Code-aided channel tracking for OFDM

Henk Wymeersch, Frederik Simoens, Heidi Steendam and Marc Moeneclaey
Department of Telecommunications and Information Processing Ghent University,
St.-Pietersnieuwstraat 41, B-9000 GENT, Belgium
E-mail: {hwymeers,fsimoens,hs,mm}@telin.UGent.be

Abstract

This contribution deals with the problem of tracking a frequency-selective channel for OFDM systems, where the data is protected by a powerful error-correcting code. We present an EM-based receiver, operating on a single OFDM symbol at a time, that iterates between channel estimation and data detection. The channel estimator accepts information from the detector in the form of a posteriori probabilities. The estimator is robust in a sense that it makes very few assumptions regarding the underlying channel model. Complexity-reducing approximations are discussed. Performance results are provided in the form of Monte Carlo simulations.

1 Introduction

Orthogonal Frequency Division Multiplexing (OFDM) is an efficient technique that converts a frequency-selective channel into many parallel frequency-flat sub-channels [1]. In high-rate wireless applications, this allows for simple equalization per sub-channel as well as aggressive modulation techniques (e.g., bit-loading). By using the Fast Fourier Transform (FFT) during modulation and demodulation, OFDM is also attractive from a complexity point of view.

In practice, data detection in OFDM is performed coherently, thus requiring knowledge regarding the channel state. Channel state information (e.g., the sampled Channel Impulse Response (CIR) or the sampled Channel Frequency Response (CFR)) can be obtained in a number of ways. Blind channel estimation algorithms [2], [3] exploit certain statistical properties of the transmitted signal. Due to relatively poor performance of blind techniques, most practical channel estimation algorithms are based on the presence of pilot (or training) symbols, allocated to different points in the time-frequency grid [4], [5]. Unfortunately, such Data-Aided (DA) techniques result in a waste in terms of power and bandwidth efficiency, especially when state-of-the-art error-correcting codes are employed.

The drawbacks of blind and semi-blind channel estimation techniques have led many research groups to consider the possibility of exploiting both pilot symbols and unknown coded data symbols for channel estimation/tracking. Probably the most promising technique in this respect is the Expectation Maximization (EM) algorithm [6], and its variations, whereby one iterates between channel state estimation and data detection. In the context of OFDM, we mention the following references: [7]–[11]. These papers differ in what is estimated (CFR or CIR) and how the code is exploited. The most general approach was taken in [10], [11]. However, [10], [11] require making *hard* decisions w.r.t. the coded bits, which may result in significant degradations when powerful error-correcting codes (such as turbo or LDPC codes) come into play.

Here, another approach is proposed, whereby *soft* decisions of coded bits are taken. In this contribution, we will apply the EM algorithm to develop a powerful and robust channel estimation algorithm, suitable for use with powerful error-correcting codes. Our technique is mainly based on [12]. This algorithm will then be applied to perform channel tracking in a time-varying environment.

Notations

Vectors (always column vectors) will be underlined, while matrices will be represented in bold. Time-domain quantities will be labeled with small letters, while frequency-domain quantities will be written in capitals. That way the FFT of $\underline{x}$ will be denoted $\underline{X}$. The operation $circ\left(\underline{x}\right)$ converts a length-N vector $\underline{x}$ to a circular $N \times N$ matrix with first column equal to $\underline{x}$. The operation $diag\left(\underline{x}\right)$ converts a length-N vector to a $N \times N$ matrix with $\underline{x}$ on the diagonal. $\mathbf{I}_K$ is the $K \times K$ identity matrix, while $\underline{0}_K$ is a vector consisting of K zeros. Finally, the Hermitian operator is denoted by $(.)^H$.

2 System model

We consider the following system (as depicted in Fig. 1): a vector of information bits is encoded, interleaved and mapped onto a sequence of complex symbols, belonging to a signalling constellation[1] Ω: the resulting N coded symbols will be denoted $\underline{X} \in \Omega^N$. Adopting the model from [8], this sequence is transformed by an inverse FFT,

This work has been supported by the Interuniversity Attraction Poles Program P5/11 - Belgian Science Policy. The second author gratefully acknowledges the support from FWO-Vlaanderen.

[1]For convenience each sub-carrier will use the same constellation. Extension to the more general case is trivial.

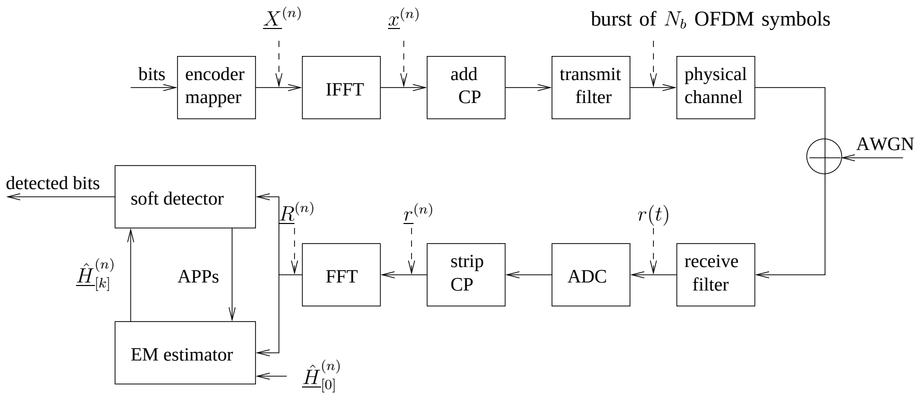

Fig. 1. coded OFDM transmitter and receiver

resulting in the vector

$$\underline{x} = \mathbf{F}\underline{X} \tag{1}$$

where $\mathbf{F}$ is an $N \times N$ matrix with $F_{m,n} = 1/\sqrt{N}e^{j2\pi nm/N}$. A cyclic prefix (CP) of length N_g is added, such that

$$x_k = x_{k+N},\ k = -N_g, \ldots, -1. \tag{2}$$

The resulting $N + N_g$ symbols are pulse-shaped with a square root Nyquist pulse with signaling interval T_s. Hence, the OFDM symbol period equals $T = (N + N_g)\, T_s$. We consider bursty transmission whereby a sequence of N_b OFDM symbols is transmitted consecutively. We denote the k-th time-domain symbol of the n-th OFDM symbol as $x_k^{(n)}$, for $n = 0, \ldots, N_b - 1$ and $k = -N_g, \ldots, N - 1$. The resulting signal passes through a time- and frequency-selective channel. We assume the channel variations are negligible within one OFDM symbol.

The signal is further corrupted with thermal white noise. After matched filtering we obtain the following equivalent model:

$$r\,(t) = \sum_{n=0}^{N_b-1} \sum_{k=-N_g}^{N-1} x_k^{(n)} h^{(n)}\,(t - nT - kT_s) + w\,(t) \tag{3}$$

where $h^{(n)}\,(t)$ represents the overall impulse response (including transmit filter, physical channel and receive filter) during the n-th OFDM symbol within the burst and $w\,(t)$ is a noise term. To avoid interference between successive OFDM symbols, the transmission system is designed such that $N_g T_s$ exceeds the delay spread of the overall channel, say LT_s. Sampling at a rate $1/T_s$ yields, for $l \in [0, \ldots, N-1]$:

$$r_l^{(n)} \doteq r\,(nT + lT_s) = \sum_{m=0}^{L-1} h_m^{(n)} x_{l-m}^{(n)} + w_l^{(n)} \tag{4}$$

where we have introduced $h_m^{(n)} = h^{(n)}\,(mT_s)$ and $w_l^{(n)} = w\,(nT + lT_s)$. Note that we do not take into account samples within the cyclic prefix. By designing the receive filter to be a unit-energy Nyquist filter, the noise component $w_l^{(n)}$ in (4) is white with variance σ^2 per real dimension. We stack $\underline{r}^{(n)} = \left[r_0^{(n)}, \ldots, r_{N-1}^{(n)}\right]^T$, $\underline{h}^{(n)} = \left[h_0^{(n)}, \ldots, h_{L-1}^{(n)}\right]^T$ and $\tilde{\underline{h}}^{(n)} = \left[h_0^{(n)}, \ldots, h_{L-1}^{(n)}, \underline{0}_{N-L}^T\right]^T$. We may write

$$\begin{aligned} \underline{r}^{(n)} &= circ\left(\tilde{\underline{h}}^{(n)}\right) \underline{x}^{(n)} + \underline{w}^{(n)} && (5)\\ &= circ\left(\tilde{\underline{h}}^{(n)}\right) \mathbf{F}\underline{X}^{(n)} + \underline{w}^{(n)} && (6) \end{aligned}$$

where $\underline{w}^{(n)} = \left[w_0^{(n)}, \ldots, w_{N-1}^{(n)}\right]^T$. Applying an FFT gives us $\mathbf{F}^H \underline{r}^{(n)} \doteq \underline{R}^{(n)}$

$$\begin{aligned} \mathbf{F}^H \underline{r}^{(n)} &= \mathbf{F}^H circ\left(\tilde{\underline{h}}^{(n)}\right) \mathbf{F}\underline{X}^{(n)} + \mathbf{F}^H \underline{w}^{(n)} && (7)\\ &= diag\left(\underline{H}^{(n)}\right) \underline{X}^{(n)} + \underline{W}^{(n)} && (8)\\ &= diag\left(\underline{X}^{(n)}\right) \underline{H}^{(n)} + \underline{W}^{(n)} && (9) \end{aligned}$$

where $\underline{H}^{(n)} = \mathbf{F}^H \tilde{\underline{h}}^{(n)}$ represents the sampled channel frequency response (CFR) during the n-th OFDM symbol in the burst.

2.1 Data detection

As far as our contribution is concerned, data detection is based on the knowledge of both $\underline{R}^{(n)}$ and $\underline{H}^{(n)}$ (coherent detection). We consider a 'soft' detector that computes a posteriori probabilities of the coded symbols (as is done in many state-of-the-art error-correcting coding schemes). This may be achieved as follows: in order to perform soft demapping and decoding (jointly: soft detection) of the n-th OFDM symbol, the decision variables $\underline{R}^{(n)}$ need to be converted to probabilities. We denote these probabilities by $\left\{p^{(1)}\left(X_k^{(n)}\right)\right\}$, $k = 0, \ldots, N-1$, with

$$p^{(1)}\left(X_k^{(n)} = \omega\right) = C \exp\left(-\frac{1}{2\sigma^2}\left|R_k^{(n)} - H_k^{(n)}\omega\right|^2\right) \tag{10}$$

where ω is a generic element in the signaling constellation Ω and C is a normalizing constant. These probabilities are then used in the detector, in order to compute a posteriori probabilities (APPs) of the information bits, coded bits and coded (frequency-domain) symbols. This may be achieved through the BCJR algorithm [13], or the more general sum-product algorithm [14], or some other technique. In any case, the APPs are given by

$$p\left(X_k^{(n)} \middle| \underline{R}^{(n)}, \underline{H}^{(n)}\right) = \gamma p^{(1)}\left(X_k^{(n)}\right) p^{(2)}\left(X_k^{(n)}\right) \tag{11}$$

for some normalizing constant γ. In turbo-processing parlance, $p^{(2)}\left(X_k^{(n)}\right)$ is the *extrinsic* probability of symbol $X_k^{(n)}$, as computed within the soft detector. It is clear that the detector requires knowledge of the CFR in order to evaluate (10).

2.2 Operating assumptions

We assume the receiver is memory- and delay-restricted in a sense that it cannot postpone data detection until the entire burst has been received: the data should be recovered in real time, one OFDM symbol at a time. This implies that the receiver should be able to recover $\underline{X}^{(n)}$, based solely on $\underline{r}^{(n)}$. We assume perfect timing- and carrier frequency synchronization. Furthermore, we have the following a priori information available regarding the channel:

- the delay spread of the overall channel does not exceed LT_s;
- the channel may vary slowly from OFDM symbol to OFDM symbol.

No further assumptions w.r.t. Doppler shifts, autocorrelation properties or any underlying channel model are made. It is our goal to develop a channel estimator that is able to operate under these very stringent constraints.

3 Code-aided channel tracking

We need to estimate $\underline{H}^{(n)}$, based solely on the observation $\underline{r}^{(n)}$, without any a priori knowledge regarding the channel statistics (as detailed above). Since Maximum Likelihood (ML) estimation is generally intractable (due to the presence of the coded symbols), we resort to the EM algorithm to find the ML estimate.

3.1 EM estimation - principle

The EM algorithm is an *iterative* technique to find the ML estimate of a parameter θ from an observation r [6]. It is based on the concept of so-called missing (or unobserved) data y, such that, if the missing data were known, estimating θ would be easy. We will denote the iteration index by a subscript k. Starting from an initial estimate $\hat{\theta}_{[0]}$, we iteratively apply the following two steps:

1) E-step:

$$Q\left(\theta \middle| \hat{\theta}_{[k]}\right) = \int \log p\left(r \mid y, \theta\right) p\left(y \mid r, \hat{\theta}_{[k]}\right) dy \tag{12}$$

2) M-step:

$$\hat{\theta}_{[k+1]} = \arg\max_{\theta} Q\left(\theta \middle| \hat{\theta}_{[k]}\right). \tag{13}$$

The EM algorithm terminates once the estimate has converged or once a certain stopping criterion has been met. We denote the final estimate by $\hat{\theta}_{[+\infty]}$.

3.1.1 Direct CFR estimation

We can now apply the EM algorithm to estimate the CFR directly by setting: $\underline{H}^{(n)} \to \theta$, $\underline{R}^{(n)} \to r$, and $\underline{X}^{(n)} \to y$. We abbreviate $diag\left(\underline{X}^{(n)}\right)$ by $\mathbf{X}$. Since

$$\log p\left(\underline{R}^{(n)} \mid \underline{X}^{(n)}, \underline{H}^{(n)}\right) \propto -\left\|\underline{R}^{(n)} - \mathbf{X}\underline{H}^{(n)}\right\|^2, \tag{14}$$

we easily find a closed-form solution for the M-step:

$$\underline{\hat{H}}_{[k+1]}^{(n)} = \left(\widetilde{\mathbf{X}^H\mathbf{X}}\right)^{-1} \widetilde{\mathbf{X}}^H \underline{R}^{(n)} \tag{15}$$

where

$$\widetilde{\mathbf{X}} = \int \mathbf{X}\, p\left(\underline{X}^{(n)} \mid \underline{R}^{(n)}, \underline{\hat{H}}_{[k]}^{(n)}\right) d\underline{X}^{(n)} \tag{16}$$

and

$$\widetilde{\mathbf{X}^H\mathbf{X}} = \int \mathbf{X}^H\mathbf{X}\, p\left(\underline{X}^{(n)} \mid \underline{R}^{(n)}, \underline{\hat{H}}_{[k]}^{(n)}\right) d\underline{X}^{(n)}. \tag{17}$$

Observe that the quantities $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{X}^H\mathbf{X}}$ depend on the *current* estimate of $\underline{H}^{(n)}$: $\underline{\hat{H}}_{[k]}^{(n)}$. Most importantly, we see that each entry in the matrices $\mathbf{X}$ and $\mathbf{X}^H\mathbf{X}$ depends only on a single frequency-domain symbol (say, $X_l^{(n)}$). This implies that $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{X}^H\mathbf{X}}$ can be computed based on the *marginal APPs* $\left\{p\left(X_l^{(n)} \mid \underline{R}^{(n)}, \underline{\hat{H}}_{[k]}^{(n)}\right)\right\}$, rather than the joint APP $p\left(\underline{X}^{(n)} \mid \underline{R}^{(n)}, \underline{\hat{H}}_{[k]}^{(n)}\right)$. Fortunately, these marginal APPs are exactly the quantities provided by the detector as described in section 2.1. Hence, we have obtained a code-aided CFR estimation algorithm that can be implemented in a practical system.

3.1.2 Indirect CFR estimation

Although direct estimation of the CFR is conceptually quite simple, is has one significant drawback: it does not take into account the fact that we know the maximum delay spread of the channel. We therefore describe an alternative formulation where we take this knowledge into account explicitly. Let us first see how $\underline{H}^{(n)}$ and $\underline{h}^{(n)}$ are related. Since, $\underline{H}^{(n)} = \mathbf{F}^H \underline{\tilde{h}}^{(n)}$, we know that

$$\underline{H}^{(n)} = \mathbf{W}^H \underline{h}^{(n)} \tag{18}$$

where $\mathbf{W}^H$ is an $N \times L$ matrix, consisting of the first L columns of $\mathbf{F}^H$, with $\mathbf{W}\mathbf{W}^H = \mathbf{I}_L$.

Now, we apply the EM algorithm with $\underline{h}^{(n)} \to \theta$, $\underline{R}^{(n)} \to r$, and $\underline{X}^{(n)} \to y$. After some straightforward manipulations, we easily find the following closed-form solution for the M-step:

$$\underline{\hat{h}}_{[k+1]}^{(n)} = \left(\mathbf{W}\widetilde{\mathbf{X}^H\mathbf{X}}\mathbf{W}^H\right)^{-1} \mathbf{W}\widetilde{\mathbf{X}}^H \underline{R}^{(n)} \tag{19}$$

where $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{X}^H\mathbf{X}}$ are again given by (16)-(17). Considering (18), we obtain the following CFR estimate

$$\underline{\hat{H}}_{[k+1]}^{(n)} = \mathbf{W}^H \left(\mathbf{W}\widetilde{\mathbf{X}^H\mathbf{X}}\mathbf{W}^H\right)^{-1} \mathbf{W}\widetilde{\mathbf{X}}^H \underline{R}^{(n)}. \tag{20}$$

Hence, we have obtained a code-aided CFR estimation algorithm that (a) takes into account the delay spread of the channel and (b) can be implemented in a practical system.

3.2 Observations

General observations

The two CFR estimation algorithms described above accept information from the decoder in the form of a posteriori probabilities. When there is no error-correcting code present, these algorithms revert into those from [15]. In case pilot symbols are present within the OFDM symbol, they are included in the EM algorithm most elegantly by means of the APPs (which in this case will be Dirac distributions). Futhermore, these algorithms were derived in a systematic (rather than ad-hoc) way based on the well-known EM algorithm. Hence, they assure convergence to a local optimum of the likelihood function. This optimum strongly depends on the initial estimate $\underline{\hat{H}}^{(n)}_{[0]}$. The issue of finding an initial estimated will be tackled later in this paper.

Initialization

The EM algorithm forces us to define an initial estimate of the CFR for each OFDM symbol: $\underline{\hat{H}}^{(n)}_{[0]}$ for $n = 0, \ldots, N_b - 1$. Such an estimate can be provided by some conventional blind or DA estimation technique. Some authors propose to use multiple random initial estimates, and then select the *best* one, according to some ad-hoc criterion. Here, we take a different approach: since we know the channel varies *slowly* from one OFDM symbol to the next, we set $\underline{\hat{H}}^{(n)}_{[0]} = \underline{\hat{H}}^{(n-1)}_{[\infty]}$ for $n \geq 1$, where $\underline{\hat{H}}^{(n-1)}_{[\infty]}$ represents the *final* estimate of the CFR during the *previous* OFDM symbol. Hence, no training symbols are needed after the first OFDM symbol, nor do we need to resort to unreliable blind or stochastic techniques. This approach allows us to decode OFDM symbols on-line in a sequential manner. No a priori information regarding the channel statistics is required, making this estimator quite robust. Should a priori information be available (e.g., correlation between channel taps), the algorithm can be modified to take this into account, as described in [16].

Several authors (including [10], [11]) have taken a detection-centric point of view in implementing the EM algorithm, by interchanging the roles of the CFR and the data symbols. This approach has several drawbacks, as first and second-order a posteriori information of the CFR or CIR needs to be available. This in turn implies the availability of an underlying channel model. Our approach does not require any such model. Secondly, the detection-centric approach requires making a *hard*, rather than a *soft* decision w.r.t. the transmitted codeword at each EM iteration.

Complexity considerations

For constant-modulus constellations, the matrix $\widetilde{\mathbf{X}^H\mathbf{X}} = \mathbf{I}_N$, which greatly simplifies both estimation algorithms. For instance, the indirect CFR estimation algorithm now gives rise to the following M-step:

$$\underline{\hat{H}}^{(n)}_{[k]} = \mathbf{W}^H\mathbf{W}\widetilde{\mathbf{X}}^H\underline{R}^{(n)} \tag{21}$$

where $\mathbf{W}^H\mathbf{W}$ can be precomputed at the receiver. Observe that $\mathbf{W}^H\mathbf{W}$ can be interpreted as a low-pass filter, filtering out channel components of the direct CFR estimate beyond the delay spread.

The EM algorithm is iterative. Since the channel decoder is also iterative, EM estimation becomes hugely complex: each time the CFR estimate is updated, APPs have to recomputed, requiring many iterations within the decoder. Hence, complexity will scale as the number of decoder iterations times the number of EM iterations. As this may be prohibitive, we employ the concept of *embedded* estimation [17]: each time the CFR estimate is updated, only a single iteration within the decoder is performed. Furthermore, the decoder maintains state information from one EM iteration to the next. This allows for a huge saving in computational complexity, at the cost of some sub-optimality in the EM algorithm (since the APPs will be less accurate).

The computational overhead may be further reduced by including the following two stopping criteria:

- Once the CFR estimate has converged, no further EM iterations are required, and all available processing power can be devoted to decoding. A possible stopping criterion is given by

$$\frac{\left\| \underline{\hat{H}}^{(n)}_{[k]} - \underline{\hat{H}}^{(n)}_{[k-1]} \right\|}{\left\| \underline{\hat{H}}^{(n)}_{[k-1]} \right\|} < \varepsilon$$

  for some small value of ε.
- Once the decoder has successfully recovered a codeword[2] corresponding to the n-th OFDM symbol, we can stop the decoding iterations as well as the EM iterations, and set:

$$\underline{\hat{H}}^{(n+1)}_{[0]} = \mathbf{W}^H \left(\mathbf{W}\mathbf{X}_h^H\mathbf{X}_h\mathbf{W}^H\right)^{-1}\mathbf{W}\mathbf{X}_h\underline{R}^{(n)}$$

  where $\mathbf{X}_h$ corresponds to taking *hard* decisions w.r.t. coded symbols. Since the probability of false detection is generally negligible by design of the code, this will improve the initial estimate for the *next* OFDM symbol, as well as reduce the processing power required for the *current* symbol.

4 Numerical results

4.1 Simulation set-up

We have carried out computer simulations to evaluate the performance of the two proposed EM-based channel tracking techniques. A turbo-coded system was considered where the constituent convolutional codes are systematic and recursive with octal generators $(21, 37)_8$ and constraint length $\nu = 5$. The constituent encoders are separated by a pseudo-random interleaver, which varies from codeword to codeword. Each codeword corresponds to 170 information bits. The overall rate of the turbo code is $1/3$ resulting in

[2]Assuming the decoder checks at each iteration if the detected word satisfies all code constraints, as is commonly done for LDPC codes.

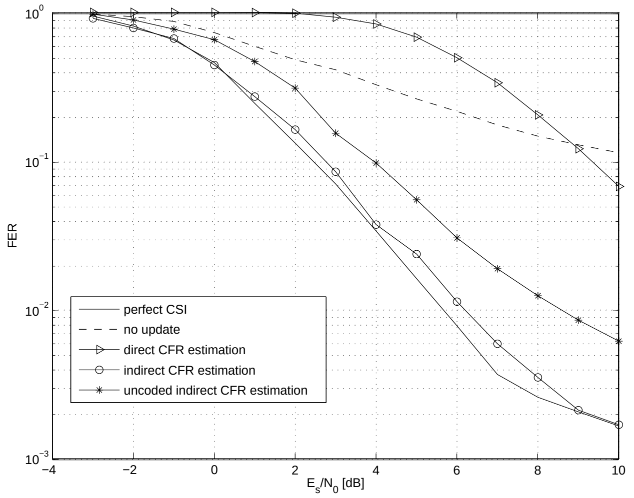

Fig. 2. Frame Error Rate (FER) vs. SNR for various channel tracking algorithms

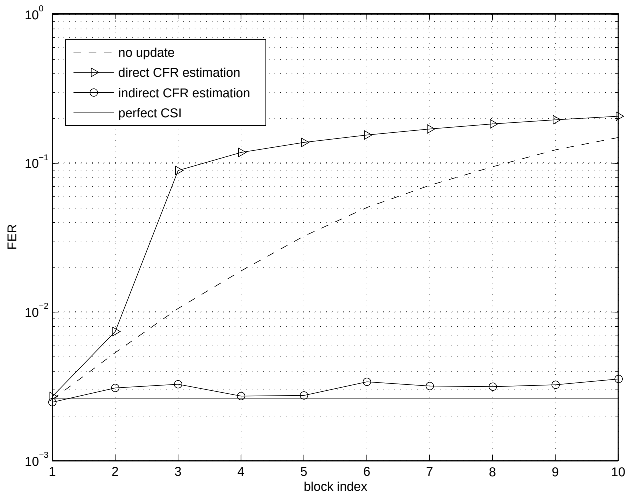

Fig. 3. Frame Error Rate (FER) vs. block index for various channel tracking algorithms

512 bits (two stuffing zero-bits were added). These bits are once more interleaved to fully exploit the frequency diversity in the channel and mapped onto a QPSK constellation. The resulting 256 QPSK symbols each modulate one of the $N = 256$ available subcarriers. The length of the CIR was set to $L = 6$, and the guard interval to $N_g = 6$. Bursts consist of $N_b = 10$ OFDM symbols.

The channel was generated according to the following model: for each burst, the taps $\left\{h_k^{(0)}\right\}$, $k = 0, \ldots, L-1$ are zero mean independent complex Gaussian random variables with variance σ_k^2. The channel delay profile is exponential with $\sigma_k^2 = \phi \exp\left(-k/\kappa\right)$, where ϕ is selected such that the expected energy per subcarrier is equal to 1 (i.e., $1/2$ per real dimension). We have set $\kappa = 5$. The time-selective nature of the channel is modeled through the following first-order Markov model, for $n > 0$:

$$h_k^{(n)} = \alpha h_k^{(n-1)} + \sqrt{1-\alpha^2}\sigma_k w_{k,n} \qquad (22)$$

where $\{w_{k,n}\}$ is a sequence of zero-mean AWGN samples with $E\left[|w_{k,n}|^2\right] = 1$. The parameter $\alpha \in [0,1]$ determines the time-correlation function of the fading process. For instance, for Jakes' model, we have

$$\alpha = J_0\left(2\pi f_d T\right) \qquad (23)$$

where $f_d T$ is the normalized Doppler shift and $J_0\left(.\right)$ is the zero-order Bessel function of the first kind. It is important to note that the model (22), used in our simulations, is not exploited in the estimation algorithms. This is in contrast to many papers in technical literature, where a model such as (22) is used explicitly in the derivation of an estimation algorithm. Of course, the resulting algorithms result in a non-robust estimator when the model is not valid.

To remove any dependency on a specific initial channel estimate, we assume the first OFDM symbol has perfect knowledge of the channel so that we set $\underline{\hat{H}}_{[k]}^{(0)} = \underline{H}^{(0)}$, for all k. No pilot or training symbols are used for channel tracking.

4.2 Performance results

Let us first consider the frame error rate (FER) (or codeword error rate) of our systems for a fixed value of $\alpha = 0.985$ (corresponding to $f_d T \approx 0.04$) as a function of the SNR ($E_s/N_0 = 2\sigma^2$) under the following circumstances (see Fig. 2):

- a receiver with perfect channel state information (marked 'perfect CSI')
- a receiver that takes the initial estimate of the CFR and uses this for the remainder of the burst (marked 'no update')
- a receiver that performs direct code-aided CFR estimation (marked 'direct CFR estimation')
- a receiver that performs indirect code-aided CFR estimation (marked 'indirect CFR estimation')
- a receiver that performs indirect CFR estimation, but without exploiting the code. After 10 EM iterations, the final estimate of the CFR is provided to the detector, which performs iterative detection (marked 'uncoded indirect CFR estimation')

It is clear that if we disregard the time-varying nature of the channel, large degradations ensue. Performing direct code-aided CFR estimation can somewhat reduce these degradations at large SNRs. For low SNR, direct CFR estimation actually increases the degradation: the high amount of degrees of freedom cause the estimates to take on random values. As is apparent, indirect code-aided CFR estimation is able to reduce the degradations to roughly 0.5 dB. We also see that when we do not exploit code properties during channel tracking, the remaining degradation is quite large, up to almost 4 dB.

Let us now take a more detailed view, and consider on a fixed SNR, say $E_s/N_0 = 8$ dB. In Fig. 3, we show the FER as a function of the block index (i.e., the index of the

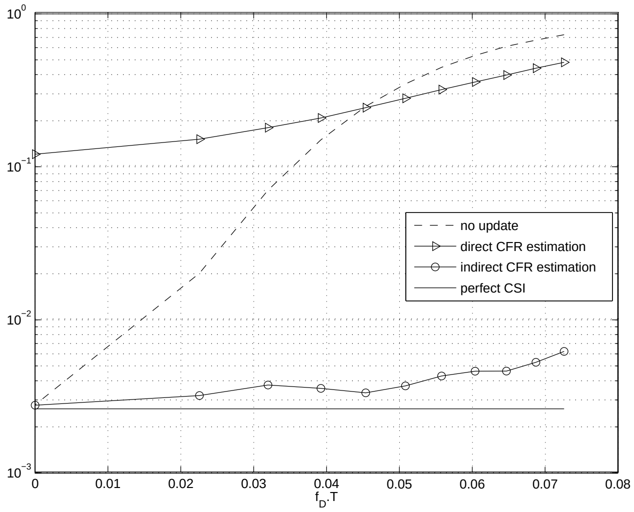

Fig. 4. Frame Error Rate (FER) vs. Doppler shift for various channel tracking algorithms

codeword within the burst). Initially, at block index 1, all algorithms use the same CFR value, so they have identical FER performance. When the block index increases, we see a steady increase in FER when we do no update the CFR estimate. Most interestingly, direct CFR estimation leads to large degradations even for low values of the block index. On the other hand, indirect CFR estimation does not give rise to a significant increase of the FER, even for larger values of the block index.

Finally, let us see how our algorithm behaves when we vary the parameter α between $\alpha = 0.95$ and $\alpha = 1.0$ (this latter case corresponds the a channel that is static during the entire burst). In Fig. 4, we show the FER performance for block index 10, with $E_s/N_0 = 8$ dB as a function of the Doppler shift (related to α through (23)). All algorithms lead to an increase in the FER as the Doppler shift increases, but the proposed indirect CFR estimation algorithm clearly has the best performance for all considered Doppler shifts.

5 Conclusions and remarks

We have presented a code-aided channel tracking algorithm for coded OFDM systems. The algorithm is based on the EM algorithm and iterates between data detection and estimation. No a priori information regarding the channel statistics is required, other than the delay spread (which is generally known, by design of the OFDM system). By associating one codeword to each OFDM symbol, channel tracking can be performed on-line. Provided an accurate initial estimate of the channel is available, code-aided channel tracking can be performed without resorting to pilot symbols, as shown by our computer simulations. Exactly how reliable the initial estimate should be, remains a topic for further research.

The algorithm can be applied without modification to a bit-loading scenario. Extensions to multi-antenna systems are conceptually straightforward, although the SAGE algorithm may have to be applied for reasons of computational complexity. Finally, the cyclic prefix can also be exploited in these estimation algorithms, as mentioned in [8], using a slightly modified observation model.

References

[1] J.A.C. Bingham. ”Multicarrier modulation for data transmission: an idea whose time has come”. *IEEE Comm. Mag.*, 28(5):5–14, May 1990.

[2] Necker, M.C.; Stuber, G.L.;. ”Totally blind channel estimation for OFDM on fast varying mobile radio channels”. *IEEE Trans. on Wireless Comm.*, 3(5):1514–1525, September 2004.

[3] Muquet, B.; de Courville, M.; Duhamel, P.;. ”Subspace-based blind and semi-blind channel estimation for OFDM systems”. *IEEE Trans. on Signal Processing*, 50(7):1699–1712, July 2002.

[4] M. Morelli and U. Mengali. ”A comparison of pilot-aided channel estimation methods for OFDM systems”. *IEEE Trans. on Signal Processing*, 49(12):3065–3073, December 2001.

[5] Barhumi, I.; Leus, G.; Moonen, M.;. ”Optimal training design for MIMO OFDM systems in mobile wireless channels”. *IEEE Trans. on Signal Processing*, 51(6):1615–1624, June 2003.

[6] A.P. Dempster, N.M. Laird and D.B. Rubin. ”Maximum likelihood from incomplete data via the EM algorithm”. *Journal of the Royal Statistical Society*, 39(1):pp. 1–38, 1977. Series B.

[7] V.Mignone and A.Morello. ”CD3-OFDM: a novel demodulation scheme for fixed and mobile receivers”. *IEEE Trans. on Communications*, 4(9):1144–1151, September 1996.

[8] Al-Rawi, G.A.; Al-Naffouri, T.Y.; Bahai, A.; Cioffi, J.;. ”Exploiting error-control coding and cyclic-prefix in channel estimation for coded OFDM systems”. *IEEE Comm. Letters*, 7(8):388–390, August 2003.

[9] Seung Young Park; Yeun Gu Kim; Chung Gu Kang. ”Iterative receiver for joint detection and channel estimation in OFDM systems under mobile radio channels”. *IEEE Trans. on Vehicular Technology*, 53(2):450–460, March 2004.

[10] Lu, B.; Xiaodong Wang; Ye Li;. ”Iterative receivers for space-time block-coded OFDM systems in dispersive fading channels”. *IEEE Trans. on Wireless Comm.*, 1(2):213–225, April 2002.

[11] Y. Xie; Georghiades, C.N.;. ”Two EM-type channel estimation algorithms for OFDM with transmitter diversity”. *IEEE Trans. on Comm.*, 51(1):106–115, January 2003.

[12] N. Noels, V. Lottici, A. Dejonghe, H. Steendam, M. Moeneclaey, M. Luise , L. Vandendorpe. ”A Theoretical Framework for Soft Information Based Synchronization in Iterative (Turbo) Receivers”. *EURASIP Journal on Wireless Communications and Networking JWCN, Special issue on Advanced Signal Processing Algorithms for Wireless Communications*, 2005(2):117–129, April 2005.

[13] L.R. Bahl, J. Cocke, F. Jelinek and J. Raviv. ”Optimal decoding of linear codes for minimising symbol error rate”. *IEEE Trans. Inform. Theory*, 20:pp. 284–287, March 1974.

[14] F. Kschischang, B. Frey and H.-A. Loeliger. ”Factor graphs and the sum-product algorithm”. *IEEE Trans. Inform. Theory*, 47(2):pp.498–519, February 2001.

[15] X. Ma, H. Kobayashi and S.C. Schwartz. ”EM-based channel estimation algorithms for OFDM”. *EURASIP Journal on Applied Signal Processing*, 10:1460–1477, 2004.

[16] Gallo, A.S.; Vitetta, G.M.; Chiavaccini, E. ”A BEM-based algorithm for soft-in soft-output detection of co-channel signals”. *IEEE Transactions on Wireless Communications*, 3(5):1533–1542, September 2004.

[17] V. Lottici and M. Luise. ”Embedding carrier phase recovery into iterative decoding of turbo-coded linear modulations”. *IEEE Transactions on Communications*, 52(4):661–669, April 2004.