Iterative EM Based IMD Synchronization for Fast Time-Variant Channel with Subspace Order Recursive LS Iterator

Jan Sykora, Jan Vcelak
Czech Technical University in Prague, Czech Republic
{Jan.Sykora,vcelakj}@fel.cvut.cz

Abstract—We solve the problem of iterative expectation-maximization (EM) based information measure directed (IMD) synchronization in fast time-variant environment. We identified correspondence between the EM iterator and the weighted least squares (LS) problem. An order recursive form of the LS iterator reduces the implementation complexity. The channel coefficients are allowed to change arbitrarily up to the granularity of the symbol period. The coefficients are modeled using parametric subspace providing a wide range of configuration for the channel dynamics while reducing the dimensionality of the problem substantially.

I. INTRODUCTION

A. Background

The iterative Information Measure Directed (IMD) synchronization (channel state estimation) has been receiving a considerable attention in recent years. It is a natural companion for the iterative decoding (see e.g. [1]). It uses the information measure (IM) from the iterative decoding network (turbo decoder) to assist the channel state estimator. This is essential since the modern turbo codes provide excellent performance at extremely low signal-to-noise ratios where traditional synchronization schemes fail to provide reliable estimates. The iterative synchronization utilizes all available IM at the receiver and provides superior results to the traditional methods.

A solution of the iterative decoding in the presence of parametric channel is generally approached by two ways. The first represented by [2], [3] modifies the first soft-in soft-out (SISO) block in the iterative decoding network to inherently provide information measures marginalized over the channel parameters. The marginalization procedure depends on the type of the parameter (random, deterministic). Those approaches increases substantially the implementation complexity of the forward-backward algorithm in SISO block.

Another approach builds on cooperative decoding and synchronization iteration loops providing one to each other the information measure estimates of the current iteration. The synchronizer becomes information measure directed (IMD). There are several flavors of the algorithms that can be used in synchronization IMD loop. It could be either soft-decision directed (SDD) one [4] or expectation-maximization (EM) based one [5]. The general framework for IMD synchronization is built in [6].

B. Problem statement

So far, the IMD iterative synchronization is considered by researchers only for time-invariant of quasi-invariant scenarios. The channel coefficients are typically assumed constant over the whole frame typically corresponding to the length of the turbo code interleaver. Some authors go somewhat further by allowing channel changes in between the blocks (e.g. [7]) however still keeping the channel constant over the block. To the best of our knowledge, there are no results for fast time-variant channel that can change as fast as per individual channel symbols. This significantly reduces a usability of the iterative synchronization in fast time-variant environment.

In the fast time-variant environment, the trace of the parameter (the number of observation signal components affected by particular parameter component) is very short and therefore the estimator performance is critical (short observation). The resulting MSE of the estimator and the error rate of the decoder are substantially influenced by the optimality of the estimator. This is on contrary to the case of a block-constant channel parameter where the trace is so long (typically hundreds of symbols) that the actual estimator optimality is not very important and the estimator provides good performance anyway.

C. Contribution of the paper

The paper develops the iterative IMD synchronization for fast time-variant channel without any restriction on rate of the changes (like block-like invariance). The channel is allowed to change arbitrarily (according to the subspace model) up to the channel symbol period granularity. We revealed the correspondence between the EM based IMD iterator and weighted least squares (LS) problem. The channel coefficients are modeled using parametric subspace that allows easy and general configuration of the algorithm for a wide range of channel dynamics. The subspace weighted LS iterator can be implemented using the order recursive algorithm which allows reduction of the implementation complexity.

This work was supported by the Grant Agency of the Czech Republic, grant 102/05/2177 and by the Ministry of education, youth and sports of the Czech Republic, grant OC273 and prog. MSM6840770014. APCC 2005, R1.2.0 (14.7.2005).
II. SYSTEM MODEL

We consider a system with general encoding network (serial, parallel concatenation) consisting of arbitrarily interconnected component encoders, interleavers. The last stage of this encoding network forms an output of the system. The signal space representation of the modulated signal is \( s(q) \) where \( q = [q_1, q_2, \ldots, q_n]^T \) is the vector of channel symbols \( q_n \in \{q_i^{(s)}\}_{i=1}^{M_s} \) \((M_s\) is the size of the channel symbols alphabet). In the case of linear (one-dimensional) modulation, it is \( s(q) = [\ldots, s(q_n), \ldots]^T \) where \( s(\cdot) \) is the signal space constellation mapping function.

Received signal in a signal space representation for \( n \)-th symbol in the sequence is given by

\[
x_n = h_n s(q_n) + w_n
\]

where \( h_n \) are complex valued time-variant generally correlated channel transfer coefficients. Quantities \( w_n \) are circularly symmetric complex IID Gaussian noise variables with variance \( \sigma_w^2 \).

We also form a vectors \( x = [\ldots, x_n, \ldots]^T \) (and similarly for \( h \) and \( w \)).

The channel coefficients are parametrized by the vector parameter \( \theta \) of generally lower dimension with respect to the dimension of \( h \). Channel parameters \( \theta = [\ldots, \theta_1, \ldots]^T \) are assumed to be IID random variables. The parametrization \( \{h(\theta)\}_\theta \) defines a subspace in the channel transfer coefficients space. The channel model in vector notation is \( x = h(\theta)^T s(q) + w \). The subspace parametrization allows easy modeling of the channel changes dynamics. Various models can be adopted (see Sec. IV-C), e.g. autoregressive (AR) model, moving average (MA) model, block fading with independent observation windows, etc.

The channel is said to have independent eliminated channel states (all ergodic parameters, e.g. AWGN) if \( p(w) = \prod_n p(w_n) \). Additionally, the channel is said to be a memoryless channel if a single component of received signal depends only on a single channel symbol and single component of eliminated channel state vector \( p(x_n|q_\theta, w) = p(x_n|q_n, \theta, w_n) \). These assumptions directly imply \( p(x_n|q_\theta) = \prod_n p(x_n|q_n, \theta) \). In our particular case, it is

\[
p(x_n|q_n, \theta) \equiv \exp\left(-\frac{1}{\sigma_w^2} |x_n - h_n(\theta)s(q_n)|^2 \right).
\]

III. ITERATIVE EM BASED INFORMATION MEASURE DIRECTED SYNCHRONIZATION

Iterative EM (Expectation-Maximization) based IMD (Information Measure Directed) synchronization is one of the methods particularly suited for the synchronization in the receiver with iterative (turbo) decoder (for details, see [6], [8], [5]). Fig. 1 shows the principle of cooperating iteration loops of the iterative decoding network and IMD synchronizer. The current iteration index for decoding loop is generally multidimensional (the network can contain multiple sub-loops) \( \mathbf{m} \) and the iteration index for the synchronizer is \( k \).

The decoding loop takes the information measures (IM) from the soft-output demodulator (SODEM) for the current channel parameter estimate \( \hat{h}^{k,\mathbf{m}} \) based on the subspace parameter \( \hat{\theta}^{k,\mathbf{m}} \). The decoding network is fed by forward information measures

\[
\mathcal{M}_F^{k,\mathbf{m}} \{ q_n \} = \{ p(x^{q_i^{(s)}}|q_n, \hat{\theta}^{k,\mathbf{m}}) \}_{n,i} = \{ p^{k,\mathbf{m}}(x|q_i^{(s)}) \}_{n,i}
\]

and provides the backward information measures \( \mathcal{M}_B^{k,\mathbf{m}} \{ q_n \} = \{ p(q_n|q_i^{(s)}) \}_{n,i} \). These cooperative decoding and synchronization loops solve the problem of generally 2-dimensional information measure of the channel output parametrized by channel symbols and channel coefficients by providing 1-dimensional hyperplane cuts (for current \( \hat{h}^{k,\mathbf{m}} \) of 2-dimensional IM (see [6] for details).

The final form of the EM based iterator is [6]

\[
\hat{\theta}^{k+1,\mathbf{m}} = \arg \max_\theta \sum_n \sum_{q_n} \ln p(x_n|q_n, \theta)p^{k,\mathbf{m}}(q_n|x, \theta^{k,\mathbf{m}})
\]

where the symbol a posteriori PDF estimation is given by

\[
p^{k,\mathbf{m}}(q_n|x, \theta^{k,\mathbf{m}}) = \frac{p(x|q_n, \theta^{k,\mathbf{m}})p^{k,\mathbf{m}}(q_n)}{\sum_{q_n} p(x|q_n, \theta^{k,\mathbf{m}})p^{k,\mathbf{m}}(q_n)}.
\]

The right-hand side depends on \( k \) even for fixed \( \mathbf{m} \) and the iteration loops can run independently. The possibility of iterating over \( k \) and improving the estimation can save the number of necessary runs of computationally expensive decoding iteration (Forward-Backward Algorithm).

IV. PARAMETER SUBSPACE ORDER RECURSIVE LS ESTIMATOR

A. LS interpretation of EM algorithm

In the additive Gaussian noise channel, the channel log-likelihood function becomes

\[
\ln p(x_n|q_n = q_i, \theta) = - |x_n - h_n(\theta)s(q_i)|^2.
\]

The a posteriori PDF estimation \( \hat{p}^{k,\mathbf{m}}(q_n = q_i|x, \theta^{k,\mathbf{m}}) \) is constant from the perspective of \( \theta \) and depends on the current iteration indices \( k, \mathbf{m} \) (which are fixed for given EM run), symbol sequence number \( n \) and the channel symbol index \( i \in \{1, \ldots, M_s\} \) through the value \( q_n = q_i \). From the point

---

Fig. 1. Iterative IMD synchronizer with subspace parametrization in the iterative decoding network.
of EM estimator view, the value \( \hat{p}^{k,m}(q_n = q^{(i)}|x, \hat{\theta}^{k,m}) \) can be considered as a weighting constant \( \beta_{n,i} = \hat{p}^{k,m}(q_n = q^{(i)}|x, \hat{\theta}^{k,m}) \). Reformulation of the EM iterator (4) gives
\[
\hat{\theta}^{k+1,m} = \arg \min_\theta \sum_n \sum_{i=1}^{M_0} |x_n - h_n(\theta)s(q^{(i)})|^2 \beta_{n,i}.
\] (7)
This can be directly recognized as weighted LS (Least Squares) problem. The only minor notation modification lies in allowing double indexation by \( n \) and \( i \). It is important to notice that the LS problem corresponds to one particular iteration in EM algorithm, i.e. at each EM iteration we repeatedly solve LS problem.

**B. Vector notation**

The EM iterator (7) can be manipulated in a more convenient vector/matrix form. We use stacked vectors and matrices as it follows. Channel transfer coefficients are parametrized by
\[
h = A\theta
\] (8)
where \( h = [h_1, \ldots, h_N]^T \). The value \( N \) is the length of the channel observation window \( n \in \{1, \ldots, N\} \). Parameter \( L \)-dimensional subspace is generated as a linear span of columns of the \((N \times L)\) matrix \( A \) (subspace generation matrix). The IID random excitation of that parameter subspace is the vector \( \theta = [\theta_1, \ldots, \theta_L]^T \). A stacked \((M_qN \times 1)\) vector \( \tilde{h} \) is defined as
\[
\tilde{h} = h \otimes I_{M_q},
\] (9)
where \( I_{M_q} = [1, \ldots, 1]^T \) is \( M_q \)-dimensional all-ones vector and \( \otimes \) denotes a Kronecker product.

The signal constellation is formed into \( M_q \)-dimensional vector \( s = [s(q^{(1)}), \ldots, s(q^{(M_q)})]^T \) and a diagonal \((M_q \times M_q)\) matrix \( S = \text{diag}(s) \). We form a block-diagonal stacked \((M_qN \times M_qN)\) constellation matrix
\[
\tilde{S} = I_N \otimes S
\] (10)
where \( I_N \) is \((N \times N)\) identity matrix.

The observation is also formed into the stacked \((M_qN \times 1)\) vector \( \tilde{x} = x \otimes I_{M_q} \) where \( x = [x_1, \ldots, x_N]^T \). The weighting coefficients are formed into a diagonal \((M_qN \times M_qN)\) matrix
\[
\tilde{B} = \text{diag}(\beta_{1,1}, \ldots, \beta_{1,M_q}, \beta_{2,1}, \ldots, \beta_{N,M_q}).
\] (11)
Using these definitions, we now reformulate the LS problem (7) into a matrix form
\[
\hat{\theta}^{k+1,m} = \arg \min_\theta (\tilde{x} - \tilde{S}\hat{\theta})^H \tilde{B} (\tilde{x} - \tilde{S}\hat{\theta}).
\] (12)
Vector \( \hat{h} \) is a function of \( \hat{\theta} \). The expression \( \tilde{S}\hat{\theta} \) can be simplified into
\[
\tilde{S}\hat{\theta} = (I_N \otimes S)(h \otimes I_{M_q})
\]
\[
= (I_N \otimes S)((A\hat{\theta}) \otimes (I_{M_q}1))
\]
\[
= (I_N \otimes S)(A \otimes I_{M_q})(\hat{\theta} \otimes 1)
\]
\[
= ((I_NA) \otimes (S1_{M_q}))(\hat{\theta})
\]
\[
= (A \otimes s)\hat{\theta}.
\] (13)
We define a new equivalent signal \((M_qN \times L)\) matrix
\[
\tilde{U} = A \otimes s.
\] (14)
The final form of the EM iterator as the weighted LS problem in matrix notation is
\[
\hat{\theta}^{k+1,m} = \arg \min_\theta (\tilde{x} - \tilde{U}\hat{\theta})^H \tilde{B} (\tilde{x} - \tilde{U}\hat{\theta}).
\] (15)
An alternative form of the iterator incorporating the weighting by rescaling \( \tilde{y} = Q\tilde{x} \) and \( \tilde{V} = Q\tilde{U} \) is
\[
\hat{\theta}^{k+1,m} = \arg \min_\theta (\tilde{y} - \tilde{V}\hat{\theta})^H (\tilde{y} - \tilde{V}\hat{\theta}).
\] (16)
The scaling matrix is get from the decomposition \( \tilde{B} = \tilde{Q}^H \tilde{Q} \) which exists for arbitrary Hermitian matrix \( \tilde{B} \) (in our case it is diagonal). The channel transfer coefficient estimates are expanded as
\[
\hat{h}^{k+1,m} = A\hat{\theta}^{k+1,m}.
\] (17)

**C. Parameter subspace model of the channel coefficients**

The LS framework allows very easy incorporation of various parametrization models for the channel transfer coefficients. Parametrization models are fully defined by the proper choice of the subspace generation matrix \( A \). We can control both the dimension of the subspace \( L \) and the particular basis of the subspace (the columns \( a_k \) of \( A \) that rules the channel subspace which in turn can be used to model e.g. the channel dynamics. It is important to stress that two different subspace generation matrices \( A \) and \( A' \) with columns \( a_k \) and \( a_k' \) respectively lead to the identical solution \( \hat{h}^{k+1,m} \) if they generate the identical subspace
\[
\text{span} \{\{a_k\}_k\} = \text{span} \{\{a_k'\}_k\}.
\] (18)
In the following, we present some of the possible choices for the subspace generation matrix imposing different types of subspace constraints.

1) **Fourier subspace:** This subspace is directly related to the spectral properties of the channel coefficients. Its basis functions are mutually orthogonal complex exponentials
\[
a_k = [e^{j2\pi \frac{n}{N}}, e^{j2\pi \frac{n}{N}}, \ldots, e^{j2\pi \frac{n}{N}}, \ldots, e^{j2\pi \frac{(N-1)n}{N}}]^T.
\] (19)
The subspace has dimension \( L \) (assumed odd) and the generation matrix is
\[
A_N = [a_{-L}, a_0, a_{L-2}, \ldots, a_0, a_{L-2}, \ldots, a_{-L}].
\] (20)
2) **Independent blocks:** The channel transfer coefficients are modeled as mutually independent, non-overlapping blocks of the length \( N_A \). The dimensionality of the subspace is clearly \( L = N/N_A \). Corresponding subspace generation matrix is
\[
A^\text{block}_{(N/N_A)} = I_L \otimes I_{N_A}.
\] (21)
The generation matrix with block structure is particularly useful for the simplification of the order recursive LS implementation since some of the intermediate results can be reused and the computational complexity is lowered.
3) AR model: The first order AR model at the level of individual components of the channel coefficients is given by

\[ h_n = \alpha h_{n-1} + \theta_n. \]  

(22)

The dimensionality of the subspace is \( L = N \). Corresponding subspace generation matrix is

\[ A_{N,1}^{\text{AR1}} = \begin{bmatrix}
1 & 0 & \cdots & 0 & 0 \\
\alpha & 1 & 0 & \cdots & 0 \\
\alpha^2 & \alpha & 1 & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
\alpha^{N-1} & \cdots & \alpha^2 & \alpha & 1
\end{bmatrix}. \]  

(23)

4) Block AR model: A modification of the previous model is the AR first order model applied on the blocks. The dimensionality of the subspace is \( L = N/N_A \). The subspace generation matrix is

\[ A_{N,N_A}^{\text{AR1,Block}} = A_{N/N_A,1}^{\text{AR1}} \otimes I_{N_A}. \]  

(24)

5) MA model: Moving average model is defined by

\[ h_n = \sum_{i=0}^{K} \alpha_i \theta_{n-i}. \]  

(25)

The dimensionality of the subspace is \( L = N \). A corresponding subspace generation matrix is

\[ A_{N,1}^{\text{MA}} = \begin{bmatrix}
\alpha_0 & 0 & \cdots & 0 & 0 \\
\alpha_1 & \alpha_0 & 0 & \cdots & 0 \\
\alpha_2 & \alpha_1 & \alpha_0 & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
\alpha_{N-1} & \cdots & \alpha_2 & \alpha_1 & \alpha_0
\end{bmatrix}. \]  

(26)

6) Block MA model: The block MA model is created similarly to the block AR model. The dimensionality of the subspace is \( L = N/N_A \). The subspace generation matrix is

\[ A_{N,N_A}^{\text{MA,Block}} = A_{N/N_A,1}^{\text{MA}} \otimes I_{N_A}. \]  

(27)

D. Order recursive LS algorithm

Now we turn our attention to solving the weighted LS problem (15). It can be expressed in a closed form [9]

\[ \hat{\theta} = (\tilde{U}^H \tilde{B} \tilde{U})^{-1} \tilde{U}^H \tilde{B} \tilde{x}. \]  

(28)

The expression contains the inversion of the matrix \( \tilde{U}^H \tilde{B} \tilde{U} \) that has dimension \( (L \times L) \). Its practical implementation is feasible only for very low dimensionality of \( L \). However this is not typical in the situation which we target for. In our case, the turbo-coded block can be very long and the dimensionality of the subspace needed to correctly model the fast time-varying channel is relatively high. We also need to keep in our mind that this operation is performed in each EM iteration. The usability of the direct solution in such a case is problematic.

Fortunately, the necessity of inverting large matrices can be avoided by using order recursive LS algorithm [9]. The order recursive algorithm can be applied on rescaled inputs (16). In our case, the weighting matrix has diagonal form which makes the scaling trivial. The order recursive algorithm requires number of recursions given by \( L \). On the other size, it requires matrix multiplications of the order \( N \). The traditional (sequential) recursive algorithm is in exactly opposite situation. The order recursive algorithm has advantage of producing the parameter estimates at arbitrary recursion that respect the full observation. This can be useful for pipelining the operations in the turbo receiver.

The order recursive LS recursion is

\[ D_{\ell+1} = \begin{bmatrix}
D_{\ell} + \frac{\tilde{D}_{\ell}^H \tilde{V}_{\ell+1} \tilde{V}_{\ell+1}^H \tilde{D}_{\ell}}{\tilde{V}_{\ell+1}^H \tilde{V}_{\ell+1} - \tilde{V}_{\ell+1}^H D_{\ell} \tilde{V}_{\ell+1}} - \frac{\tilde{D}_{\ell}^H \tilde{V}_{\ell+1} \tilde{\psi}_{\ell+1} \tilde{\psi}_{\ell+1}^H \tilde{D}_{\ell}}{\tilde{V}_{\ell+1}^H \tilde{V}_{\ell+1} - \tilde{V}_{\ell+1}^H D_{\ell} \tilde{V}_{\ell+1}} \\
\tilde{V}_{\ell+1}^H \tilde{V}_{\ell+1} \end{bmatrix}, \]  

(29)

where \( \tilde{V}_{\ell} \) is a matrix containing the first \( \ell \) columns of \( \tilde{V} \).

A corresponding parameter estimate recursion is

\[ \hat{\theta}_{\ell+1}^{k+1,m} = \hat{\theta}_{\ell}^{k+1,m} + \frac{D_{\ell} \tilde{V}_{\ell+1} \tilde{\psi}_{\ell+1} \tilde{\psi}_{\ell+1}^H \hat{\theta}_{\ell}}{\tilde{V}_{\ell+1}^H \tilde{V}_{\ell+1} - \tilde{V}_{\ell+1}^H D_{\ell} \tilde{V}_{\ell+1}}. \]  

(30)

where \( \hat{\theta}_{\ell+1}^{k+1,m} = [\hat{\theta}_{\ell+1}^{k+1,1}, \ldots, \hat{\theta}_{\ell+1}^{k+1,m}]^T \).

V. EXAMPLE APPLICATION

A. Example system

As an example, we chose a simple serially concatenated encoding network. The outer code \( d \mapsto c \) is defined by the output \( c_n(d^{(i)}, \sigma^{(k)}) = c^{(k)} \), where \( \Sigma = [1, 3, 2, 4] \) (Matlab-like row-wise notation). The inner code \( \tilde{c} \mapsto q \) is \( q_n(\tilde{c}^{(i)}, \sigma^{(k)}) = q^{(k)} \), where \( \tilde{Q} = [1, 2, 3, 4] \). The state equation QPSK constellation mapping is

\[ s_n(q^{(k)}) = 1. \]  

(31)

The signal is passed through the AWGN channel with channel coefficients \( h_n = c^{(k)} \). The coefficients \( h_n' \) are generated by the subspace Fourier model with \( L = 3 \) (see Sec. IV-C). This means that only phase manipulation is considered in the channel. The true channel model is thus the phase rotation of the Fourier subspace model. The noise has the complex envelope power spectral density \( 2N_0 \). The average received bit energy to \( N_0 \) ratio is \( \gamma_B \).

The EM based iterative phase \( \varphi \) synchronizer uses the order recursive LS iterator with the subspace model for channel coefficients estimates \( \hat{\theta} \) (Sec. IV-C) generally different from the true one used for the channel coefficients generation. This demonstrates the dependence of the performance on the estimator subspace model fidelity. The synchronizer is equipped by the phase unwrapping algorithm needed due to the phase ambiguity of the QPSK constellation.
The performance is evaluated by characteristics relevant to the estimator itself, i.e. the phase error and MSE. Evaluation of the overall bit error rate would obscure the particular influence of the estimator algorithm. Fig. 2 demonstrates the differences in estimator performance for two different subspace models with the same dimensionality \( L \). The mean square error (MSE) for different subspace dimension and signal-to-noise ratio is evaluated in Fig. 3. Traditional iterative estimator schemes (assuming constant channel phase during the observation) fail completely in the time-variant channel.

VI. CONCLUSIONS AND DISCUSSION OF THE RESULTS

We have developed an iterative EM based IMD synchronizer for fast time-variant channel with subspace channel parametrization. This is inherently much more difficult task having much more critical dependence on the estimator optimality than the one for constant channel. Two major achievements were reached. The first, we identified a correspondence between the EM iterator and weighted LS problem. That was implemented using order recursive algorithm providing lower implementation complexity. The second achievement is in allowing a general subspace model for the channel transfer coefficients which can be adjusted easily for particular channel dynamics and substantially lowers the dimensionality of the problem.

The algorithm was verified by means of computer simulation on the simple example of the serially concatenated coded system. We have observed that the proper choice of the subspace model (with the same dimension \( L \)) can substantially decrease the required number of iterations (Fig. 2). The increase of the subspace dimension leads to the improvement of the estimator MSE behavior (Fig. 3). The dimension of the subspace appears to influence mainly the asymptotic value of the MSE while keeping a similar convergence speed.

REFERENCES