A STOCHASTIC METHOD FOR TRAINING BASED CHANNEL IDENTIFICATION

ABSTRACT

In this paper, we propose a new iterative stochastic method to identify convolutive channels when training sequences are inserted in the transmitted signal. We consider the case where the channel is quasi-static (i.e. the sampling period is several orders of magnitude below the coherence time of the channel). There are no requirements on the length of the training sequences and all the received symbols that contain contributions from the training symbols are exploited. The interference from the unknown data symbols surrounding the training sequences is considered as additive noise colored by the transmission channel. An iterative weighted least squares approach is used to filter out the contribution of both this interference term and the additive white gaussian noise term.

1. INTRODUCTION

The market need for high data rate communications has driven the research in telecommunications during the last years. In order to increase data rates when transmitting data over wireless channels, it is often needed to use broadband communication systems. A major impediment of such systems is that the sampling period can get smaller than the delay spread of the channel, especially in multipath scenarios. This results in ISI (Inter Symbol Interference), a phenomenon that needs to be combatted at the receiver in order to restore the transmitted information. This is usually done using serial or block equalization techniques. Channel State Information (CSI) is needed at the receiver in order to design the equalizer and combat the ISI in an efficient way.

The CSI is obtained through the use of channel identification algorithms. In this paper, we focus on the family of training-based channel identification algorithms. The problem of identifying the channel at the receiver when training sequences are inserted in the transmitted signals has been widely discussed in the literature. Classical channel identification methods [1] require training sequences that are longer than the channel order. In that case, the received symbols that only contain contributions from the known training sequences (as opposed to the received symbols that contain contributions from both the unknown data symbols and the training symbols or from the data symbols only) are exploited to identify the channel. This approach is sub-optimal since not all the received symbols that contain contributions from the training sequences are considered. More efficient techniques have been proposed recently in the framework of Known Symbol Padding (KSP) transmission [2], [3]. These techniques are deterministic and exploit all the energy that is received from the training sequences. These methods only work when a constant training sequence (zero or non-zero), which is at least as long as the channel order, is periodically inserted in the transmitted sequence.

The method we present here does not have such requirements on the length or composition of the training sequences. All the received symbols that contain contributions from the training sequences are used for the identification of the channel. The method is stochastic for it relies on the first and second order statistics of the transmitted signals; we consider quasi-static channels (the channel stays constant during the transmission of several training sequences). Under this assumption, the method can be used to identify the channel for all transmission schemes where training symbols are inserted in the data, which covers classical training-based transmission but also KSP transmission [4] or Pilot Symbol Assisted Modulation (PSAM) [5].

Notation: We use upper (lower) case bold face letters to denote matrices (column vectors). $\mathbf{I}_N$ is the identity matrix of size $N \times N$ and $\mathbf{O}_{M\times N}$ is the all-zero matrix of size $M \times N$. The operator $(.)^*$ denotes the complex conjugate, $(.)^T$ the transpose of a matrix and $(.)^H$ its complex conjugate transpose. Finally, $(.)^{1/2}$ represents a square root of a Matrix and $tr(.)$ its trace.

2. DATA MODEL

We consider stationary Finite Impulse Response (FIR) convolutive channels of order L: $\mathbf{h} = [h[0], h[1], \cdots, h[L]]^T$. A sequence $x[n]$ of symbols is transmitted over the channel. The received sequence $y[n]$ is the linear convolution of the transmitted sequence with the channel impulse response: $y[n] = \sum_{i=0}^{L} h[i]x[n-i] + \eta[n]$, where $\eta[n]$ is the Additive White Gaussian Noise (AWGN) at the receiver. A total number of K training sequences is inserted between the unknown data symbols. The kth training sequence, $\mathbf{t}_k = [t_k[1], \ldots, t_k[n_t]]^T$, starts at position n_k: $[x[n_k], \ldots, x[n_k + n_t - 1]]^T = \mathbf{t}_k$. We refer to the situation where the same training sequence is repeated ($\mathbf{t}_k = \mathbf{t}, \forall k$) as the constant

training case. We refer to the situation where all the training sequences differ as the changing training case.

Define the vector $\mathbf{u}_k$ of received symbols that contain a contribution from the kth transmitted training sequence: $\mathbf{u}_k = [y[n_k], \ldots, y[n_k + n_t + L - 1]]^T$. The vector $\mathbf{u}_k$ contains a contribution from the training sequence $\mathbf{t}_k$ plus an additional term that collects the contributions from both the unknown surrounding data symbols and the noise. We can thus describe $\mathbf{u}_k$ as the sum of a deterministic and a stochastic term:

$$\mathbf{u}_k = \mathbf{T}_k \mathbf{h} + \boldsymbol{\epsilon}_k, \tag{1}$$

where $\mathbf{T}_k$ is an $(n_t + L) \times (L + 1)$ tall Toeplitz matrix with $\left[\mathbf{t}_k^T, 0, \ldots, 0\right]^T$ as first column and $[t_k[1], 0, \ldots, 0]$ as first row. $\mathbf{T}_k\mathbf{h}$ is the deterministic term; the stochastic term, $\boldsymbol{\epsilon}_k$, is described as follows:

$$\boldsymbol{\epsilon}_k = \underbrace{\left[\begin{array}{ccc|ccc} h_L & \cdots & h_1 & & & \mathbf{0} \\ & \ddots & \vdots & h_0 & & \\ & & h_L & \vdots & \ddots & \\ \mathbf{0} & & & h_{L-1} & \cdots & h_0 \end{array}\right]}_{\mathbf{H}_s (n_t + L) \times (2L)} \mathbf{s}_k + \boldsymbol{\eta}_k, \tag{2}$$

where $\mathbf{s}_k = [x[n_k - L], \ldots, x[n_k - 1], x[n_k + n_t], \ldots, x[n_k + n_t + L]]^T$ is the vector of surrounding data symbols, and $\boldsymbol{\eta}_k$ is the corresponding AWGN term. Assuming that both the noise and the data are white and zero-mean ($E\{s_k[i]s_l[j]^*\} = E\{\eta[k]\eta[l]^*\} = 0, \ \forall i, j, k, l$ when $k \neq l$ or $i \neq j$, and $E\{s_k[i]\} = E\{\eta[k]\} = 0$), we can say that $\boldsymbol{\epsilon}_k$ is zero-mean. Defining the signal and noise variances as $\lambda^2 = E\{s_k[i]s_k[i]^*\}$ and $\sigma^2 = E\{\eta[k]\eta[k]^*\}$ respectively, we can derive the covariance matrix of $\boldsymbol{\epsilon}_k$ from (2) as $E\left\{\boldsymbol{\epsilon}_k\boldsymbol{\epsilon}_k^H\right\} \triangleq \mathbf{Q} = \lambda^2\mathbf{H}_s\mathbf{H}_s^H + \sigma^2\mathbf{I}$. Defining n_s as the length of the shortest sequence of data symbols ($n_s = \min_k \{n_{k+1} - (n_k + n_t - 1)\}$), we assume $n_s \geqslant 2L$. This ensures that the $\boldsymbol{\epsilon}_k$'s are uncorrelated. The first and second order statistics of the stochastic term are thus the following: $E\{\boldsymbol{\epsilon}_k\} = 0$ **(C1)**, $E\{\boldsymbol{\epsilon}_k\boldsymbol{\epsilon}_k^H\} = \mathbf{Q}$ **(C2)** and $E\{\boldsymbol{\epsilon}_k\boldsymbol{\epsilon}_l^H\} = 0, \ \forall k \neq l$ **(C3)**.

We can alternatively define a data model that collects all the K equations of (1) in a single relation. Define $\mathbf{T}$ as the collection of all the $\mathbf{T}_k$ matrices: $\mathbf{T} = \left[\mathbf{T}_1^H, \cdots, \mathbf{T}_K^H\right]^H$, $\mathbf{u}$ as the collection of all $\mathbf{u}_k$ vectors: $\mathbf{u} = \left[\mathbf{u}_1^H, \cdots, \mathbf{u}_K^H\right]^H$, and $\boldsymbol{\epsilon}$ as the collection of all $\boldsymbol{\epsilon}_k$ terms: $\boldsymbol{\epsilon} = \left[\boldsymbol{\epsilon}_1, \cdots, \boldsymbol{\epsilon}_K\right]^H$. Collecting the K equations from (1) then yields:

$$\mathbf{u} = \mathbf{T}\mathbf{h} + \boldsymbol{\epsilon}. \tag{3}$$

It is clear that $E\{\boldsymbol{\epsilon}\} = 0$. Define $\mathbf{Q}_{tot} = E\{\boldsymbol{\epsilon}\boldsymbol{\epsilon}^H\}$. We see from **C2** and **C3** that $\mathbf{Q}_{tot}$ is a block-diagonal matrix with $\mathbf{Q}$ repeated along the main diagonal.

3. STOCHASTIC CHANNEL IDENTIFICATION

3.1. LS Channel Estimate

It is straightforward to identify the channel using a Least Squares (LS) approach (see e.g. [6]). The LS estimate of the channel corresponding to the kth training sequence can be expressed as: $\hat{\mathbf{h}}_k^{LS} = \left(\mathbf{T}_k^H\mathbf{T}_k\right)^{-1}\mathbf{T}_k^H\mathbf{u}_k$. Due to **C1** this estimator is unbiased: $E\{\hat{\mathbf{h}}_k\} = \mathbf{h}$. Note that the inverse in this expression always exists since $\mathbf{T}_k$ is always full column rank, independently of the length and the composition of the training sequence. Relying on the larger data model (3), we can build a similar LS estimator that jointly uses all the transmitted training sequences in order to estimate the channel: $\hat{\mathbf{h}}_{LS} = \left(\mathbf{T}^H\mathbf{T}\right)^{-1}\mathbf{T}^H\mathbf{u}$, or equivalently:

$$\hat{\mathbf{h}}_{LS} = \left(\sum_{k=1}^{K}\mathbf{T}_k^H\mathbf{T}_k\right)^{-1}\sum_{k=1}^{K}\mathbf{T}_k^H\mathbf{u}_k. \tag{4}$$

3.2. WLS Channel Estimate

We know from **C2** that the interference plus noise term $\boldsymbol{\epsilon}_k$ present in the LS estimate proposed in the previous paragraph is not white. The accuracy of the estimate can thus be improved if we use a weighted least squares (WLS) approach that first whitens the noise term. Assume for now that the auto-correlation $\mathbf{Q}$ of the noise term is known at the receiver.The WLS channel estimate based on the kth training sequence can then be expressed as : $\hat{\mathbf{h}}_k^{WLS} = \left(\mathbf{T}_k^H\mathbf{Q}^{-1}\mathbf{T}_k\right)^{-1}\mathbf{T}_k^H\mathbf{Q}^{-1}\mathbf{u}_k$. Here again, this approach can be extended to the larger data model (3) to build a WLS estimator of the channel that jointly uses all the training sequences: $\hat{\mathbf{h}}_{WLS} = (\mathbf{T}^H\mathbf{Q}_{tot}^{-1}\mathbf{T})^{-1}\mathbf{T}^H\mathbf{Q}_{tot}^{-1}\mathbf{u}$. Exploiting the block-diagonal structure of $\mathbf{Q}_{tot}$ allows us to split the product of large matrices in a sum of products of smaller matrices. We can thus re-express the WLS channel estimate in a way that is computationally less demanding:

$$\hat{\mathbf{h}}_{WLS} = \left(\sum_{k=1}^{K}\mathbf{T}_k^H\mathbf{Q}^{-1}\mathbf{T}_k\right)^{-1}\sum_{k=1}^{K}\mathbf{T}_k^H\mathbf{Q}^{-1}\mathbf{u}_k \tag{5}$$

Note that $\mathbf{Q}^{-1}$ always exists and is always full rank thanks to the noise term in $\mathbf{Q}$. Additionally, because it is an auto-correlation matrix, $\mathbf{Q}$ is strictly positive definite. If we note that $\mathbf{T}_k$ is always full column rank, we see that all the terms in the summation on the left side of (5) are strictly positive definite as well. Taking into account that the sum of strictly positive definite matrices is a strictly positive definite matrix proves that the inverse in (5) always exists.

3.3. Iterative Method

Since $\mathbf{Q}$ is not known at the receiver, it is not possible to apply the proposed WLS method directly. We propose an

iterative method that overcomes this problem.

Assume a channel estimate $\hat{\mathbf{h}}^{(i)}$ is available at the receiver. Based on this channel estimate, it is possible to produce an estimate $\mathbf{Q}^{(i+1)}$ of $\mathbf{Q}$. The first possibility is to rely on the definitions of $\mathbf{H_s}$ and $\mathbf{Q}$ in order to produce a parametric estimate $\mathbf{Q}^{(i+1)}$; this approach requires an estimate of the noise power σ^2 at every iteration. The alternative is to use a non-parametric approach, which is what we do next. Using (1) together with the channel estimate $\hat{\mathbf{h}}^{(i)}$ allows us to estimate the interference plus noise term: $\hat{\boldsymbol{\epsilon}}_k^{(i+1)} = \mathbf{u}_k - \mathbf{T}_k\hat{\mathbf{h}}^{(i)}$. This estimate can be used to produce an estimate of the autocorrelation matrix:

$$\hat{\mathbf{Q}}^{(i+1)} = K^{-1}\sum_{k=1}^{K}\hat{\boldsymbol{\epsilon}}_k^{(i+1)}\hat{\boldsymbol{\epsilon}}_k^{(i+1)H} \tag{6}$$

We next compute $\hat{\mathbf{h}}^{(i+1)}$ using the WLS approach (5) where $\mathbf{Q}$ is replaced by its estimate $\hat{\mathbf{Q}}^{(i+1)}$. Note that in the WLS step we need this estimate $\hat{\mathbf{Q}}^{(i+1)}$ to be full rank which requires $K \geqslant n_t + L$.

The LS estimate of the channel (4) can be used as the starting point $\hat{\mathbf{h}}^{(0)}$ of the iterations, which is strictly equivalent to choosing $\hat{\mathbf{Q}}^{(0)} = \mathbf{I}$. The iterative procedure is stopped when there is no significant difference between two successive channel estimates.

4. EXPERIMENTAL RESULTS

The performance metric that is used throughout this section is the Normalized Mean Squared Error (NMSE) of the proposed channel estimate: $NMSE(\hat{\mathbf{h}}) = \frac{||\hat{\mathbf{h}}-\mathbf{h}||^2}{||\mathbf{h}||^2}$. The training sequences are random constant modulus sequences. The Signal to Noise Ratio (SNR) is defined as $SNR = \frac{||\mathbf{h}||^2\lambda^2}{\sigma^2}$. The experiments are performed on convolutive Rayleigh Fading channels of varying order L. The different taps of the channel are assumed to be independent identically distributed. The simulation results are obtained by averaging the performance over a set of 500 random channel realizations and over a set of 100 different training sequences for each of these channel realizations. A remarkable property of this new method is its fast convergence: experiments show that it converges in a single iteration at low SNRs.

4.1. Performance of the proposed method

In Fig. 1, we compare the performance (after one iteration) of the method when constant or changing training sequences are used. The experiments are done in a transmission scheme where the length of the training sequences is set to $n_T = 7$. We make this comparison for three different channel orders (L=2, L=6 and L=9). We set the number of observed training sequences to K=200. We first see that the method always performs better when changing training is used rather than constant training. Furthermore, we see that a floor appears at high SNRs in the NMSE of the channel estimate when the channel order increases. This saturation effect appears when there is no exact solution to the channel identification problem in the noiseless case. When changing training sequences are used, this happens for all channel orders $L \geqslant n_t$. When constant training sequences are used, the saturation occurs when $L \geqslant \frac{n_t-1}{2}$. The use of changing training sequences thus yields significantly more accurate channel estimates for intermediate channel orders.

Further experiments show as well that reducing the number of observed training sequences K results in reduced performance, which is not surprising. Additionally, when K gets too small, a floor NMSE appears at high SNRs independently of the channel order. Finally, these experiments clearly show the advantage of using changing training sequences rather than constant training sequence.

4.2. Comparison with existing methods

In Fig 2, we compare the NMSE of the channel as a function of the SNR expressed in dB for the proposed iterative WLS method after one iteration, the initial LS channel estimate and other existing methods, namely the classical ML channel estimate based on the received samples that are not corrupted by unknown data symbols and the method we proposed in [2] for KSP transmission. The comparison is done for two different numbers of observed training sequences, namely K=40 and K=160. The channel order is set to L=6 and the length of the training sequences is set to n_t=7 so that there is at least one received symbol per transmitted training sequence that has no contribution from the unknown surrounding data symbols. The comparisons are done using the same channel, noise and data realizations. We use changing training sequences, except for the method presented in [2] which requires the use of constant training. The experiments clearly show a saturation effect at high SNRs for the initial LS channel estimate. This saturation happens when the channel NMSE is dominated by the unknown data term. When our iterative WLS scheme is used, this data term is filtered out and the saturation effect disappears. When K is large, we see clearly that the iterative method outperforms the existing ones, especially at low SNRs. When K is smaller, our method still outperforms existing methods at low SNRs but we see that the ML method clearly outperforms the proposed iterative WLS at high SNRs. Also, the deterministic method presented in [2] performs slightly better than ours at high SNRs but this improved performance comes at the cost of a significantly higher computational complexity.

Keep in mind that whilst existing methods stop working as soon as the channel order outpasses the length of the training sequences (in which case their solution simply does not exist), our method is always able to provide us with re-

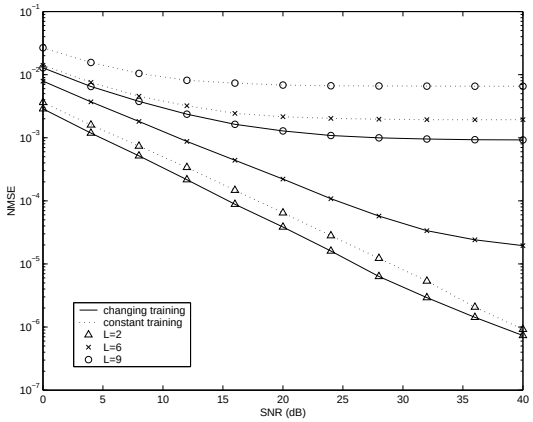

Figure 1: Comparison of the NMSE vs. SNR curves when constant or changing training sequences are used

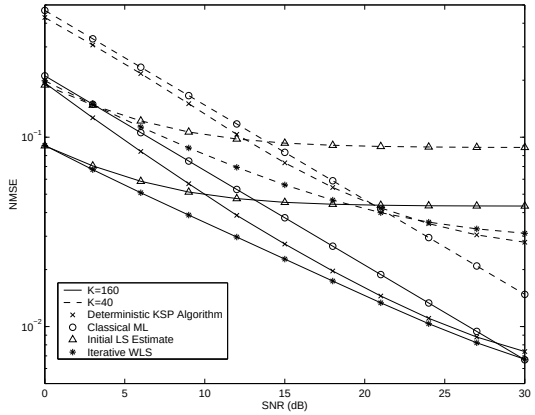

Figure 2: NMSE vs. SNR for different Channel Estimation Techniques

liable channel estimates whatever the actual length of the channel and is thus much more robust against severe multipath

These experiments clearly show that the use of the proposed iterative WLS method results in improved performance in most contexts. Only when a few training sequences can be used to identify the channel (if the channel changes too rapidly for instance), and under the additional condition that the event of having a channel order effectively higher than the length of the training sequence is highly unlikely, the use of the deterministic method presented in [2] or the use of classical ML channel identification yields more accurate performance at high SNRs.

5. CONCLUSION

An important feature of the iterative WLS channel identification method we present in this paper is its flexibility. Unlike existing methods, it can be used in any transmission scheme where training sequences are inserted without any requirement on the length and composition of the training sequences or on the channel order. Existing methods perform slightly better only when there is a small amount of training symbols available and the noise power is small, whereas the proposed method outperforms existing ones in other contexts. Additionally, when the actual channel order increases, our method keeps working whereas existing ones are not able to identify the channel anymore.

Other interesting points include the fast convergence properties as well as the low computational cost of the proposed method.

6. REFERENCES

[1] H. Vikalo, B. Hassibi, B. Hochwald, T. Kailath, "Optimal Training for Frequency-Selective Fading Channels," in *Proc. of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Salt Lake City, Utah, May 2001.

[2] G. Leus and M. Moonen, "Semi-Blind Channel Estimation for Block Transmission with Non-Zero Padding," in *Proc. of the Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, Nov. 4-7 2001.

[3] S. Barbarossa, A. Scaglione and G. B. Giannakis, "Performance Analysis of a Deterministic Channel Estimator for Block Transmission Systems with Null Guard Intervals," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 684–695, March 2002.

[4] L. Deneire, B. Gyselinckx and M. Engels., "Training Sequence vs. Cyclic Prefix: A New Look on Single Carrier Communication," *IEEE Communication Letters*, vol. 5, no. 7, pp. 292–294, 2001.

[5] J. K. Cavers, "An analysis of pilot symbol assisted modulation for Rayleigh fading channels (mobile radio)," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 4, pp. 686–693, November 1991.

[6] O. Rousseaux, G. Leus, M. Moonen, "A suboptimal Iterative Method for Maximum Likelihood Sequence Estimation in a Multipath Context," *EURASIP Journal on Applied Signal Processing (JASP)*, vol. 2002, no. 12, pp. 1437–1447, December 2002.