \nabla_{\mathbf{H}} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = \begin{bmatrix} 0.18 &amp; 0.81 \ -0.31 &amp; -1.76 \end{bmatrix} )</td>
</tr>
<tr>
<td>( \mathbf{P}^T = \begin{bmatrix} 0 &amp; 1 \ -1 &amp; 0 \end{bmatrix} )</td>
<td>( \hat{\mathbf{H}} = \begin{bmatrix} -0.40 &amp; 0.83 \ 0.91 &amp; -0.32 \end{bmatrix} )</td>
<td>( \frac{1}{\lambda} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = -7.67 )</td>
<td>( \frac{1}{\lambda} \nabla_{\mathbf{H}} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = \begin{bmatrix} 1.14 &amp; -0.14 \ -1.31 &amp; -0.76 \end{bmatrix} )</td>
</tr>
<tr>
<td>( \mathbf{P}^T = \begin{bmatrix} -1 &amp; 0 \ 0 &amp; 1 \end{bmatrix} )</td>
<td>( \hat{\mathbf{H}} = \begin{bmatrix} -0.91 &amp; 0.32 \ 0.40 &amp; -0.83 \end{bmatrix} )</td>
<td>( \frac{1}{\lambda} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = -7.77 )</td>
<td>( \frac{1}{\lambda} \nabla_{\mathbf{H}} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = \begin{bmatrix} 1.67 &amp; 0.38 \ -0.67 &amp; -0.11 \end{bmatrix} )</td>
</tr>
<tr>
<td>( \mathbf{P}^T = \begin{bmatrix} 0 &amp; -1 \ 1 &amp; 0 \end{bmatrix} )</td>
<td>( \hat{\mathbf{H}} = \begin{bmatrix} 0.40 &amp; -0.83 \ -0.91 &amp; 0.32 \end{bmatrix} )</td>
<td>( \frac{1}{\lambda} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = -7.95 )</td>
<td>( \frac{1}{\lambda} \nabla_{\mathbf{H}} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = \begin{bmatrix} 0.71 &amp; 1.34 \ 0.33 &amp; -1.11 \end{bmatrix} )</td>
</tr>
<tr>
<td>( \mathbf{P}^T = \begin{bmatrix} -1 &amp; 0 \ 0 &amp; -1 \end{bmatrix} )</td>
<td>( \hat{\mathbf{H}} = \begin{bmatrix} -0.91 &amp; 0.32 \ -0.40 &amp; 0.83 \end{bmatrix} )</td>
<td>( \frac{1}{\lambda} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = -12.35 )</td>
<td>( \frac{1}{\lambda} \nabla_{\mathbf{H}} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = \begin{bmatrix} 1.83 &amp; 1.18 \ -1.00 &amp; -1.89 \end{bmatrix} )</td>
</tr>
<tr>
<td>( \mathbf{P}^T = \begin{bmatrix} 0 &amp; -1 \ -1 &amp; 0 \end{bmatrix} )</td>
<td>( \hat{\mathbf{H}} = \begin{bmatrix} -0.40 &amp; 0.83 \ -0.91 &amp; 0.32 \end{bmatrix} )</td>
<td>( \frac{1}{\lambda} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = -11.57 )</td>
<td>( \frac{1}{\lambda} \nabla_{\mathbf{H}} \hat{\mathbf{P}}(\hat{\mathbf{H}}) = \begin{bmatrix} 1.51 &amp; 1.50 \ -1.44 &amp; -1.45 \end{bmatrix} )</td>
</tr>
</tbody>
</table>
matrices, we have considered a Rayleigh fading model, i.e., the channel coefficients, $h_{ij}(l)$, are complex i.i.d. Gaussian random variables with zero-mean and variance $\sigma_h^2 = 1$.

6.1. Average performance for a $3 \times 3$ system with MSK modulation

Fig. 3 shows the bit error rate (BER) attained with the considered STC system using several decoding strategies. The optimum decoder (labeled Viterbi) consists of a Viterbi detector with perfect knowledge of the MIMO channel matrix. The other curves correspond to:

- The conventional decoder, labeled $LS(15) + Viterbi$. It consists of a LS channel estimator, that uses just the training sequence (with length $M = 15$), followed by the Viterbi detector.
- The proposed EM-based decoder, labeled $EM(15, 150) + Viterbi$, that combines semiblind channel estimation using the EM algorithm (24) and Viterbi detection. Notation $EM(15, 150)$ indicates $M = 15$, the training sequence length for one antenna, and $K = 150$, the frame length for one antenna.
- The DDEM-based decoder, labeled $DDEM(15, 150) + Viterbi$. Channel estimation is carried out using the semiblind DDEM algorithm (27), that involves soft and hard symbol estimation using (28) and (35), respectively. Once the channel is estimated, the ST codeword is detected using the Viterbi algorithm.

It is apparent that both semiblind approaches outperform the conventional decoder and practically match the optimum BER.

The proposed semiblind method has an inherent advantage over conventional approaches because it makes use of the whole block of observations in order to estimate the channel parameters. The main goal of the training sequence is to guarantee the proper convergence of the EM channel estimation algorithm. For this reason, it can remain relatively short and the transmission overhead is minimized. This is illustrated in Fig. 4. The BER corresponding to the semiblind decoders, with $M = 15$, is compared with the performance achieved by the conventional decoder when the training sequence length is $M = 45$. It can be clearly seen that the conventional receiver requires a training sequence three times longer (thus falling into a very high transmission overhead) in order to attain a BER close to the one achieved by the semiblind one.

We have also extended the iterative GMLSDE scheme proposed in [53] to the MIMO case in order to compare it with the proposed DDEM space–time decoder. In Fig. 5, it is seen that both receivers perform nearly the same but the complexity of the DDEM approach is much lower because the Viterbi algorithm is only run once, after channel estimation, whereas the GMLSDE decoder requires one run of the Viterbi algorithm each time it iterates.

Finally, we consider the convergence speed of the EM-based channel estimation algorithms. Fig. 6 illustrates the convergence speed of the EM and DDEM
7. Conclusions

We have introduced a novel space–time semiblind decoding scheme that performs maximum likelihood (ML) based channel estimation and data detection. The multiple input multiple output (MIMO) time-scattering channel is estimated using a block iterative expectation-maximization (EM) algorithm that fully exploits the statistical features of the transmitted signal together with the knowledge of a small number of transmitted symbols, hence the term semiblind. Data detection is efficiently carried out using the Viterbi algorithm, that uses some side information provided by the EM algorithm in order to reduce the number of operations. The main limitation of the semiblind decoder is its computational complexity. We have addressed this problem introducing a heuristic modification of the EM algorithm that allows lower complexity implementations. Finally, computer simulation results have been presented showing that the proposed semiblind decoders clearly outperform conventional receivers that estimate the channel parameters exclusively from the known transmitted data.

Acknowledgements

This work has been supported by FEDER funds (grant number 1FD97-0082) and Xunta de Galicia (PGIDT00PXI10504PR).

Appendix A. Probability density function of an observation vector

The p.d.f. of a single observation vector, \( \mathbf{x} \) (we drop the discrete-time index for simplicity), can be expressed in terms of the p.d.f. of the received symbol vector, \( f_s(\cdot) \), and the complex AWGN vector, \( f_g(\cdot) \), which are both assumed to be known. Let us define the \((Nm+L) \times 1\) vectors \( \mathbf{s}_c = [\mathbf{s}^T, \mathbf{g}^T]^T \) and \( \mathbf{x}_c = [\mathbf{s}^T, \mathbf{x}^T]^T \). It is well known [27] that the p.d.f.’s of \( \mathbf{s}_c \) and \( \mathbf{x}_c \) are related by

\[
    f_{\mathbf{x}_c|\mathbf{H}}(\mathbf{x}_c) = \frac{f_{\mathbf{s}_c}(\mathcal{T}_H^{-1}(\mathbf{x}_c))}{|J_{\mathcal{T}_H}|},
\]

where \( f_{\mathbf{x}_c|\mathbf{H}}(\cdot) \) is the p.d.f. of \( \mathbf{x}_c \), that depends on \( \mathbf{H} \), \( f_{\mathbf{s}_c}(\cdot) \) is the p.d.f. of \( \mathbf{s}_c \), \( \mathcal{T}_H \) is the invertible

\[
\mathcal{T}_H = \frac{1}{60} \sum_{i=1}^{60} \mathcal{N}(\mathbf{H}^{(i)}),
\]
transformation
\[
\mathcal{F}_H(s_c) = \begin{bmatrix} s \\ H^H s + g \end{bmatrix} = x_c, \tag{38}
\]
\[
\mathcal{F}_H^{-1}(x_c) = \begin{bmatrix} s \\ x - H^H s \end{bmatrix} = s_c,
\]
\(J_\mathcal{F}\) is the Jacobian of \(\mathcal{F}_H(\cdot)\) and \(|\cdot|\) denotes absolute value. It is straightforward to show that \(J_\mathcal{F}\) reduces to unity, since
\[
J_\mathcal{F} = \det \begin{bmatrix}
\frac{\partial s_1}{\partial s_1} & \cdots & \frac{\partial s_1}{\partial s_{Nm}} & \frac{\partial s_1}{\partial g_1} & \cdots & \frac{\partial s_1}{\partial g_L} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\frac{\partial s_{Nm}}{\partial s_1} & \cdots & \frac{\partial s_{Nm}}{\partial s_{Nm}} & \frac{\partial s_{Nm}}{\partial g_1} & \cdots & \frac{\partial s_{Nm}}{\partial g_L} \\
\frac{\partial x_1}{\partial s_1} & \cdots & \frac{\partial x_1}{\partial s_{Nm}} & \frac{\partial x_1}{\partial g_1} & \cdots & \frac{\partial x_1}{\partial g_L} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\frac{\partial x_L}{\partial s_1} & \cdots & \frac{\partial x_L}{\partial s_{Nm}} & \frac{\partial x_L}{\partial g_1} & \cdots & \frac{\partial x_L}{\partial g_L}
\end{bmatrix}
\cdot \begin{bmatrix} I_{Nm} \\ 0_{Nm \times L} \end{bmatrix}
= \begin{bmatrix} \frac{\partial x_1}{\partial s_1} & \cdots & \frac{\partial x_1}{\partial s_{Nm}} \\
\vdots & \ddots & \vdots \\
\frac{\partial x_L}{\partial s_1} & \cdots & \frac{\partial x_L}{\partial s_{Nm}}
\end{bmatrix}
\cdot \begin{bmatrix} I_L \\ 0_{L \times L} \end{bmatrix} = 1, \tag{39}
\]
where \(s_j\) is the \(j\)th element in \(s\). If we assume that the AWGN is statistically independent of the symbols, the p.d.f. \(f_{s_c}(\cdot)\) can be factorized as
\[
f_{s_c}(\mathcal{F}_H^{-1}(x_c)) = f_s(s)f_g(x - H^H s). \tag{40}
\]
Substituting (39) and (40) into (37), the desired expression for \(f_{x,H}(\cdot)\) is obtained as the marginal p.d.f.
\[
f_{x,H}(x) = \int f_s(s)f_g(x - H^H s) \, ds
= E_s f_g(x - H^H s)
= \frac{1}{\pi L \sigma_g^2} E_s e^{-(1/\sigma_g^2)||x - H^H s||^2}. \tag{41}
\]
Expression (41) could have also been obtained, in a somehow simpler way, resorting to the total probability theorem. We have chosen the more general derivation in this appendix, however, because it allows to use the same notation already employed to obtain the EM algorithm.

References


[45] G. Wei, L. Rasmussen, R. Wyrwas, Near optimum tree-search detection schemes for bit-synchronous multiuser...


