The Expectation and Sparse Maximization Algorithm

Steffen Barembruch, Anna Scaglione, and Eric Moulines

Abstract: **In recent years, many sparse estimation methods, also known as compressed sensing, have been developed. However, most of these methods presume that the measurement matrix is completely known. We develop a new blind maximum likelihood method—the expectation-sparse-maximization (ESpaM) algorithm—for models where the measurement matrix is the product of one unknown and one known matrix. This method is a variant of the expectation-maximization algorithm to deal with the resulting problem that the maximization step is no longer unique. The ESpaM algorithm is justified theoretically. We present as well numerical results for two concrete examples of blind channel identification in digital communications, a doubly-selective channel model and linear time invariant sparse channel model.**

Index Terms: **Compressive sensing (CS), deconvolution, multipath channels, smoothing methods.**

I. INTRODUCTION

The field of compressed sensing (CS) has evolved in the recent years to tackle under-determined, but sparse linear systems. A vector $\mathbf{x}$ of dimension m is called sparse if the number of active components $\|\mathbf{x}\|_0 = r \ll m$, i.e., components that are different from 0 is small compared to m. A CS problem can then be written as

$$\min \|\mathbf{x}\|_0 \quad \text{subject to} \quad \mathbf{Y} = \mathbf{A}\mathbf{x} + \varepsilon \tag{1}$$

where Y is the (known) observation of dimension K, also called the measurements of $\mathbf{x}$. $\mathbf{A}$ is the (known) sensing or measurement matrix and ε is some Gaussian noise vector, see for example [1], [2].

However, in CS the measurement matrix $\mathbf{A}$ is assumed known. We will generalize this to a blind setting where it can be decomposed into a product of two matrices where only one matrix is known. The blind compressed sensing problem (BCS) is then given by

$$\min \|\mathbf{x}\|_0 \quad \text{subject to} \quad \mathbf{Y} = \mathbf{S}\Psi\mathbf{x} + \varepsilon \tag{2}$$

where only the matrix Ψ is assumed known. We note that this form is quite general. For example, the matrix $\mathbf{S}$ may be considered as the measurement matrix and Ψ as a basis transformation. In that case, the true parameter vector $\mathbf{b} = \Psi\mathbf{x}$ would not be sparse, and the original problem would be written as $\mathbf{Y} = \mathbf{S}\mathbf{b} + \varepsilon$. On the other hand, if Ψ is the identity matrix, then the problem is completely blind. The problem formulation differs, however, significantly from the CS problem as in (1). We consider $\mathbf{S}$ to be a unknown random variable with a known distribution. Therefore, it is obvious that in the BCS problem as in (2) more measurements are necessary to recover $\mathbf{x}$ as compared to (1). The focus of the CS problem is rather on how one may use the sparsity to reduce the necessary number of measurements while still being able to recover $\mathbf{x}$. In the BCS problem, the focus is rather on how one may use the sparsity to improve the estimate of $\mathbf{x}$ given the measurements $\mathbf{Y}$. To solve this problem, we propose what we call the expectation-sparse-maximization (ESpaM) algorithm. We will also give precise conditions for Ψ and $\mathbf{S}$ for which the estimation becomes unique if the number of measurements tends to infinity.

We will use a general and well known approach to blind estimation, namely maximum-likelihood (ML) estimation of the unknown vector $\mathbf{x}$. Without the sparsity constraint, the expectation-maximization (EM) algorithm [3] is an efficient method to derive the ML estimate, see [4] and the references therein. It is an iterative procedure repeating the expectation and the maximization step. In the first step, the expectation of the symbol sequence with respect to a given parameter estimate needs to be calculated. In the second step, the parameter estimate is updated.

However, it turns out that in general in model (2) the maximization step is no longer unique, i.e., the set of solutions forms a subspace of positive dimension. In the ESpaM algorithm, we propose to perform a sparse signal reconstruction step using methods such as matching pursuit (MP), orthogonal matching pursuit (OMP) or $\ell 1$ regularization to choose the sparsest element of this subspace as the new parameter estimate of $\mathbf{x}$ for the EM iteration.

Even though the expectation step remains unchanged, it is in general not easily implemented. Therefore, we will present the model in the most general form, but we will then rather concentrate on a number of examples for which an implementation of the expectation step is known or will be derived in this contribution (c.f. subsection IV-A).

For example, we may assume that the unknown part is a Markov chain with a finite state space, such that (2) becomes a hidden Markov model (HMM). The noise is assumed to be independently normally distributed. This setting corresponds to a blind deconvolution problem in digital communications. The expectation step is then efficiently implemented by the Baum-Welch algorithm [5].

If the state space of the Markov chain is too large, the Baum-Welch algorithm is no longer feasible. In this case, we propose to use a particle smoothing algorithm to reduce the complexity. Particle smoothing algorithms approximate the smoothing distribution, i.e., the distribution of a symbol given all the observations up to time K. Most of them are based on a particle filtering algorithm approximating the filtering distribution of $\mathbf{s}_k$, i.e., given the observations up to time k. Particle filtering has

Manuscript received February 26, 2010; approved for publication by Helmut Bölcskei, Guest Editor, June 29, 2010.

S. Barembruch and E. Moulines are with the Institut des Telecommunications, Telecom ParisTech, email: steffen.barembruch@telecom-paristech.fr, moulines@tsi.enst.fr.

A. Scaglione is with the department ECE at University of California Davis, email: ascaglione@ucdavis.edu.

already proved to be useful in many contexts [6], for example in digital communications [7], [8]. Several smoothing algorithms based on particle filtering have been proposed in the literature [6], [9]–[11]. We will exemplarily use the fixed-interval smoothing [6], which is closest to the Baum-Welch algorithm and has proved to perform well in digital communications settings [10].

To demonstrate the potential of the ESpaM algorithm, we will present two applications, both in blind channel identification for digital communications. In this background, the unknown matrix $\mathbf{S}$ corresponds to the transmitted, unknown symbol sequence stemming from a finite alphabet and $\mathbf{Y}$ to the signal at the receiver. The parameter vector $\mathbf{x}$ describes the parameters of the transmission channel.

The remainder is organized as follows. We will start with describing the background work in blind CS as well as in channel identification, in particular coupled with CS methods. We will then present the general model, followed by a short review of the EM algorithm and the presentation of the ESpaM algorithm. We will discuss why we think the ESpaM algorithm is the only efficient way to include the sparsity in the EM algorithm and we discuss theoretical properties of the algorithm. We will then present the two concrete applications for blind channel identification, a linear modulation model on firstly a time-invariant and secondly a doubly-selective channel. We finish with numerical results for both linear modulation models.

A. Background Work

The (non-blind) CS problem as in (1) has been well studied in general in the literature, see for example [1], [2]. Many methods to solve this problem have been proposed, for example the MP [12], the OMP [13], [14], or a minimization with respect to the $L1$-norm.

On the other hand, the BCS problem as in (2) has rarely been considered in this form. The recent work by Eldar and Gleichman [15] proposes methods for a different type of BCS problems. They concentrate on the case, when the sparsity basis is unknown while the measurement matrix is known. This may thus be seen as the inverse of (2). Furthermore, in contrast to our approach they consider the basis to be deterministic and demonstrate several constraints for the basis such that the recovery remains unique. The possible constraints are a finite number of potential bases, the sparsity of the basis itself or structural constraints on the basis like block diagonality and orthogonality.

CS has also been often used for the two applications in digital communications that we consider. As before, most results are on training based or non-blind channel identification, where the transmitted symbol $\mathbf{S}$ is known and thus the problem reduces to a CS problem instead of the BCS problem that we consider. Bajwa *et al.* [16] give a comprehensive overview of recent advances in what they call compressed channel sensing. One of the first applications has for example been [17] to cater for an unknown number of channel paths. Sparse frequency-selective channels occurring in underwater communications, residential ultrawideband channels and digital television channels amongst others have been considered for example in [18], [19]. Compressed channel sensing for doubly-selective channels has been for example considered in [20], [21] and for multi-antenna channels in [22] amongst many others. Rapidly time-varying sparse channels have been covered in [23], [24]. Even if the channel is not sparse, a benefit from sparse methods is possible by introducing an over-complete basis [25] to improve the estimation accuracy. Since all of these methods only consider training based identification and therefore essentially assume the measurement matrix to be known, we refer to [16], [21] for a more thorough overview of compressed channel sensing. The focus for training based methods is furthermore not only on the reconstruction or estimation of the channel, but also on the sensing or the design of the training sequence. Since we consider blind identification, we do not have influence on the design and concentrate thus solely on the reconstruction of the channel.

Blind deconvolution in general without a sparsity constraint has been often considered in the literature [26]. Those methods are often based on order moments and require a large number of observations. Most recent literature takes into account coding schemes. We will, however, consider cases without coding or without assuming the coding to be known. A coded symbol sequence is no longer a Markov chain, such that the algorithms we propose to implement the expectation step do not apply. However, as soon as it is possible to estimate the posterior distribution of $\mathbf{S}$, the ESpaM algorithm may be applied.

A common approach to blind identification of doubly-selective channels are basis expansion models [26] that introduce a basis for the parameter space such that the impulse response of the channel is represented by a linear combination of a finite number of basis vectors. The idea has been adapted by using over-complete bases rendering the estimation sparse [25]. Furthermore, many methods are based on ML criterion. In general the exact ML estimate is prohibitive in a reasonable computational complexity. This leads to applying approximate algorithms as for example the method by Salut [27]. This algorithm is an iterative ML method combining basis expansion, Kalman filtering and particle filtering. The EM algorithm [3] on which our algorithm is based is another well-known method for parameter estimation in incomplete data models and more specifically for blind identification [28], [29].

As if *et al.* [30] published one of the first and only results on a blind compressive sensing approach to deconvolution in communications. However, the method they propose is for the case without noise and of a randomly precoded signal where each symbol is drawn from $\mathbb{R}$. Their algorithm is an optimization procedure over the joint space of the signal and the channel impulse response. This space is, however, not convex if the symbol alphabet is finite and therefore not applicable in the two applications we consider.

II. MODEL DESCRIPTION

We consider the sparse linear model

$$\mathbf{Y} = \mathbf{S}\Psi(\theta)\mathbf{x} + \varepsilon \tag{3}$$

where the vector $\mathbf{x} = (x_1, \cdots, x_Q)^T$ of size $Q \times 1$ is considered to be sparse, i.e., $r = \|\mathbf{x}\|_0 \ll Q$. The observations or measurements of length K is given by $\mathbf{Y} = (y_1, \cdots, y_K)^T$ and the noise is given by $\varepsilon = (\varepsilon_1, \cdots, \varepsilon_K)^T$. The unknown matrix $\mathbf{S}$ has K rows and n columns. Each of the $K \times n$ entries lies in some state space $\mathcal{X}$, such that the state space of $\mathbf{S}$ is $\mathcal{S} = \mathcal{X}^{K \times \backslash}$. The

second matrix $\Psi(\theta)$ is assumed known and depends on some known parameter $\theta \in \Theta$. It is of size $n \times Q$.

We will show that the sparse EM algorithm independently splits up in the expectation and the sparse maximization step. The latter step is easily implemented for this general model formulation. However, the expectation step requires one to compute or at least estimate the posterior conditional probability distribution of $\mathbf{S}$ given the observations and an estimate $\mathbf{x}'$ of the sparse vector $\mathbf{x}$. Thus, more assumptions on the distribution of $\mathbf{S}$ and on the distribution of the noise have to be made and be known. Even then, in general calculating the posterior distribution is not feasible. We will therefore now introduce further exemplary assumptions on the model for which we may provide an efficient implementation of the expectation step. We stress that as an efficient implementation of the expectation step for the plain EM algorithm is available, the application of the sparse EM algorithm is straightforward.

If the symbol matrix $\mathbf{S}$ corresponds in fact to a sequence of symbols in time $\mathbf{s}_k$ relating to the observations y_k, then it is reasonable to consider the following block structure for $\mathbf{S}$:

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}_1 & 0 & \cdots & & 0 \\ 0 & \mathbf{s}_2 & 0 & \cdots & 0 \\ & & \vdots & & \\ 0 & \cdots & & 0 & \mathbf{s}_K \end{bmatrix}. \tag{4}$$

We assume that $\mathbf{s}_k$ is a time-homogeneous Markov chain on the state space $\mathcal{S} = \mathcal{X}^{\mathcal{L}}$ of dimension $1 \times L$. Each of the entries $\mathbf{s}_k = [\mathbf{s}_k(1), \cdots, \mathbf{s}_k(L)]$ lies in the alphabet $\mathcal{X}$. One could also allow $\mathbf{s}_k$ to be time-inhomogeneous, since the algorithms we present cater also for these cases as long as the transition kernels are known. The matrix $\mathbf{S}$ thus has $n = KL$ columns. In this case, the measurement matrix

$$\Psi(\theta) = \begin{bmatrix} \Psi_1(\theta) \\ \vdots \\ \Psi_K(\theta) \end{bmatrix} \tag{5}$$

is given in block form, where each block $\Psi_k(\theta)$ corresponds to one of the symbols $\mathbf{s}_k$. Furthermore, we assume that the parameter may be written as $\theta = (\theta_1, \cdots, \theta_Q)$ such that the qth column $\psi_k(\theta_q)$ of $\Psi_k(\theta)$ is a function of θ_q. Hence, the sensing matrix at time k decomposes into

$$\Psi_k(\theta) = [\psi_k(\theta_1), \cdots, \psi_k(\theta_Q)].$$

The EM algorithm is known to work well for exponential families, i.e., for the noise stemming from that family. For the ease of notation, we will assume that each of the components ε_k is independently drawn from the complex normal distribution $\mathcal{N}_{\mathbb{C}}(0, \sigma^2)$ with variance σ^2.

In many applications the sparsity of $\mathbf{x}$ may arrive naturally, see for example in subsection V-B. A second intuitive application of model (3) is, when the actual model is given by

$$\mathbf{Y} = \mathbf{S}\Psi(\lambda)\beta + \varepsilon$$

where now the parameter λ is also unknown. If the measurement function ψ is complicated, direct estimation of the unknown parameters λ_m and β_m becomes infeasible. In many cases, it can be accurately enough estimated by approximating the continuous parameter space Θ by a finite, discrete grid $\theta = (\theta_1, \cdots, \theta_Q)$, see for example subsection V-A. The grid might be chosen such that the columns of the sensing matrix $\Psi(\theta)$ are an over-complete basis for the image space of $\mathbf{S}$. This is obviously the case if rank $(\Psi(\theta)) \geq KL$, implying $Q \geq KL$. It is exact if the true parameters λ_m lie on the grid and approximate if not. Then, model (3) clearly is sparse because only those grid points close to the λ_m will have corresponding non-zero coefficients.

We will now introduce notations for the statistical quantities and necessary assumptions. Let the density of the joint posterior distribution of the symbol matrix $\mathbf{S}$ given $\mathbf{Y}$ and an estimate $\mathbf{x}'$ of the sparse vector $\mathbf{x}$ be given by $p(\mathbf{S}|\mathbf{Y}; \mathbf{x}')$.

If $\mathbf{S}$ is given sequentially as in (4), then the observation y_k at time k only depends on $\mathbf{s}_k$, such that $(\mathbf{s}_k, y_k)$ is a HMM. The observation equation is given by

$$y_k = \sum_{q=1}^{Q} \mathbf{s}_k \psi_k(\theta_q) x_q + \varepsilon_k = \mathbf{s}_k \Psi_k(\theta)\mathbf{x} + \varepsilon_k. \tag{6}$$

In the remainder, we assume that the standard deviation of the noise σ is known for the ease of presentation, but we note that it comes at almost no extra costs to include the estimation of σ in the ML estimation. Let $g_k(\mathbf{s}, y_k, \mathbf{x})$ denote the likelihood function of the observation at time step k given $\mathbf{s}_k = \mathbf{s}$, i.e.,

$$g_k(y_k|\mathbf{s}; \mathbf{x}) = \frac{1}{\pi\sigma^2} \exp\left(-\tfrac{1}{\sigma^2} |\mathbf{s}\Psi_k(\theta)\mathbf{x} - y_k|^2\right). \tag{7}$$

We assume that the transition kernel of the Markov chain $\mathbf{s}_k$ from time step k to $k+1$ is known. For $\mathbf{s}_k = \mathbf{s}$, its density is given by $q(\cdot|\mathbf{s})$, such that

$$\mathbb{P}(\mathbf{s}_{\neg+1} \in \mathbb{B}|\mathbf{s}_{\neg} = \mathbf{s}) = \int_{\mathbb{B}} \text{ı}(\mathbf{s}'|\mathbf{s})\mathbb{P}(\mathbf{s}')$$

for some measurable set B.

We denote the density of the marginal smoothing distribution of $\mathbf{s}_k$ at time step k given the complete observation vector $\mathbf{Y}$ by $p_k(\mathbf{s}_k|\mathbf{Y}, \mathbf{x})$.

III. MAXIMUM LIKELIHOOD ESTIMATION

The estimation of the unknown parameters $\mathbf{x}$ is based on maximizing the likelihood function $l(\mathbf{x})$ of the observation sequence $\mathbf{Y}$ with respect to $\mathbf{x}$. It is defined as the marginal

$$l(\mathbf{x}) = \int_{\mathcal{S}} p(\mathbf{S}, \mathbf{Y}; \mathbf{x}) d\mathbb{P}(\mathbf{S}) \tag{8}$$

by integrating over the space of possible symbol matrices.

The ML estimate of $\mathbf{x}$ is hence given by

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} l(\mathbf{x}). \tag{9}$$

Since $\hat{\mathbf{x}}$ may not be derived analytically in general, we have to resort to an iterative procedure to derive an estimate of it. The EM algorithm [3] is a well known method for ML estimation

in incomplete data models with noise stemming from the exponential family. Instead of maximizing the likelihood function directly, the EM algorithm maximizes the intermediate quantity in each iteration step. It is defined for the two parameter values $\mathbf{x}$ and $\mathbf{x}'$ as

$$Q(\mathbf{x},\mathbf{x}') = \mathbb{E}_{\mathbf{S}}\left[\log\left(\imath(\mathbf{S}|\mathbf{Y};\mathbf{x})\right)|\mathbf{Y};\mathbf{x}'\right] = \int_{\mathcal{S}} \log\left(p(\mathbf{S}|\mathbf{Y};\mathbf{x})\right) p(\mathbf{S}|\mathbf{Y};\mathbf{x}')d\mathbb{P}(\mathbf{S}). \tag{10}$$

By $\mathbb{E}_{\mathbf{S}}[\cdot|\mathbf{Y};\mathbf{x}']$ we mean expectation with respect to $\mathbf{S}$ conditional on $\mathbf{Y}$ and given the parameter value $\mathbf{x}'$. If $\mathbf{S}$ is given sequentially as in (6), then (10) reduces to

$$Q(\mathbf{x},\mathbf{x}') = \frac{1}{\sigma^2}\sum_{k=1}^{K}\int_{\mathcal{S}} \|\mathbf{s}_k\Psi_k(\theta)\mathbf{x} - y_k\|^2 p_k(\mathbf{s}_k|\mathbf{Y};\mathbf{x}')d\mathbb{P}(\mathbf{s}_\neg) + \text{const}$$

where the constant term does not depend on the unknown parameter vector $\mathbf{x}$.

It turns out that if $Q(\mathbf{x},\mathbf{x}') \geq Q(\mathbf{x}'',\mathbf{x}')$, then we know as well that $l(\mathbf{x}) \geq l(\mathbf{x}'')$, i.e., an increase in the intermediate quantity means also an increase in the likelihood. After defining an initial guess of the parameter $\hat{\mathbf{x}}^{(0)}$, the EM algorithm then consists of the two iterative steps.

1) Expectation: Calculate $Q(\mathbf{x},\hat{\mathbf{x}}^{(i)})$.
2) Maximization: Calculate $\hat{\mathbf{x}}^{(i+1)} = \arg\max_{\mathbf{x}} Q(\mathbf{x},\hat{\mathbf{x}}^{(i)})$.

The EM algorithm is known [4] to converge to a critical point of the likelihood function. If the likelihood function has several local maxima, then the EM algorithm has to be set up with several different initial parameter values to increase the probability to converge to the global maximum. In certain cases as the time-invariant channel model in subsection V-B, the convergence to the global maximum can be ensured by a certain choice of a set of initial values [31].

The expectation step consists thus of calculating the marginal smoothing probabilities $p(\mathbf{S}|\mathbf{Y};\mathbf{x})$ given some parameter estimate $\mathbf{x}$ which may for example be done with the Baum-Welch algorithm [5] in a HMM with a small finite state space.

A. Sparse Parameter Maximization

In contrast to the likelihood function, $Q(\mathbf{x},\mathbf{x}')$ can be maximized more easily. The best channel estimate is given as a solution of the following system of linear equations:

$$\mathrm{E}_{\mathrm{sy}}(\mathbf{x}') = \mathrm{E}_{\mathrm{ss}}(\mathbf{x}')\,\mathbf{x} \tag{11}$$

where

$$\mathrm{E}_{\mathrm{sy}}(\mathbf{x}') = \mathbb{E}_{\mathbf{S}}\left[(\mathbf{S}\gtrdot(\theta))^{\mathbb{H}}\mathbf{Y}\middle|\mathbf{Y};\mathbf{x}'\right] = \int_{\mathcal{S}}\Big((\mathbf{S}\Psi(\theta))^H\mathbf{Y}\Big)p(\mathbf{S}|\mathbf{Y};\mathbf{x}')d\mathbb{P}(\mathbf{S}),$$

$$\mathrm{E}_{\mathrm{ss}}(\mathbf{x}') = \mathbb{E}_{\mathbf{S}}\left[(\mathbf{S}\gtrdot(\theta))^{\mathbb{H}}\mathbf{S}\gtrdot(\theta)\middle|\mathbf{Y};\mathbf{x}'\right] = \int_{\mathcal{S}}\Big((\mathbf{S}\Psi(\theta))^H\mathbf{S}\Psi(\theta)\Big)p(\mathbf{S}|\mathbf{Y};\mathbf{x}')d\mathbb{P}(\mathbf{S}).$$

We denote by $(\cdot)^H$ the Hermitian of a complex matrix.

If the model is a HMM according to (6), then these quantities write

$$\mathrm{E}_{\mathrm{sy}}(\mathbf{x}') = \sum_{k=1}^{K} y_k \int_{\mathcal{S}} \big(\mathbf{s}_k\Psi_k(\theta)\big)^H p_k(\mathbf{s}_k|\mathbf{Y};\mathbf{x}')d\mathbb{P}(\mathbf{s}_\neg),$$

$$\mathrm{E}_{\mathrm{ss}}(\mathbf{x}') = \sum_{k=1}^{K} \int_{\mathcal{S}} \big(\mathbf{s}_k\Psi_k(\theta)\big)^H\big(\mathbf{s}_k\Psi_k(\theta)\big)p_k(\mathbf{s}_k|\mathbf{Y};\mathbf{x}')d\mathbb{P}(\mathbf{s}_\neg).$$

If $\mathrm{rk}\,(\mathrm{E}_{\mathrm{ss}}(\mathbf{x}')) = Q$, then the solution to (11) is unique and given by $(\mathrm{E}_{\mathrm{ss}}(\mathbf{x}'))^{-1}\mathrm{E}_{\mathrm{sy}}(\mathbf{x}')$. In this case, however, $\mathrm{rk}\,(\mathrm{E}_{\mathrm{ss}}(\mathbf{x}')) = \min(Q,L)$ and the solution is hence only unique if $L = Q$ as it is the case in subsection V-B. In general, $L \ll Q$ and the solution to (11) is a subspace of dimension $Q - L$. We propose thus to use a sparse algorithm to select the sparsest vector in this subspace as the new parameter estimate. That is, solving the following problem:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \mathrm{E}_{\mathrm{sy}}(\mathbf{x}') = \mathrm{E}_{\mathrm{ss}}(\mathbf{x}')\,\mathbf{x} \tag{12}$$

its Lasso problem [2], [32], [33]:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 + \lambda\left\|\mathrm{E}_{\mathrm{sy}}(\mathbf{x}') - \mathrm{E}_{\mathrm{ss}}(\mathbf{x}')\,\mathbf{x}\right\|^2 \tag{13}$$

for $\lambda > 0$ or the Basis pursuit problem [34]:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \mathrm{E}_{\mathrm{sy}}(\mathbf{x}') = \mathrm{E}_{\mathrm{ss}}(\mathbf{x}')\,\mathbf{x}. \tag{14}$$

Even if $L = Q$, sparse algorithms may be used to improve the robustness of the EM algorithm, see for example the simulation results for the time-invariant model of subsection V-B in Section VII.

Hence, we propose to solve the maximization problem of the intermediate quantity by applying a sparse algorithm like the MP [12], the OMP [13], [14] or a $\ell 1$-regularization to (11). The ESpaM algorithm on the initial parameter estimate $\hat{\mathbf{x}}^{(0)}$ is thus given by iterating the following steps:

1) Calculate $p(\mathbf{S}|\mathbf{Y};\hat{\mathbf{x}}^{(i)})$ or $p_k(\mathbf{s}_k|\mathbf{Y};\hat{\mathbf{x}}^{(i)})$ for $k = 1,\cdots,K$ in the sequential model.
2) Derive $\mathrm{E}_{\mathrm{sy}}(\hat{\mathbf{x}}^{(i)})$ and $\mathrm{E}_{\mathrm{ss}}(\hat{\mathbf{x}}^{(i)})$.
3) $\hat{\mathbf{x}}^{(i+1)} = \mathrm{Sparse}(\mathrm{E}_{\mathrm{sy}}(\hat{\mathbf{x}}^{(i)}), \mathrm{E}_{\mathrm{ss}}(\hat{\mathbf{x}}^{(i)}))$.

The function $\mathrm{Sparse}(\cdot,\cdot)$ denotes the specific sparse algorithm that takes as input the matrix $\mathrm{E}_{\mathrm{ss}}(\hat{\mathbf{x}}^{(i)})$ and the vector $\mathrm{E}_{\mathrm{sy}}(\hat{\mathbf{x}}^{(i)})$ to solve problem (11).

B. Discussion on the Sparse ML Estimation

Instead of finding a sparse solution to the actual problem (10), the sparse EM algorithm described above uses a sparsity constraint for the derivative. But since the necessary condition for a minimum of (10) is that the derivative is equal to 0, the sparsest solution to (11) will also be the sparsest minimum of (10).

Both the MP and the OMP do not guarantee to find exact sparse solutions to (11), but since the intermediate quantity is in general sufficiently smooth, a point having a gradient close to zero will also be close to the maximum of the intermediate quantity. Furthermore, it is not necessary to find the exact maximum, since as long as Q increases, the likelihood will also increase

and so the sequence of iterative parameter estimates will still converge to a critical point of the likelihood function.

If the symbol matrix $\mathbf{S}$ is known, ML estimation of the channel $\mathbf{x}$ is equivalent to minimizing

$$\|\mathbf{S}\Psi(\theta)\mathbf{x} - \mathbf{Y}\|_2^2 = \sum_{k=1}^{K} \|\mathbf{s}_k \Psi_k(\theta)\mathbf{x} - y_k\|_2^2 \qquad (15)$$

with respect to $\mathbf{x}$. A solution to this problem under additional sparsity constraints is readily available since sparse algorithms like the MP and OMP apply directly to (15). Unfortunately, this may not be generalized to the blind convolution problem. The maximization step of the EM algorithm now consists of minimizing problem (10) with respect to $\mathbf{x}$, such that $\|\mathbf{x}\|_0 \leq \kappa$ for some $\kappa < Q$. Observe that, if the symbol sequence is known, the probability distribution in (10) becomes a point mass and (10) reduces to (15).

It is not possible to rewrite (10) in matrix form such that the sparse algorithms like the MP and the OMP may be applied. $\ell 1$-regularization is still applicable. Its implementation is very complex because evaluating the right hand side of (10) is very costly. We have, however, run Monte Carlo experiments for a moderate channel order L and the numerical results did not show any performance with respect to the sparse EM algorithm presented in subsection III-A. Therefore, we strongly recommend to use the ESpaM algorithm.

A different sparse expectation-maximization type algorithm that allows to apply the MP and OMP directly to the maximization criterion may be established by using a slightly different maximization criterion. Instead of maximizing the intermediate quantity, one could maximize

$$\|\mathbb{E}[\mathbf{S}|\mathbf{Y};\mathbf{x}'] \gtrless (\theta)\mathbf{x} - \mathbf{Y}\|_{\lessgtr}^{\lessgtr} \qquad (16)$$

where the conditional expectation of $\mathbf{S}$ is given by

$$\mathbb{E}[\mathbf{S}|\mathbf{Y};\mathbf{x}'] = \int_{\mathcal{X}^{\mathcal{K}\times\mathcal{L}}} \mathbf{S}\, \mathbb{P}(\mathbf{S}|\mathbf{Y};\mathbf{x}')\mathbb{P}(\mathbf{S}). \qquad (17)$$

This is analogous to (15), where $\mathbf{S}$ is replaced by the expectation of it based on the current parameter estimate. It is also similar to the maximization-maximization algorithm, where the expectation step is replaced by a Viterbi search [35]. Now the sparse minimization methods like MP or OMP are readily applicable. However, contrarily to the EM algorithm convergence is not assured and Monte Carlo experiments again on the time-invariant model showed that the performance is clearly inferior to the ESpaM algorithm.

C. Convergence Properties of the ESpaM Algorithm

We will now discuss theoretically the convergence properties of the ESpaM algorithm.

Lemma 1: Let $\left(\hat{\mathbf{x}}^{(i)}\right)_i$ be a sequence of estimates of $\mathbf{x}$ obtained by the ESpaM algorithm.

Then, for every i

$$l\left(\hat{\mathbf{x}}^{(i+1)}\right) \geq l\left(\hat{\mathbf{x}}^{(i)}\right). \qquad (18)$$

This lemma follows directly from the fact, that the ESpaM algorithm does not alter the principle of the EM algorithm. Indeed, the intermediate quantity is still maximized. The ESpam algorithm only provides a criterion which solution in the subspace of maximal values of the intermediate quantity is preferable.

We have thus established that the ESpaM algorithm will converge (or possibly diverge if the parameter space is not compact). As for the EM algorithm, convergence to the global maximum of the likelihood function is not ensured. In general, if the measurement matrix Ψ is quadratic and of full rank, the likelihood function has isolated local maxima, i.e., the maximization of the intermediate quantity is unique. An almost immediate consequence is thus, that the true parameter is a fix point of the algorithm. However, this is not obvious if Ψ is not of full rank, since then the maximal values of the the intermediate quantity form a subspace of the parameter space.

We will now give a sufficient condition for which the true parameter $\mathbf{x}$ still remains a fix point of the ESpaM algorithm.

Assumption 1: We assume that $\mathbf{S}$ is not degenerated such that $\mathbb{E}_{\mathbf{x}}[\mathbf{S}^{\mathbb{H}}\mathbf{S}]$ has full rank. Since this is the covariance matrix of $\mathbf{S}$, we just require that there is no affine relation between two of the columns of $\mathbf{S}$.

Assumption 2: We assume that every combination of $2p$ columns of Ψ is linearly independent. p denotes again the number of non-zero coefficients in the true parameter vector $\mathbf{x}$. This requires obviously that $L \geq 2p$.

Lemma 2: Let Assumptions 1 and 2 be true. Then, the true parameter vector $\mathbf{x}$ is a fix point of the ESpaM algorithm, i.e., there exists no sparser solution.

The proof is not difficult and given in Appendix A. This result is a lot stronger than the equivalent result in non-blind sparse models, since then this result only holds true in the noiseless case. However, in the ESpaM algorithm, the noise is already taken into account in the expectation step. Hence, the problem (12) is correct. The solution will satisfy the equality constraints.

IV. SMOOTHING IN HIDDEN MARKOV MODELS

As mentioned before, the expectation step (E-step) of the EM algorithm remains the same whether using the plain EM algorithm or the ESpaM algorithm. In general, it is not readily implementable because an analytic solution of the integration is not possible and a numeric integration is not feasible since the dimension of the state space is huge. However, in many practical applications, the system model is given sequentially as in (6), i.e., the symbol sequence $\mathbf{s}_k$ is a Markov chain and the process $(\mathbf{s}_k, y_k)$ is a HMM. If the transition kernel of the hidden Markov chain $\mathbf{s}_k$ is unknown, it may be included in the parameter estimation, but we will assume here that it is known. Then, the E-step consists of calculating the marginal smoothing distributions $p_k(\cdot|\mathbf{Y};\mathbf{x})$. Owing to the immense work that has been done on particle filtering, several generic particle smoothing methods have been developed to solve this problem [6], [9], [11] that are based on a particle filter and backward iterations to approximate the smoothing distribution from the filtering distribution.

In the case of digital communications without coding or with unknown coding as in models in subsections V-A and V-B, the state space is furthermore finite. In this case, the generic particle smoothing may be improved by exploiting the structure of the state space. Because of the practical relevance of this case,

we will now present a specific discrete particle smoothing algorithm in more detail. This algorithm has also been used for the simulations. For the remainder of this section, we consider a parameter estimate $\mathbf{x}$ to be fixed.

A. Smoothing for Finite State Hidden Markov Models

If the state space $\mathcal{S}$ is reasonably small, i.e., the alphabet $\mathcal{X}$ is small and the channel order L is not too large, then the Baum-Welch algorithm [5] is a very efficient and fast implementation. Unfortunately, the sparsity constraint does not decrease the complexity of the Baum-Welch algorithm and remains $\mathcal{O}(\mathbb{1}^{\mathcal{L}}\mathcal{Q})$.

In many applications, the state space is too large for the Baum-Welch algorithm to be applicable. The complexity may, however, be significantly reduced with the help of particle smoothing [6], [10]. These smoothing algorithms rely, in general, on a particle filter approximating the marginal filtering probabilities of $\mathbf{s}_k$ given the observations $y_{1:k} = (y_1, \cdots, y_k)^T$ which we denote

$$p_{k|k}(\cdot|y_{1:k};\mathbf{x}) = \mathbb{P}(\mathbf{s}_\neg = \cdot|\curvearrowright_{k:}{}_\neg, \mathbf{x}).$$

It is approximated by using a discrete, small set of positions in the state space, called particles, and neglecting the remaining part of the state space. It is updated sequentially in time. Assume that at time step k such a set of N particles $\xi_k^i \in \mathcal{S}$ for $i \in \{1, \cdots, N\}$ with associated weights w_k^i is given, such that

$$\hat{p}_{k|k}(\mathbf{s};\mathbf{x}) = \textstyle\sum_{i=1}^N w_k^i \delta\left(\xi_k^i - \mathbf{s}\right) \tag{19}$$

approximates $p_{k|k}(\mathbf{s}|y_{1:k};\mathbf{x})$ for $\mathbf{s} \in \mathcal{S}$. The iterative update to the next time step $k+1$ is based on the standard filtering decomposition:

$$\begin{aligned} &p_{k+1|k+1}(\mathbf{s}|y_{1:k+1};\mathbf{x}) \\ &\propto \sum_{\mathbf{s}'\in\mathcal{S}} g_{k+1}(y_{k+1}|\mathbf{s}_{k+1};\mathbf{x})\, q(\mathbf{s}|\mathbf{s}') p_{k|k}(\mathbf{s}'|y_{1:k};\mathbf{x}). \end{aligned} \tag{20}$$

A deterministic approximation for the filtering distribution at time $k+1$ is then available by replacing $p_{k|k}$ by $\hat{p}_{k|k}$ giving

$$\hat{\pi}_{k+1|k+1}(\mathbf{s};\mathbf{x}) \propto \textstyle\sum_{i=1}^{\tilde{N}} \tilde{w}_{k+1}^i\, \delta\left(\tilde{\xi}_{k+1}^i - \mathbf{s}\right)$$

where $\tilde{\xi}_{k+1}^i$ denote all the $\tilde{N} \leq Nm$ possible offsprings of the current particles. The updated weights are given according to (20) by

$$\tilde{w}_{k+1}^i = \textstyle\sum_{j=1}^N g_{k+1}\left(y_{k+1}\middle|\tilde{\xi}_{k+1}^i;\theta\right) q\left(\tilde{\xi}_{k+1}^i\middle|\xi_k^j\right) w_k^j.$$

To maintain the number of particles at a feasible size, after each update step, N particles are selected from $\tilde{N}$ possible offsprings $\tilde{\xi}_{k+1}^i$. The selected particles are then again denoted by ξ_{k+1}^i. In [10], it has been shown that a random selection scheme minimizing the expected $\ell 2$-norm [36] or the Chi-Squared distance [10] outperforms deterministic schemes like the Best-Weights selection [37] for the blind ML estimation.

As mentioned before many algorithms have been developed to estimate the smoothing distribution based on the the particle filtering approximation. We will exemplarily use the fixed-interval smoothing [6] which is based on the following decomposition of the smoothing probabilities:

$$p_k(\mathbf{s}_k|\mathbf{Y};\mathbf{x}) = \sum_{\mathbf{s}'\in\mathcal{S}} \frac{p_{k+1}(\mathbf{s}'|\mathbf{Y};\mathbf{x}) q(\mathbf{s}'|\mathbf{s}_k)}{\sum_{\mathbf{s}''\in\mathcal{S}} q(\mathbf{s}'|\mathbf{s}'')\, p_{k|k}(\mathbf{s}''|y_{1:k};\mathbf{x})} p_{k|k}(\mathbf{s}_k|y_{1:k};\mathbf{x}).$$

For further information on the smoothing, see [6], [10].

V. APPLICATION: BLIND CHANNEL IDENTIFICATION

We will now consider blind channel identification in digital communications as an application of the ESpaM algorithm. We present a doubly-selective multipath channel model for which the sparsity arises because we use an overcomplete basis expansion to linearize the model. The second blind identification application is for frequency-selective channels with a sparse finite impulse response of the channel.

A. Doubly-Selective Multipath Channel

As the first example, we consider a linear modulation scheme in presence of a doubly-selective multipath channel. Let $\mathcal{X}$ be the alphabet of the modulation scheme and $a_{0:K} = (a_0, \cdots, a_K)$ a symbol sequence generated independently and uniformly from $\mathcal{X}$ at symbol rate T. We do not consider coding or assume the coding to be unknown. The analog transmitted signal is then given by

$$a(t) = \sum_{\kappa=0}^{K} a_\kappa p(t - \kappa T)$$

where p is the modulation pulse.

The impulse response of the channel with M paths is given by

$$h(t,\tau) = \sum_{m=1}^{M} \beta_m \mathrm{e}^{j\omega_m t} \delta(\tau - \tau_m)$$

where $\delta(0) = 1$ and 0 otherwise. The attenuations on each path are given by β_m, the Doppler frequencies by ω_m and the delays by τ_m. We assume that we have lower and upper bounds on the delay, $(\tau_{\min}, \tau_{\max})$, and on the Doppler frequencies, $(\omega_{\min}, \omega_{\max})$.

Then, the observation at time t is given by

$$\begin{aligned} y(t) &= \int_{\tau_{\min}}^{\tau_{\max}} h(t,\tau) a(t) \mathrm{d}\tau + \varepsilon(t) && (21) \\ &= \sum_{\kappa=0}^{K} a_\kappa \sum_{m=1}^{M} \beta_m \mathrm{e}^{j\omega_m t} p(t - \kappa T - \tau_m) + \varepsilon(t). && (22) \end{aligned}$$

If the modulation function p decays quickly such that its support lies within L taps, then after resampling at the symbol rate the model is equivalent to

$$y_k = y(kT) = \sum_{l=0}^{L-1} a_{k-l} \sum_{m=1}^{M} \phi_l(\lambda_m, k)\beta_m + \varepsilon_k \tag{23}$$

where

$$\phi_l(\lambda_m, k) = \mathrm{e}^{j\omega_m kT} p(lT - \tau_m) \tag{24}$$

with $\lambda_m = (\tau_m, \omega_m)$.

Direct estimation of the unknown parameters λ_m and β_m is not feasible, especially if M is unknown. Therefore, we introduce a grid $\theta = (\theta_1, \cdots, \theta_Q)$ of Q points on the two-dimensional space of delays and frequencies. If the actual parameters lie on the grid, the new model is equivalent. Otherwise model (23) is now approximated by

$$y_k = \sum_{l=0}^{L-1} a_{k-l} \sum_{q=1}^{Q} \phi_l(\theta_q, k) x_q + \varepsilon_k. \tag{25}$$

We define again $\mathbf{x} = (x_1, \cdots, x_Q)^T$ and introduce the further notations $\mathbf{s}_k = (a_k, \cdots, a_{k-L+1})$ and

$$\psi_k(\theta_q) = (\phi_0(\theta_q, k), \cdots, \phi_{L-1}(\theta_q, k))^T, \tag{26}$$

as well as the sensing matrix $\Psi_k(\theta)$ with qth column $\Psi_k(\theta)[q] = \psi_k(\theta_q)$. Each coefficient x_q corresponds to the attenuation of a path of channel that has the Doppler frequency and the delay of grid point q. Since we assume only a few relevant paths, most of the coefficients x_q will be equal to zero.

Then, model (25) rewrites

$$y_k = \mathbf{s}_k \Psi_k(\theta) \mathbf{x} + \varepsilon_k \tag{27}$$

for $k = 1, \cdots, K$ or in matrix form

$$\mathbf{Y} = \mathbf{S}\Psi(\theta)\mathbf{x} + \varepsilon \tag{28}$$

with the same notations as in (3)–(5).

We note, however, that the convergence results of the ESpaM algorithm hold assuming that the model is exact. However, due to the fact that we use a finite grid modeling errors will be present. These errors are, however, due to the modelling and not to the ESpaM algorithm. Since the focus of this work is on the concept of the ESpaM algorithm and not on the applications, we will not consider how to choose the modelling grid and refer to other works like [25] that discuss this choice.

B. Sparse Time-Invariant Multipath Channel

The time-invariant frequency-selective multipath channel model is similar to the doubly-selective channel model with the difference that no Doppler frequencies appear. Assume that a_k is drawn uniformly and independently from the alphabet $\mathcal{X}$ of size m. Then, model (23) reduces to

$$y_k = \sum_{l=0}^{L-1} a_{k-l} \sum_{m=1}^{M} p(lT - \tau_q)\beta_m + \varepsilon_k \tag{29}$$

$$= \sum_{l=0}^{L-1} a_{k-l} h_l + \varepsilon_k \tag{30}$$

with $h_l = \sum_{m=1}^{M} p(lT - \tau_q)\beta_m$. We denote the finite impulse response of the channel by $\mathbf{h} = (h_0, \cdots, h_{L-1})^T$. Let $\mathbf{s}_k$ regroup the L most current symbols, i.e., $\mathbf{s}_k = (a_k, \cdots, a_{k-L+1})^T$. Then, (30) rewrites

$$y_k = \mathbf{s}_k^T \mathbf{h} + \varepsilon_k. \tag{31}$$

Instead of the sparsity coming from the introduction of an overcomplete basis as model in subsection V-A, we now assume that the impulse response $\mathbf{h}$ itself is sparse, i.e., the channel order L is quite large, but only a few coefficients are unequal to zero.

This model is a special case of model (6) with $L = Q$ and

$$\Psi = [\underbrace{\mathcal{I}_L, \cdots, \mathcal{I}_L}_{K \text{ times}}]^T$$

where $\mathcal{I}_L$ denotes the identity matrix of dimension $L \times L$. In this case, the matrix form of the model may be simplified to

$$\mathbf{Y} = \mathbf{S}'\mathbf{h} + \varepsilon \tag{32}$$

where $\mathbf{S}'$ denotes the matrix with rows equal to $\mathbf{s}_1$ to $\mathbf{s}_K$.

Alternatively, if the pulse shaping filter is known, one can let $\phi_l(\theta_q) = p(lT - \tau_q)$ so that each entry of $\mathbf{x}$ directly corresponds to an attenuation and the unknown parameters are given by τ_q and $\mathbf{x}$.

VI. COMPUTATIONAL COMPLEXITY

Due to the generality of the presented model, it is not possible to give a general discussion of the computational complexity. We will therefore only consider the case of a discrete HMM, where the Baum-Welch algorithm or discrete particle smoothing as in subsection IV-A may be used to implement the expectation step. This includes the presented examples of the time-invariant channel and the doubly-selective channel.

Since the expectation step and the maximization step are independent, we will analyze their complexity separately for each of the iterations of the sparse EM algorithm.

The Baum-Welch algorithm as an implementation of the expectation step is in general of complexity $\mathcal{O}(\Updownarrow^{\in \mathcal{L}} \mathcal{K})$, since at each time step each of the m^L states involves a sum over the m^L states at the preceding time step. m denotes again the size of the modulation constellation. However, in both presented models, because of the trellis structure of the symbol sequence, the transition matrix is very sparse and an efficient implementation reduces the complexity to $\mathcal{O}(\Updownarrow^{\mathcal{L}} \mathcal{K})$, since every state only has m possible offsprings, and always m states have exactly the same offsprings.

The analysis of the complexity particle smoothing algorithm has again to be split up into two parts, the particle filtering and the smoothing correction. The particle filtering is in general of complexity $\mathcal{O}(\mathcal{N} \Updownarrow^{\mathcal{L}} \mathcal{K})$, where N is the number of particles, i.e., each particle has m^L possible offsprings. This implies as well that we use a selection method which is linear in the number of particles, which is the case for the $\ell 2$-optimal selection. The complexity of the smoothing iterations is in general unfortunately $\mathcal{O}(\mathcal{N}^{\in} \mathcal{K})$ for most smoothing algorithms.

However, in both presented models the complexity is again considerably smaller due to the sparsity of the transition matrix. It turns out that the complexity of the smoothing reduces to $\mathcal{O}(\mathcal{N} \log \mathcal{N} \mathcal{K})$ [10], where the logarithmic factor comes from sorting the particles in a particular way. The complexity of the filtering part is $\mathcal{O}(\mathcal{N} \Updownarrow \mathcal{K})$, since each particle now has m offsprings. In practice, the filtering part takes more time than the smoothing part.

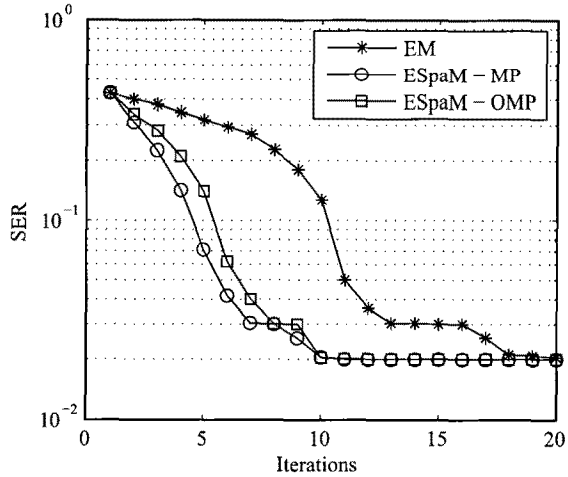

Fig. 1. Time-invariant channel: SER over iterations of EM using the maximization methods, $L = 8$, $p = 2$, SNR 12 dB.

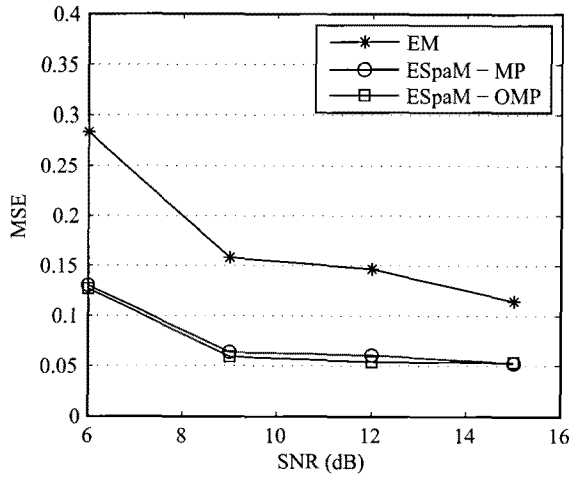

Fig. 3. Time-invariant channel: MSE of channel of EM using the maximization methods, $L = 7$, $p = 2$, over different SNR.

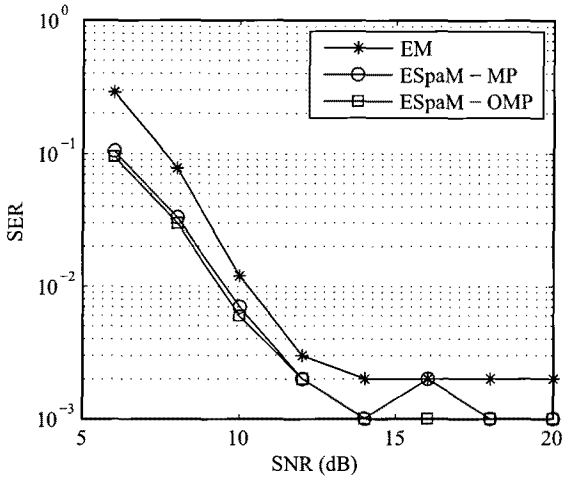

Fig. 2. Time-invariant channel: SER of EM using the maximization methods, $L = 8$, $p = 2$, over different SNR.

Only in such a model with sparse transition matrix a deterministic particle propagation has its use. Otherwise new particle positions should be randomly sampled from some importance distribution to keep the complexity at bay.

The complexity of the maximization step depends on the specific sparse algorithm. The complexity of the MP and OMP applied to (11) are about $\mathcal{O}(\mathcal{Q}^{\epsilon})$, since the matrix E_{ss} is quadratic of size $Q \times Q$. If Q is small, then $\mathcal{O}(\mathcal{N} \log \mathcal{N}\mathcal{K})$ for the particle smoothing or $\mathcal{O}(\Updownarrow^{\epsilon \mathcal{L}} \mathcal{K})$ for the Baum-Welch algorithm are much larger than $\mathcal{O}(\mathcal{Q}^{\epsilon})$. Hence, the expectation step is computationally much more complex. This is for example the case for the time-invariant channel. The maximization step becomes, however, more complex if the number of grid points Q gets too large.

VII. SIMULATIONS

A. Sparse Time-Invariant Multipath Channel

We start by considering the time-invariant channel model in subsection V-B, since in this case we have a comparable method readily at hand by using the standard non-sparse maximization step of the EM algorithm as the solution of (11). This is possible because only in this case the number of relevant symbols L is equal to the taps of the channel Q, such that E_{ss} has full rank.

We used a QPSK modulation and measured the performance in terms of the symbol error rate (SER) and the mean-squared error $\mathrm{MSE}(\hat{\mathbf{x}}) = \mathbb{E}(\|\hat{\mathbf{x}} - \mathbf{x}\|^{\not\epsilon})$ of the channel. In the figures, we refer to the exact solution to (11) as the EM, to the ESpaM using matching pursuit and the orthogonal matching pursuit solving (11) as ESpaM—MP and ESpaM—OMP, respectively. For the MP and OMP, we used $p+3$ iterations, where p is again the number of active components in the channel impulse response. This is to show, that it is not necessary to know exactly p, the algorithms work well even if they are run with more iterations. However, as we explain later the number of iterations should not be chosen too large.

Since the algorithms work completely blindly, i.e., no symbol is known, there are obviously symmetries for the estimated sequence, which are removed before calculating the SER and the MSE. The support of the sparse channel as well as its coefficients are drawn from a uniform distribution for each Monte-Carlo run. For each method, we use one single random initial parameter estimate $\hat{\mathbf{x}}^{(0)}$.

The first simulations were run with channel order $L = 8$ and $p = 2$ non-zero components. Fig. 1 shows the MSE over the first 20 EM iterations at SNR 12 dB. The sparse methods MP and OMP converge considerably faster than the exact method. The SER after convergence is also slightly smaller for the MP and OMP, see Fig. 2. In Fig. 3, we compare the MSE of the maximization methods after 20 iterations of the EM for different SNR (with $L = 7$). Even after convergence, the OMP shows still better performance than the exact method. The MP has a slightly higher MSE if the SNR is large. This might indicate that the MP introduces a slight bias, which is, however, not significant for the estimation of the SER.

These first simulations showed that for small channel orders the exact maximization still provides satisfactory results. We now turn to larger channel orders and replace the Baum-Welch algorithm by particle smoothing. Fig. 4 compares the approx-

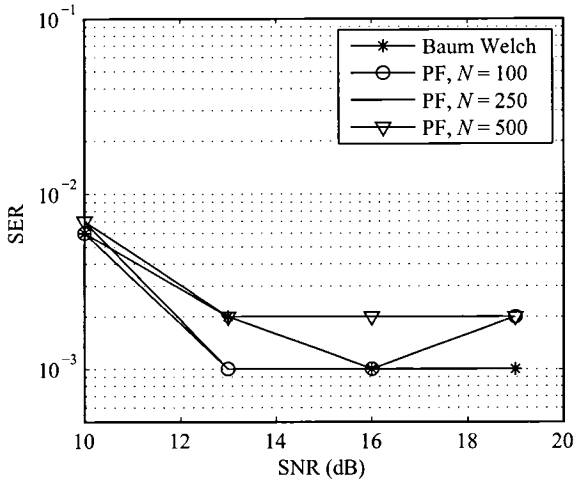

Fig. 4. Time-invariant channel: SER of ESpaM using Baum-Welch algorithm vs. particle filtering with different particle sizes, $L = 8$, $p = 2$, OMP maximization.

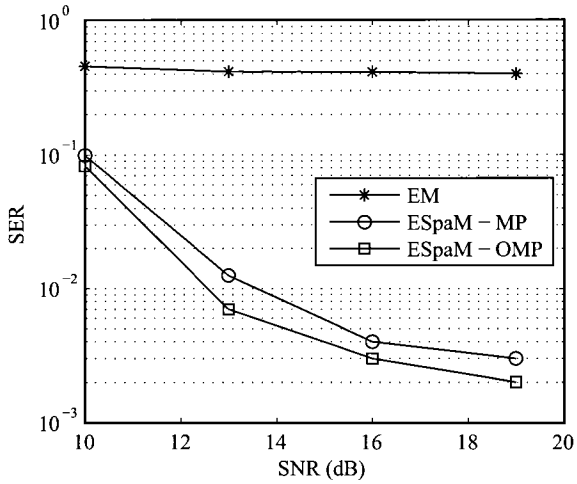

Fig. 5. Time-invariant channel: SER of particle filtering using the maximization methods, $L = 15$, $p = 3$, $N = 100$ vs. SNR.

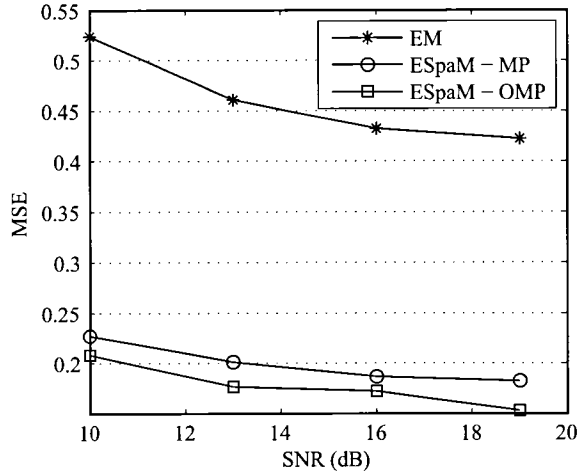

Fig. 6. Time-invariant channel: MSE of channel of particle filtering using the maximization methods, $L = 15$, $p = 3$, $N = 100$ vs. SNR.

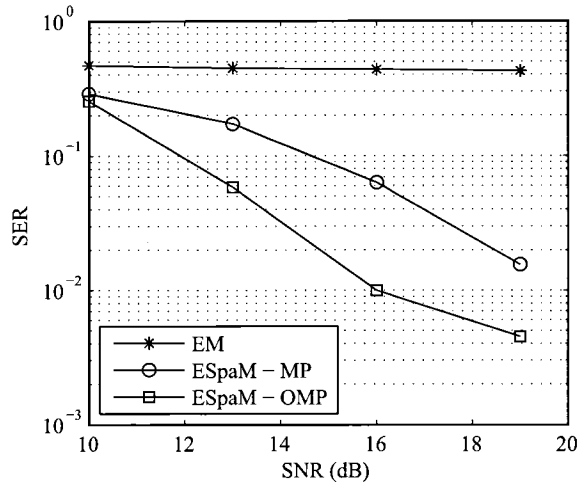

Fig. 7. Time-invariant channel: SER of particle filtering using the maximization methods, $L = 20$, $p = 4$, $N = 100$ vs. SNR.

imate EM involving particle smoothing (with different particle sizes) to the plain EM with the Baum-Welch algorithm both using the OMP maximization method. We have thus verified that there is no difference essential between the exact smoothing and the particle smoothing, i.e., the loss of using particle filtering is very moderate.

Fig. 5 shows the SER for the four maximization methods using particle smoothing for a significantly higher channel order $L = 15$ with $p = 3$ active components. It appears that the non-sparse likelihood function has now many local maxima such that the EM algorithm is not robust anymore, while the OMP and the MP still have a very low SER. The same behavior is apparent in Fig. 5 showing the MSE after 20 iterations.

Finally, we used the channel order $L = 20$ with $p = 4$ active coefficients. Fig. 7 shows the MP and the OMP are the only methods capable of tracking a channel of such a large order. The development of the MSE in Fig. 8 reveals that the exact maximization method is not converging in contrast to the MP and OMP.

Since E_{ss} has full rank, the correct sparsest solution to (11) coincides with the solution of the system of equations. Thus, the EM-algorithm coincides with an ESpaM algorithm that uses a sparse algorithm that gives the exact sparsest solution, unless there are numerical instabilities with the standard EM algorithm. However, the huge performance increase of the ESpaM algorithm coupled with OMP and MP comes from the fact that these greedy algorithms if stopped after a few number of iterations force the solution to be exactly sparse. The EM algorithm converges always, but its problem is the convergence to local maxima. Restricting the parameter space such that it only contains sparse vectors obviously avoids or eliminates many of these local maxima, such that the ESpaM algorithm with OMP or MP is much more robust with respect to convergence to local maxima. This shows, that it is essential not to choose the number of iterations of the OMP or MP to large, otherwise the convergence will be similar to the EM algorithm.

B. Doubly-Selective Multipath Channel

We now turn to the doubly-selective channel model in subsection V-A. In contrast to the time-invariant model in subsection V-B, the plain EM algorithm may not be applied since the

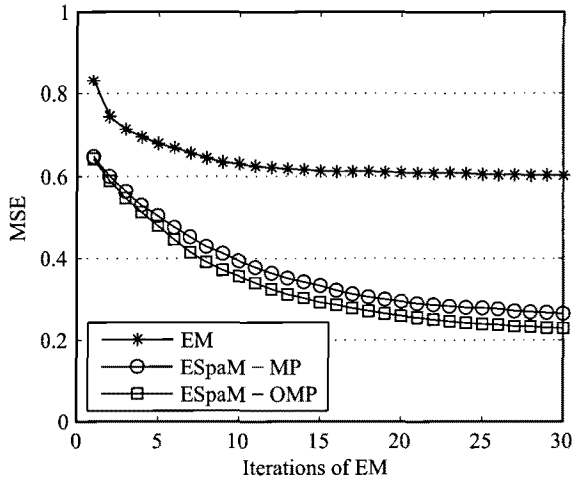

Fig. 8. Time-invariant channel: MSE over Iterations of EM using the maximization methods, $L = 20$, $p = 4$, $N = 100$, SNR 16 dB.

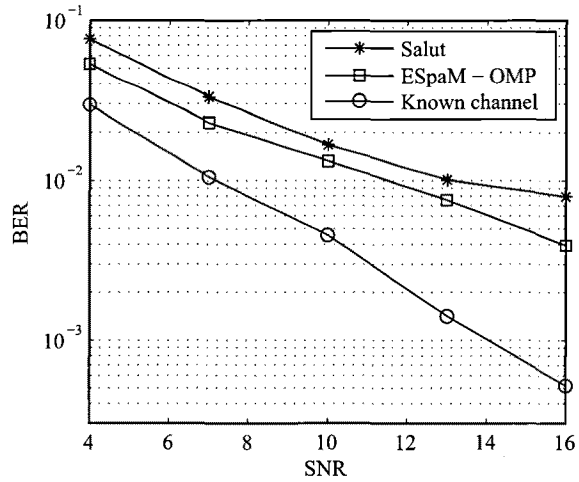

Fig. 9. Doubly selective channel, BPSK, BER over different SNRs, 2 random delays, and Doppler frequencies off grid.

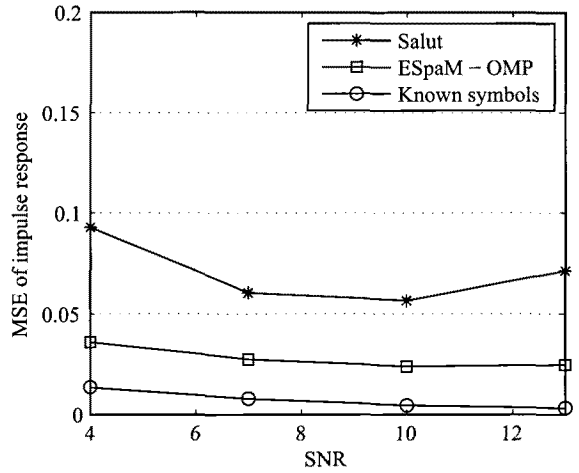

Fig. 10. Doubly selective channel, BPSK, MSE of channel impulse response averaged over time over different SNR, 2 random delays, and Doppler frequencies off grid.

matrix E_{ss} does not have full rank. This is because the number of grid points Q is much larger than the number of relevant symbols L for each observation. Therefore, the ESpaM algorithm does not only improve the performance, but is the only applicable ML method. We use the OMP algorithm in the M step.

For the following Monte Carlo experiments, we assume that the channel consists of two paths, each with a random attenuation, a random delay and a random Doppler frequency which are drawn independently at each Monte Carlo iteration. The number of observations is $K = 100$.

The SER of the ESpaM algorithm is compared to a genius bound where the Doppler frequencies and the delays are assumed to be known, i.e., by using the expected symbols $\mathbb{E}_{\mathbf{x}}[\mathbf{s}\neg|\curvearrowright\!\!\!\!\!\!\not\in\!:\!\mathbb{K}]$. Furthermore, we use the MSE of the channel impulse response which is now averaged over time. As a performance bound we will use the same sparse method but the symbol matrix is now assumed known. Then, the problem reduces to the sparse minimization problem

$$\|\mathbf{S}\Psi(\theta)\mathbf{x} - \mathbf{Y}\|_2^2$$

to which we also apply the OMP algorithm.

As mentioned before, the algorithm by Salut [27] with 16 particles is used as a comparison. The initial channel coefficients were chosen such that the coefficients corresponding to the time-constant basis vector were random, while the coefficients of the remaining basis vectors were set to 0. This was clearly superior to a completely random initialization.

As mentioned before, the algorithm by Salut [27] with 16 particles is used as a comparison. The initial channel coefficients were chosen such that the coefficients corresponding to the time-constant basis vector were random, while the coefficients of the remaining basis vectors were set to 0. This was clearly superior to a completely random initialization.

We start with a BPSK modulation and set the maximal Doppler spread to $\omega_{\max} = 1 \times e^{-2}/T$ and the maximal delay to $2T$ such that the channel order $L = 4$ is sufficient. The grid step size for the estimation is fixed to $1 \times e^{-3}/T$ for the Doppler frequencies and $0.33T$ for the delays. For each Monte Carlo run, the delays and doppler frequencies are chosen uniformly randomly in the complete range, i.e., they do not lie on the grid points. The ESpaM algorithm is run over 30 iterations until convergence. Salut's algorithm as well as the ESpaM algorithm are started with a set of two different initial parameter estimates. Fig. 9 shows the BER over different SNRs. The ESpaM algorithm is thus slightly superior to Salut's algorithm and not too far away from the BER for known channel parameters. The MSE is given in Fig. 10. The ESpaM algorithm is thus even more superior to Salut's algorithm regarding the estimation of the channel. Furthermore, in contrast to Salut's algorithm the ESpaM algorithm also gives an estimate of the Doppler frequencies and delays as well as the number of paths of the channel.

The next simulations have been run with a QPSK modulation, while the remaining parameters as Doppler frequencies, delays, and channel order have been kept the same. Fig. 11 shows again the SER over different SNRs. Obviously, Salut's algorithm is not adapted to this more complicated model, while the ESpaM algorithm still maintains a low SER.

We regard again the QPSK modulation but with a higher max-

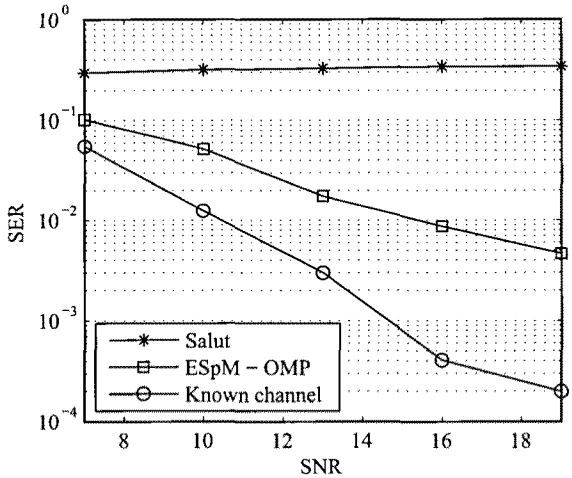

Fig. 11. Doubly selective channel, QPSK, BER over different SNRs, 2 random delays, and Doppler frequencies off grid.

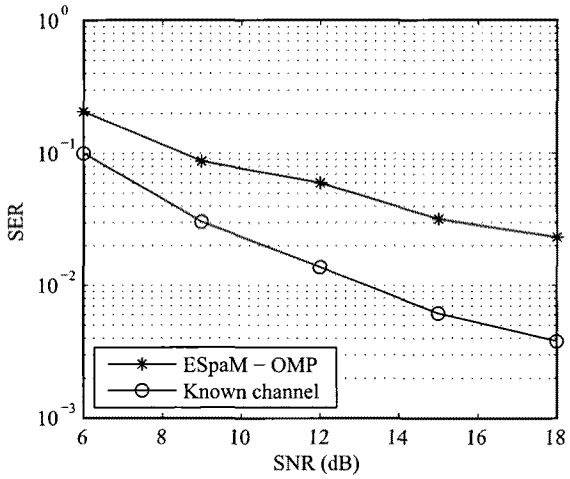

Fig. 12. Doubly selective channel, QPSK, SER over different SNRs, 2 random delays, and Doppler frequencies off grid, maximal delay $5T$, after 25 iterations of the ESpaM algorithm.

imal delay of $5T$. This is still small enough, such that the relevant number of symbols $L = 6$ is sufficiently small to apply the Baum-Welch algorithm. If it is bigger, the Baum-Welch algorithm may be replaced by a particle smoothing algorithm (see for example [6], [10] and many others). The maximal bound on the Doppler spread is again set to $\omega_{\max} = 1 \times e^{-2}/T$. The grid step size for the estimation is fixed to $1 \times e^{-3}/T$ for the Doppler frequencies and $0.33T$ for the delays. For the following simulation, one single random initialization was used for the ESpaM algorithm.

The delays and Doppler frequencies are sampled randomly in the range between minimal and maximal values. i.e., not on the grid. Fig. 12 shows the SER after 25 iterations of the ESpaM algorithm using one single random initialization. The true delays as well as the Doppler frequencies have been generated on the grid points. Obviously, the SER of the ESpaM algorithm is larger than for the known channel, but it is still satisfactorily small. The MSE, see Fig. 13, converges quickly over the iterations of the ESpaM algorithm. It can be seen that an even better performance could be achieved by using more EM iterations. Almost always the performance of the EM algorithm may be considerably improved by using several initial values, such that these results should be understood as a benchmark of the capacity of the algorithm and not as the lowest MSE or SER achievable in an actual real world implementation.

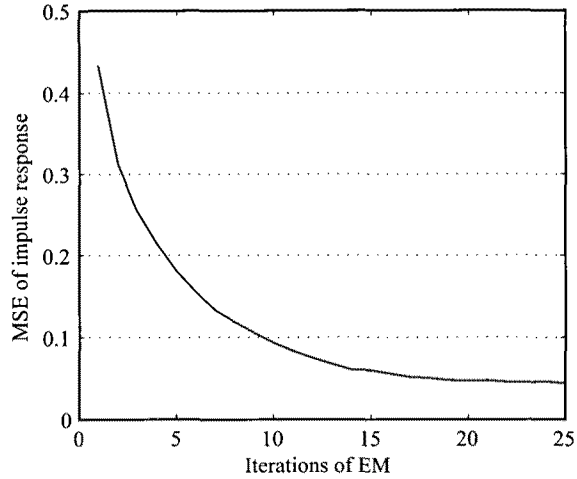

Fig. 13. Doubly selective channel, QPSK, MSE of channel impulse response averaged over time, over iterations of ESpaM, 2 random delays, and Doppler frequencies off grid, maximal delay $5T$, SNR 21 dB.

VIII. CONCLUSION

We have presented a sparse ML method that brings together recent advances in CS and blind ML estimation for a broad variety of models and applications. We have presented two models in digital communications to demonstrate the capacity of this algorithm. For the time-invariant channel, the sparse EM algorithm works well as a robust version of the plain EM algorithm, whereas in the second case of the doubly-selective channel, our method is a necessary tool to establish an EM algorithm, since the matrix in the updating equation is not invertible anymore.

The sparse methods are applied to the gradient of the actual problem, giving the advantage of a well-posed, comparably small sparse problem, that can be efficiently solved by the MP or OMP.

APPENDIX

A. *Proof of Lemma 2*

Note, that standard results of the standard EM algorithm give that $\mathbf{x}$ maximizes $Q(\cdot, \mathbf{x})$, i.e., it is a solution of (11) such that

$$\mathrm{E}_{\mathrm{sy}}(\mathbf{x}) = \mathrm{E}_{\mathrm{ss}}(\mathbf{x})\,\mathbf{x}. \tag{33}$$

To show that $\mathbf{x}$ is as well a fix point of the proposed ESpaM algorithm, we first show that every combination of $2p$ rows of $\mathrm{E}_{\mathrm{ss}}(\mathbf{x})$ is also independent. Secondly, we show that $\mathbf{x}$ is a fix point, i.e., that every other solution to (12) has more non-zero components than p.

1) We recall that the measurement matrix Ψ is of size $Q \times L$. Let T be a subset of $\{1, \cdots, Q\}$ of size $2p$ and let T^- be the

remaining indices. For a matrix A let A_T denote the submatrix of A consisting of the columns with indices corresponding to T. Without loss of generality, we assume for the ease of presentation that $T = \{1, \cdots, 2p\}$.
Then, $\mathrm{E}_{\mathrm{ss}}(\mathbf{x})$ may be decomposed as

$$\mathrm{E}_{\mathrm{ss}}(\mathbf{x}) = \begin{bmatrix} \Psi_T^H \mathbb{E}_{\mathbf{x}}[\mathbf{S}^{\mathbb{H}}\mathbf{S}]_{\geq \mathbb{T}} & \\ & \Psi_{T^-}^H \mathbb{E}_{\mathbf{x}}[\mathbf{S}^{\mathbb{H}}\mathbf{S}]_{\geq \mathbb{T}^-} \end{bmatrix}. \quad (34)$$

With Assumption 1 $\mathbb{E}_{\mathbf{x}}[\mathbf{S}^{\mathbb{H}}\mathbf{S}]$ has full rank L and with Assumption 2 the matrix Ψ_T has rank $2p$ with $L \geq 2p$. Thus, the upper left block in the block decomposition in (34) has also rank $2p$. Hence, the columns of $\mathrm{E}_{\mathrm{ss}}(\mathbf{x})$ with indices T are independent. Since this is true for every T, every combination of $2p$ rows is independent.

2) Let $\mathbf{x}'$ be a second solution of (33), and assume that $\mathbf{x} \neq \mathbf{x}'$ and that $\|\mathbf{x}'\|_0 \leq p$. Then,

$$\mathrm{E}_{\mathrm{ss}}(\mathbf{x} - \mathbf{x}') = 0.$$

Since $(\mathbf{x} - \mathbf{x}')$ has less than $2p$ non-zero components, but any $2p$ columns of $\mathrm{E}_{\mathrm{ss}}(\mathbf{x})$ are independent, it follows that $(\mathbf{x} - \mathbf{x}') = 0$. This is a contradiction.

REFERENCES

[1] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[2] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum-likelihood from incomplete data via the EM algorithm,"*J. Roy. Stat. Soc.*, vol. B39, pp. 1–38, 1977.

[4] O. Cappé, E. Moulines, and T. Rydé, *Inference in Hidden Markov Models*, Springer, 2nd ed., 2007.

[5] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Stat.*, vol. 41, pp. 164–171, 1970.

[6] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[7] E. Punskaya, *Sequential Monte Carlo Methods for Digital Communications*, Ph.D. thesis, Cambridge Univ., Cambridge, U.K., 2003.

[8] T. Ghirmai, M. F. Bugallo, J. Miguez, and P. M. Djuric, "A sequential Monte Carlo method for adaptive blind timing estimation and data detection," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2855–2865, 2005.

[9] M. Briers, A. Doucet, and S. R. Maskell, "Smoothing algorithms for state-space models," Tech. Rep., Cambridge University Engineering Department Technical Report, CUED/F-INFENG/TR.498, 2004.

[10] S. Barembruch, A. Garivier, and E. Moulines, "On approximate maximum likelihood methods for blind identification: How to cope with the curse of dimensionality," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4247 – 4259, 2009.

[11] P. Fearnhead, D. Wyncoll, and J. Tawn, "A sequential smoothing algorithm with linear computational cost," *submitted*, 2008.

[12] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, 1993.

[13] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. ACSSC*, Nov. 1993, vol. 1, pp. 40–44.

[14] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.

[15] S. Gleichman and Y. C. Eldar, "Blind compressed sensing," *submitted to IEEE Trans. Inf. Theory, CCIT Report; 759 Feb. 2010, EE Pub No. 1716, EE Dept., Technion–Israel Institute of Technology, [Online] arXiv 1002.2586.*

[16] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *to appear in Proc. IEEE*, 2010.

[17] J.-J. Fuchs, "Multipath time-delay estimation," in *Proc. ICASSP*, Apr. 1997, vol. 1, pp. 527–530.

[18] W. U. Bajwa, J. Haupt, G. Raz, and R. Nowak, "Compressed channel sensing," in *Proc. CISS*, Mar. 2008, pp. 5–10.

[19] S. F. Cotter and B. D. Rao, "Sparse channel estimation via matching pursuit with application to equalization," *IEEE Trans. Commun.*, vol. 50, no. 3, pp. 374–377, Mar. 2002.

[20] W. U. Bajwa, A.M. Sayeed, and R. Nowak, "Learning sparse doubly-selective channels," in *Proc. ACCCC*, Sept. 2008, pp. 575–582.

[21] G. Taubock, F. Hlawatsch, D. Eiwen, and H. Rauhut, "Compressive estimation of doubly selective channels in multicarrier systems: Leakage effects and sparsity-enhancing processing," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 255–271, Apr. 2010.

[22] W. U. Bajwa, A. Sayeed, and R. Nowak, "Compressed sensing of wireless channels in time, frequency, and space," in *Proc. ACSSC*, Oct. 2008, pp. 2048–2052.

[23] Y. Lui and D. K. Borah, "Estimation of time-varying frequency-selective channels using a matching pursuit technique," in *Proc. IEEE WCNC*, Mar. 2003, vol. 2, pp. 941–946.

[24] W. Li and J. C. Preisig, "Estimation of rapidly time-varying sparse channels," *IEEE J. Ocean. Eng.*, vol. 32, no. 4, pp. 927–939, Oct. 2007.

[25] M. Sharp and A. Scaglione, "Estimation of sparse multipath channels," in *Proc. MILCOM*, Nov. 2008, pp. 1–7.

[26] G. B. Giannakis and C. Tepedelenlioglu, "Basis expansion models and diversity techniques for blind identification and equalization of time-varying channels," *Proc. IEEE*, vol. 86, no. 10, pp. 1969–1986, Oct. 1998.

[27] F. B. Salem and G. Salut, "Deterministic particle receiver for multipath fading channels in wireless communications. part I: FDMA," *Traitement du Signal*, vol. 21, no. 4, pp. 347–358, 2004.

[28] W. Turin, "MAP decoding in channels with memory," *IEEE Trans. Commun.*, vol. 48, no. 5, pp. 757–763, May 2000.

[29] C. N. Georghiades and J. C. Han, "Sequence estimation in the presence of random parameters via the em algorithm," *IEEE Trans. Commun.*, vol. 45, no. 3, pp. 300–308, Mar. 1997.

[30] M. S. Asif, W. Mantzel, and J. Romberg, "Random channel coding and blind deconvolution," in *Proc. ACCCC*, 2009.

[31] H. Nguyen and B. C. Levy, "The expectation-maximization Viterbi algorithm for blind adaptive channel equalization," *IEEE Trans. Commun.*, vol. 53, no. 10, pp. 1671–1678, Oct. 2005.

[32] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *J. Roy. Stat. Soc B.*, vol. 58, pp. 267–288, 1996.

[33] D. L. Donoho, "For most large underdetermined systems of equations, the minimal L1-norm near-solution approximates the sparsest near-solution," *Comm. Pure Appl. Math*, vol. 59, pp. 907–934, 2006.

[34] S. S. Chen, D. L. Donoho, and M. L. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

[35] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[36] P. Fearnhead and P. Clifford, "On-line inference for hidden Markov models via particle filters," *J. Roy. Stat. Soc. B*, vol. 65, no. 4, pp. 887–899, 2003.

[37] J. K. Tugnait, "Detection and estimation for abruptly changing systems," in *Proc. Decision and Control including the Symposium on Adaptive Processes*, vol. 20, Dec., 1981, pp. 1357–1362.

Steffen Barembruch received the Diplom. (M.Sc. degree) in Mathematics in 2007 from the Technical University of Darmstadt, Germany. From 2004 to 2005, he spent one year at the University of St Andrews, Scotland. He is currently a Ph.D. student at Télécom ParisTech in Paris, France, under the supervision of Eric Moulines. In 2009, he was staying at University of California at Davis for several months. His research interest is in statistical signal processing for digital communications.

Anna Scaglione received the Laurea (M.Sc. degree) in 1995 and the Ph.D. degree in 1999 from the University of Rome, "La Sapienza." She is currently Professor in Electrical and Computer Engineering at University of California at Davis, where she joined as Associate Professor in July 2008. She was previously at Cornell University, Ithaca, NY, from 2001 where she became Associate Professor in 2006; prior to joining Cornell she was Assistant Professor in the year 2000-2001, at the University of New Mexico. She served as Associate Editor for the IEEE Transactions on Wireless Communications from 2002 to 2005, and serves since 2008 the Editorial Board of the IEEE Transactions on Signal Processing, where she is now Area Editor. She has been in the Signal Processing for Communication Committee from 2004 to 2009. She was general chair of the workshop SPAWC 2005 and keynote speaker in SPAWC 2008. She is the first author of the paper that received the 2000 IEEE Signal Processing Transactions Best Paper Award; she has also received the NSF Career Award in 2002 and she is co-recipient of the Ellersick Best Paper Award (MILCOM 2005). Her expertise is in the broad area of signal processing for communication systems and networks. Her current research focuses on, signal processing for wireless/wireline channel communications, cooperative wireless networks and sensors' systems for monitoring and control applications.

Eric Moulines was born in Bordeaux, France, in 1963. He received the M.Sc. degree from Ecole Polytechnique, Paris, France, in 1984, the Ph.D. degree in signal processing from Ecole Nationale Suprieure des Telcommunications (ENST), Paris, in 1990, and the "Habilitation Diriger des recherches" degree in applied mathematics (probability and statistics) from the Université René Descartes (Paris V) in 1995. From 1986 to 1990, he was a Member of the Technical Staff of the Centre National de Recherche des Télécommunications (CNET). Since 1990, he has been with ENST, where he has been a Professor since 1996. His teaching and research interests include applied probability, computational statistics, and statistical signal processing. His current research interests include time series analysis, hidden Markov models, and sequential Monte Carlo methods. He served on the editorial board of Speech Communication and the IEEE Transactions on Signal Processing. He is presently and editor of ESAIM: Probability and Statistics and Signal Processing. He is a Member of the IEEE Committee "Signal Processing: Theory and Methods."