This research has been supported in part by European Commission
FP6 IYTE-Wireless Project (Contract No: 017442)

Best Linear Unbiased Channel Estimation for Frequency Selective Multipath Channels with Long Delay Spreads

Christopher Pladdy*, Serdar Özen*, Mark J. Fimoff*, S. M. Nerayanuru* and Michael D. Zoltowski†
*Zenith Electronics Corp., R & D
Lincolnshire, IL 60069 - USA
Phone: (847) 941 8797
{christopher.pladdy, serdar.ozen, mark.fimoff, snerayanuru}@zenith.com
†School of Electrical & Computer Engineering,
Purdue University
W. Lafayette, IN 47907-1285 - USA
Phone: (765) 494-3512
mikedz@ecn.purdue.edu

***Abstract*— We provide an iterative channel impulse response (CIR) estimation algorithm for communication systems which utilize a periodically transmitted training sequence within a continuous stream of information symbols. This iterative procedure calculates the (semi-blind) *Best Linear Unbiased Estimate* (BLUE) of the CIR. We first provide a formulation of the received data and correlation processing with the adjacent information symbol correlation taken into account, and we then present the connections of the correlation based CIR estimation scheme to the ordinary least squares CIR estimation. Simulation results are provided to demonstrate the performance of the novel algorithm.**

I. Introduction

For communications systems utilizing a periodically transmitted training sequence, *least-squares* (LS) based channel estimation or *correlation* based channel estimation algorithms have been the two most widely used alternatives [1]. Both methods use a stored copy of the known transmitted training sequence at the receiver. The properties and the length of the training sequence are generally different depending on the particular communication system's standard specifications. However most channel estimation schemes do not account for the *baseline noise* term which occurs due to the correlation of the stored copy of the training sequence with the unknown symbols adjacent to transmitted training sequence, as well as the additive channel noise [1], [8]. In the sequel, although the examples following the derivations of the BLUE channel estimator will be drawn from the ATSC digital TV 8-VSB system [2], to the best of our knowledge it could be applied with minor modifications to any digital communication system with linear modulation which employs a periodically transmitted training sequence. The novel algorithm presented in the sequel is targeted for the systems that are desired to work with channels having long delay spreads L_d; in particular we consider the case where $(NT+1)/2 < L_d < NT$, where NT is the duration of the available training sequence. For instance the 8-VSB digital TV system has 728 training symbols, whereas the delay spreads of the terrestrial channels have been observed to be at least 400-500 symbols long [5], [6].

II. Baseband Data Transmission Model

The baseband symbol rate sampled receiver pulse-matched filter output is given by

$$y[n] \equiv y(t)|_{t=nT} = \sum_k I_k h[n-k] + \nu[n], \quad (1)$$

$$= \sum_k I_k h[n-k] + \sum_k \eta[k] q^*[-n+k] \quad (2)$$

where

$$I_k = \begin{cases} a_k, & 0 \le k \le N-1 \\ d_k, & N \le n \le N'-1, \end{cases} \in \mathcal{A} \equiv \{\alpha_1, \cdots, \alpha_M\} \quad (3)$$

is the M-ary complex valued transmitted sequence, $\mathcal{A} \subset \mathbb{C}^1$, and $\{a_k\} \in \mathbb{C}^1$ denote the first N symbols within a *frame* of length N' to indicate that they are the known training symbols; $\{d_k\} \in \mathbb{C}^1$ denote the remaining $N'-N$ random data within a frame; $\nu(t) = \eta(t) * q^*(-t)$ denotes the complex (colored) noise process after the receiver (pulse) matched filter, with $\eta(t)$ being a zero-mean white Gaussian noise process with spectral density σ_η^2 per real and imaginary part; $h(t)$ is the complex valued impulse response of the composite channel, including pulse shaping transmit filter $q(t)$, the physical channel impulse response $c(t)$, and the receive filter $q^*(-t)$, and is given by

$$h(t) = p(t) * c(t) = \sum_{k=-K}^{L} c_k p(t - \tau_k), \quad (4)$$

and $p(t) = q(t) * q^*(-t)$ is the convolution of the transmit and receive filters where $q(t)$ has a finite support of $[-T_q/2, T_q/2]$, and the span of the transmit and receive filters, T_q, is integer multiple of the symbol period, T; that is $T_q = N_q T = 2L_q$, $N_q = 2L_q \in \mathbb{Z}^+$. $\{c_k\} \subset \mathbb{C}^1$ denote complex valued physical channel gains, and $\{\tau_k\}$ denote the multipath delays, or the Time-Of-Arrivals (TOA). It is assumed that the time-variations of the channel are slow enough that $c(t)$ can be assumed to be a static inter-symbol interference (ISI) channel, at least throughout the training period. We also note that for 8-VSB system [2] the transmitter pulse shape is the Hermitian symmetric root-raised cosine pulse, which implies $q(t) = q^*(-t)$. In the sequel $q[n] \equiv q(t)|_{t=nT}$ will be used to denote both the transmit and receive filters. In the sequel the sampled matched filter output signal $y[n]$ will be used extensively in vector form, and to help minimize introducing new variables, the notation of $\boldsymbol{y}_{[n_1:n_2]}$ with $n_2 \geq n_1$, will be adopted to indicate the column vector $\boldsymbol{y}_{[n_1:n_2]} = [y[n_1], y[n_1+1], \cdots, y[n_2]]^T$. Same notation will also be applied to the noise variables $\eta[n]$ and $\nu[n]$.

Without loss of generality, symbol rate sampled, complex valued, composite CIR $h[n]$ can be written as a finite dimensional vector

$$\boldsymbol{h} = [h[-N_a], \cdots, h[-1], h[0], h[1], \cdots, h[N_c]]^T \quad (5)$$

where N_a and N_c denote the number of anti-causal and the causal taps of the channel, respectively, and are given by

$$N_a = \text{round}\left\{\frac{\tau_{-K} - TN_q}{T}\right\}, \text{ and } N_c = \text{round}\left\{\frac{\tau_L + TN_q}{T}\right\},$$

and $L_d = (N_a + N_c + 1)T$ is the delay spread of the channel (including the pulse tails), or equivalently $N_a + N_c + 1$ is the total memory of the channel. Based on Equation (2) and assuming that $N \geq N_a + N_c + 1$, we can write the pulse matched filter output corresponding *only* to the known training symbols compactly as

$$\begin{aligned} \boldsymbol{y}_{[N_c:N-N_a-1]} &= \tilde{\boldsymbol{A}}\boldsymbol{h} + \boldsymbol{\nu}_{[N_c:N-N_a-1]} & (6)\\ &= \tilde{\boldsymbol{A}}\boldsymbol{h} + \tilde{\boldsymbol{Q}}\boldsymbol{\eta}_{[N_c-L_q:N-1-N_a+L_q]}, & (7)\end{aligned}$$

where

$$\begin{aligned} \tilde{\boldsymbol{A}} &= \mathcal{T}\left\{[a_{N_c+N_a}, \cdots, a_{N-1}]^T, [a_{N_c+N_a}, \cdots, a_0]\right\} & (8)\\ &= \begin{bmatrix} a_{N_c+N_a} & a_{N_c+N_a-1} & \cdots & a_0 \\ a_{N_c+N_a+1} & a_{N_c+N_a} & \cdots & a_1 \\ \vdots & \vdots & \ddots & \vdots \\ a_{N-1} & a_{N-2} & \cdots & a_{N-1-N_a-N_c} \end{bmatrix}, \end{aligned}$$

where $\tilde{\boldsymbol{A}}$ is $(N - N_a - N_c) \times (N_a + N_c + 1)$ Toeplitz convolution matrix with first column $[a_{N_c+N_a}, \cdots, a_{N-1}]^T$ and first row $[a_{N_c+N_a}, \cdots, a_0]$, and $\boldsymbol{\nu}_{[N_c:N-N_a]} = \tilde{\boldsymbol{Q}}\boldsymbol{\eta}_{[N_c-L_q:N-1-N_a+L_q]}$ is the colored noise at the receiver matched filter output, with

$$\begin{aligned} \tilde{\boldsymbol{Q}} &= \begin{bmatrix} \boldsymbol{q}^T & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}^T \end{bmatrix}_{(N-N_a-N_c)\times(N-N_a-N_c+N_q)} & (9)\\ \boldsymbol{q} &= [q[+L_q], \cdots, q[0], \cdots, q[-L_q]]^T, & (10)\end{aligned}$$

where $\boldsymbol{q}$ denotes the symbol rate sampled receiver pulse matched filter.

Similarly the pulse matched filter output which includes *all* the contributions from the known training symbols (which includes adjacent random data as well) can be written as

$$\begin{aligned} \boldsymbol{y}_{[-N_a:N+N_c-1]} &= (\boldsymbol{A} + \boldsymbol{D})\,\boldsymbol{h} + \boldsymbol{\nu}_{[-N_a:N+N_c-1]} & (11)\\ &= \boldsymbol{A}\boldsymbol{h} + \boldsymbol{D}\boldsymbol{h} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}, & (12)\end{aligned}$$

where

$$\boldsymbol{A} = \mathcal{T}\left\{[a_0, \cdots, a_{N-1}, \underbrace{0, \cdots, 0}_{N_a+N_c}]^T, [a_0, \underbrace{0, \cdots, 0}_{N_a+N_c}]\right\} \quad (13)$$

is a $(N + N_a + N_c) \times (N_a + N_c + 1)$ Toeplitz matrix with first column $[a_0, a_1, \cdots, a_{N-1}, 0, \cdots, 0]^T$, and first row $[a_0, 0, \cdots, 0]$, and

$$\begin{aligned} \boldsymbol{D} = \mathcal{T}\Big\{&[\underbrace{0, \cdots, 0}_{N}, d_N, \cdots, d_{N_c+N_a+N-1}]^T, & (14)\\ &[0, \underbrace{d_{-1}, \cdots, d_{-N_c-N_a}}_{\text{data from previous frame}}]\Big\}, & (15)\end{aligned}$$

is a Toeplitz matrix which includes adjacent random information symbols only, prior to and after the training sequence. The data sequence $[d_{-1}, \cdots, d_{-N_c-N_a}]$ is the unknown information symbols transmitted at the end of the frame prior to the current frame being transmitted. $\boldsymbol{Q}$ is of dimension $(N+N_a+N_c) \times (N + N_a + N_c + N_q)$ and has the same convolution matrix structure with $\tilde{\boldsymbol{Q}}$ as displayed in Equation (9). We now write the contributions of the unknown symbols $\boldsymbol{D}\boldsymbol{h}$ in Equation (12) in a different format which will prove to be more useful in the subsequent derivations.

We first define $\boldsymbol{d} = \boldsymbol{S}\tilde{\boldsymbol{d}}$, or equivalently $\tilde{\boldsymbol{d}} = \boldsymbol{S}^T\boldsymbol{d}$, where

$$\begin{aligned} \tilde{\boldsymbol{d}} &= [d_{-N_c-N_a}, \cdots, d_{-1}, \boldsymbol{0}_{1\times N}, d_N, \cdots, d_{N+N_c+N_a-1}]^T & (16)\\ \boldsymbol{d} &= [d_{-N-N_a}, \cdots, d_{-1}, d_N, \cdots, d_{N+N_c+N_a-1}]^T & (17)\\ \boldsymbol{S} &= \begin{bmatrix} \boldsymbol{I}_{N_a+N_c} & \boldsymbol{0}_{(N_a+N_c)\times N} & \boldsymbol{0}_{(N_a+N_c)} \\ \boldsymbol{0}_{(N_a+N_c)} & \boldsymbol{0}_{(N_a+N_c)\times N} & \boldsymbol{I}_{N_a+N_c} \end{bmatrix} & (18)\end{aligned}$$

where $\boldsymbol{S}$ is $(2(N_c + N_a)) \times (N + 2(N_a + N_c))$ dimensional *selection* matrix which retains the random data, eliminates the N zeros in the middle of the vector $\tilde{\boldsymbol{d}}$. We also introduce

$$\begin{aligned} \boldsymbol{\mathcal{H}} &= \begin{bmatrix} \bar{\boldsymbol{h}}^T & 0 & \cdots & 0 \\ 0 & \bar{\boldsymbol{h}}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \bar{\boldsymbol{h}}^T \end{bmatrix}_{(N+N_c+N_a)\times(N+2(N_a+N_c))} & (19)\\ \bar{\boldsymbol{h}} &= \boldsymbol{J}\boldsymbol{h} & (20)\\ \boldsymbol{J} &= \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 1 & 0 \\ \vdots & \vdots & & \vdots \\ 1 & 0 & \cdots & 0 \end{bmatrix}_{(N_a+N_c+1)\times(N_a+N_c+1)} & (21)\\ \boldsymbol{H} &= \boldsymbol{\mathcal{H}}\boldsymbol{S}^T & (22)\end{aligned}$$

where $\bar{\boldsymbol{h}}$ is the time reversed version of $\boldsymbol{h}$ (re-ordering is accomplished by the permutation matrix $\boldsymbol{J}$), and $\boldsymbol{H}$ is of dimension $(N+N_a+N_c)\times(2(N_c+N_a))$ with a "hole" inside which is created by the selection matrix $\boldsymbol{S}$ as defined in Equation (18). Then it is trivial to show that

$$\boldsymbol{D}\boldsymbol{h}=\boldsymbol{\mathcal{H}}\tilde{\boldsymbol{d}}=\boldsymbol{\mathcal{H}}\boldsymbol{S}^T\boldsymbol{d}=\boldsymbol{H}\boldsymbol{d}. \tag{23}$$

Based on the Equations (16-23) we can rewrite Equation (12) as

$$\boldsymbol{y}_{[-N_a:N+N_c-1]}=\boldsymbol{A}\boldsymbol{h}+\boldsymbol{H}\boldsymbol{d}+\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}. \tag{24}$$

III. Overview of Generalized Least Squares

Consider the linear model

$$\boldsymbol{y} \;=\; \boldsymbol{A}\boldsymbol{x}+\boldsymbol{\nu} \tag{25}$$

where $\boldsymbol{y}$ is the observation (or response) vector, $\boldsymbol{A}$ is the regression (or design) matrix, $\boldsymbol{x}$ is the vector of unknown parameters to be estimated, and $\boldsymbol{\nu}$ is the observation noise (or measurement error) vector. Assuming that it is known that the random noise vector $\boldsymbol{\nu}$ is zero mean, and is correlated, that is $\text{Cov}\{\boldsymbol{\nu}\}=\boldsymbol{K}_\nu\equiv\frac{1}{2}E\{\boldsymbol{\nu}\boldsymbol{\nu}^H\}\neq\sigma_{\boldsymbol{\nu}}^2\boldsymbol{I}$, we define the (generalized) objective function for the model of (25) by

$$J_{GLS}(\boldsymbol{x}) \;=\; (\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x})^H\boldsymbol{K}_\nu^{-1}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}). \tag{26}$$

The least squares estimate that minimizes Equation (26) is

$$\hat{\boldsymbol{x}}_{gls} \;=\; (\boldsymbol{A}^H\boldsymbol{K}_\nu^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}^H\boldsymbol{K}_\nu^{-1}\boldsymbol{y}, \tag{27}$$

The generalized least squares estimate $\hat{\boldsymbol{x}}_{gls}$ given by Equation (27) is called the *best linear unbiased estimate* (BLUE) [7] among all *linear* unbiased estimators if the noise covariance matrix is *known* to be $\text{Cov}\{\boldsymbol{\nu}\}=\boldsymbol{K}_\nu$. If the noise $\boldsymbol{\nu}$ is *known* to be *Gaussian* with zero mean and with covariance matrix $\boldsymbol{K}_\nu$, that is if it is known that $\boldsymbol{\nu}\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{K}_\nu)$, then the estimator of (27) is called the *minimum variance unbiased estimator* (MVUE) among *all* unbiased estimators (not only linear).

IV. Overview of the Proposed CIR Estimator

For comparison purposes we first provide the well known correlation and ordinary least squares based estimators, where correlations based estimation is denoted $\hat{\boldsymbol{h}}_u$ (the subscript $_u$ stands for the *uncleaned* CIR estimate). Cross correlating the stored training sequence with the received sequence, which is readily available in digital receivers for the primary purpose of *frame synchronization* [3], yields a raw channel estimate

$$\tilde{h}_u[n]=\frac{1}{r_a[0]}\sum_{k=0}^{N-1}a_k^*y[k+n], n=-N_a,\cdots,0,\cdots,N_c \tag{28}$$

where $r_a[0]=\sum_{k=0}^{N-1}\|a_k\|^2$. Equivalently Equation (28) can be written as

$$\hat{\boldsymbol{h}}_u \;=\; \frac{1}{r_a[0]}\boldsymbol{A}^H\boldsymbol{y}_{[-N_a:N+N_c-1]}. \tag{29}$$

Substituting Equation (24) into (29) we get

$$\hat{\boldsymbol{h}}_u=\frac{1}{r_a[0]}\boldsymbol{A}^H\Big(\boldsymbol{A}\boldsymbol{h}+\boldsymbol{H}\boldsymbol{d}+\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\Big). \tag{30}$$

In order to get rid of the sidelobes of the aperiodic autocorrelation we can simply invert the normalized autocorrelation matrix $\boldsymbol{R}_{aa}$ of the training symbols, defined by

$$\boldsymbol{R}_{aa} \;=\; \frac{1}{r_a[0]}\boldsymbol{A}^H\boldsymbol{A}. \tag{31}$$

Then the *cleaned* channel estimate $\hat{\boldsymbol{h}}_c$ (the subscript $_c$ stands for the *cleaned* CIR estimate) is obtained from

$$\begin{aligned}\hat{\boldsymbol{h}}_c &= \boldsymbol{R}_{aa}^{-1}\hat{\boldsymbol{h}}_u & (32)\\ &= (\boldsymbol{A}^H\boldsymbol{A})^{-1}\boldsymbol{A}^H\boldsymbol{y}_{[-N_a:N+N_c-1]}. & (33)\end{aligned}$$

Substituting Equation (30) into (32) we get

$$\hat{\boldsymbol{h}}_c=\boldsymbol{h}+\underbrace{\Big(\boldsymbol{A}^H\boldsymbol{A}\Big)^{-1}\boldsymbol{A}^H\Big(\boldsymbol{H}\boldsymbol{d}+\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\Big)}_{\text{base-line noise}}. \tag{34}$$

As can be seen from Equation (34) the channel estimate $\hat{\boldsymbol{h}}_c$ has the contributions due to unknown symbols prior to and after the training sequence, which are elements of the vector $\boldsymbol{d}$, as well as the additive channel noise; only the sidelobes due to aperiodic auto-correlation is removed. The term $\Big(\boldsymbol{A}^H\boldsymbol{A}\Big)^{-1}\boldsymbol{A}^H\Big(\boldsymbol{H}\boldsymbol{d}+\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\Big)$ is called *baseline noise* in the channel estimate [3]. Also note that the channel estimate of Equation (33) can be recognized as the (ordinary) least-squares estimate[1].

We can denote the two terms on the right side of Equation (24) by $\boldsymbol{v}=\boldsymbol{H}\boldsymbol{d}+\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}$. Hence we rewrite (24) as

$$\boldsymbol{y}_{[-N_a:N+N_c-1]} \;=\; \boldsymbol{A}\boldsymbol{h}+\boldsymbol{v}. \tag{35}$$

By noting the statistical independence of the random vectors $\boldsymbol{d}$ and $\boldsymbol{\eta}$, and also noting that both vectors are zero mean, the covariance matrix, $\boldsymbol{K}_{\boldsymbol{v}}$ of $\boldsymbol{v}$ is given by

$$\text{Cov}\{\boldsymbol{v}\}=\boldsymbol{K}_{\boldsymbol{v}}\equiv\frac{1}{2}E\{\boldsymbol{v}\boldsymbol{v}^H\}\;=\;\frac{\mathcal{E}_d}{2}\boldsymbol{H}\boldsymbol{H}^H+\sigma_\eta^2\boldsymbol{Q}\boldsymbol{Q}^H, \tag{36}$$

where $\mathcal{E}_d$ is the energy of the transmitted information symbols, and equals to 21 if the symbols $\{d_k\}$ are chosen from the set $\{\pm1,\pm3,\pm5,\pm7\}$. Realizing that the model of (35) can be seen as the general linear model of Equation (25), and using the same arguments summarized in Section III the generalized least squares objective function to be minimized is written as $J_{GLS}(\boldsymbol{h})=\Big(\boldsymbol{y}_{[-N_a:N+N_c-1]}-\boldsymbol{A}\boldsymbol{h}\Big)^H\boldsymbol{K}_{\boldsymbol{v}}^{-1}\Big(\boldsymbol{y}_{[-N_a:N+N_c-1]}-\boldsymbol{A}\boldsymbol{h}\Big)$. Then the generalized least-squares solution to the model of Equation (35) which minimizes the objective function $J_{GLS}(\boldsymbol{h})$ is given by

$$\hat{\boldsymbol{h}}_K \;=\; (\boldsymbol{A}^H\boldsymbol{K}_{\boldsymbol{v}}^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}^H\boldsymbol{K}_{\boldsymbol{v}}^{-1}\boldsymbol{y}_{[-N_a:N+N_c-1]}. \tag{37}$$

The problem with Equation (37) is that the channel estimate $\hat{\boldsymbol{h}}_K$ is based on the covariance matrix $\boldsymbol{K}_{\boldsymbol{v}}$, which is a function of the true channel impulse response vector $\boldsymbol{h}$ as well as the channel noise variance σ_η^2. In actual applications the BLUE channel estimate of Equation (37) can not be exactly obtained. Hence we need an *iterative* technique to calculate generalized least squares estimate of (37) where every iteration produces an updated estimate of the covariance matrix as well as the noise variance. Due to space limitations without going into the details, a simplified version of the iterations, which yield a closer approximation to the exact BLUE CIR estimate after each step, is provided in Algorithm 1. In the algorithm noise variance for each step $k = 0, 1, \ldots, N_{iter}$ is estimated by

$$\widehat{\sigma_\eta^2}[k] = \frac{1}{2\mathcal{E}_q(N - N_a - N_c)} \|\hat{\boldsymbol{y}}[k]_{[N_c:N-N_a]} - \boldsymbol{y}_{[N_c:N-N_a]}\|^2, \quad (38)$$

where $\mathcal{E}_q = \|\boldsymbol{q}\|^2$ and $\hat{\boldsymbol{y}}[k]_{[N_c:N-N_a]} = \tilde{\boldsymbol{A}}\hat{\boldsymbol{h}}^{(th)}[k]$, where $\tilde{\boldsymbol{A}}$ is given in (8).

Due to space limitations we can not provide the details of the *thresholding* steps (**[2]**, **[4-c]**) of the algorithm. A fixed predetermined threshold can be used at the initial thresholding step and this initial threshold could be refined as the CIR estimate gets better after each iteration. In addition other heuristic or statistical thresholding techniques may also be utilized to detect the presence of the non-zero channel taps within baseline noise, which is shown for the case of the ordinary LS-CIR estimation in (34). For further details regarding thresholding techniques readers are referred to [4].

Algorithm 1 Iterative Algorithm to obtain a CIR estimate via Generalized Least-Squares

[1] Get an initial CIR estimate $\hat{\boldsymbol{h}}[0]$ (use either (29) or (33));
[2] Threshold the initial CIR estimate, and denote it $\hat{\boldsymbol{h}}^{(th)}[0]$;
[3] Estimate the noise variance $\widehat{\sigma_\eta^2}[0]$;
[4]
for $k = 1, \ldots, N_{iter}$ **do**
  [4-a] Compute the inverse of the (estimated) covariance matrix $\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k] = \left[\frac{\mathcal{E}_d}{2}\boldsymbol{H}(\hat{\boldsymbol{h}}^{(th)}[k-1])\boldsymbol{H}^H(\hat{\boldsymbol{h}}^{(th)}[k-1]) + \widehat{\sigma_\eta^2}[k-1]\boldsymbol{Q}\boldsymbol{Q}^H\right]^{-1}$;
  [4-b] $\hat{\boldsymbol{h}}_K[k] = (\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k]\boldsymbol{A})^{-1}\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k]\boldsymbol{y}_{[-N_a:N+N_c-1]}$;
  [4-c] Threshold the CIR estimate $\hat{\boldsymbol{h}}_K[k]$, and denote it $\hat{\boldsymbol{h}}^{(th)}[k]$;
  [4-d] Estimate the noise variance $\widehat{\sigma_\eta^2}[k]$.
end for

A. Complexity of the Algorithm

The complexity estimate for a single iteration of the algorithm is given by the following:

Step **[4-a]**:

1) To compute $\boldsymbol{H}(\hat{\boldsymbol{h}}^{(th)}[k-1])\boldsymbol{H}^H(\hat{\boldsymbol{h}}^{(th)}[k-1])$ takes $2(N_a + N_c)(N + N_a + N_c)^2$ multiplications
2) To compute $\boldsymbol{Q}\boldsymbol{Q}^H$ takes $(N+N_a+N_c+N_q)(N+N_q+N_c)^2$ multiplications
3) To compute $\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}$ takes $(N+N_a+N_c)^3$ multiplications

Step **[4-b]**:

1) To compute $\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k]\boldsymbol{A}$ takes $(N_a+N_c+1)(N+N_a+N_c)^2 + (N_a + N_c + 1)^2(N + N_a + N_c)$ multiplications
2) To compute $(\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k]\boldsymbol{A})^{-1}$ takes $(N_a + N_c + 1)^3$ multiplications
3) To compute $\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k]\boldsymbol{y}_{[-N_a:N+N_c-1]}$ takes $(N + N_a + N_c)^2$ multiplications
4) To compute $\boldsymbol{A}^H(\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k]\boldsymbol{y}_{[-N_a:N+N_c-1]})$ takes $(N_a + N_c + 1)(N + N_a + N_c)$ multiplications
5) To compute $\hat{\boldsymbol{h}}_K[k]$ takes $(N_a+N_c+1)^2$ multiplications.

We note that the multiplication count of $O(N^3)$ to invert an $N \times N$ matrix which we have included in our complexity estimate is the worst case scenario. Asymptotically, the complexity estimate is dominated by this term (i.e., the asymptotic complexity of the algorithm is $O((N+N_a+N_c)^3)$) and this is the worst case scenario without taking any special structures of the matrices into account. We are presently working on versions the algorithm which are more economical in terms of the complexity, specifically versions of the algorithm where matrix inversions are not explicitly computed.

TABLE I
SIMULATED CHANNEL DELAYS IN SYMBOL PERIODS AND RELATIVE GAINS ($K = 2$ *pre-cursor* GHOSTS, $L = 6$ *post-cursor* GHOSTS)

Channel taps	Delay $\{\tau_k\}$	Gain $\{\lvert c_k\rvert\}$
$k = -2$	-60.277	0.55
$k = -1$	-0.957	0.7263
Main $k = 0$	0	1
$k = 1$	3.551	0.6457
$k = 2$	15.250	0.9848
$k = 3$	24.032	0.7456
$k = 4$	29.165	0.8616
$k = 5$	221.2345	0.6150
$k = 6$	332.9810	0.4900

V. SIMULATIONS

We considered an 8-VSB [2] receiver with a single antenna. 8-VSB system has a complex raised cosine pulse shape with roll-off factor $\beta = 0.115$ [2]. The CIR we considered is given in Table I. The phase angles of individual paths for all the channels are taken to be $\arg\{c_k\} = \exp(-j2\pi f_c\tau_k)$, for $k = -2, \cdots, 6$ where $f_c = \frac{50}{T_{sym}}$ and $T_{sym} = 92.9$nsec. According to the tap delays given in Table I and having $N_q = 60$, the delay spread including the pulse tails is $L_d \approx (60 + 333 + 2N_q)T = 513T \approx 50\mu$sec. The simulations were run at 28dB Signal-to-Noise-Ratio (SNR) measured at the input to the receive pulse matched filter, and it is calculated by

$$\text{SNR} = \frac{\mathcal{E}_d \|\{c(t) * q(t)\}_{t=kT}\|^2}{\sigma_\eta^2}. \quad (39)$$

Figure 1-5 shows the simulation results for the test channel provided in Table I. Figure 1 shows the real part of the actual CIR. Figure 2 shows the correlation based CIR estimate $\hat{\boldsymbol{h}}_u$, of Equation (29). Figure 3 shows the LS based CIR estimate $\hat{\boldsymbol{h}}_c$, of Equation (33). Figure 4 show the BLUE based CIR estimate, $\hat{\boldsymbol{h}}_K[1]$, after the first iteration only. Figure 5 where we assumed that the covariance matrix $\boldsymbol{K}_{\boldsymbol{v}}$ is *known*, provides a bound for the BLUE iterations. Note that, since knowing the true covariance matrix $\boldsymbol{K_v}$ implies that the channel convolution matrix and the noise power are also known, the result of Figure 5 can only be reached asymptotically in practice, and hence serves as a bound for the BLUE iterations. We note superior performance of the BLUE algorithm even after the first iteration, as compared to the correlation based and ordinary least squares based CIR estimation schemes. The performance measure is the normalized least-squares error which is defined by $\mathcal{E}_{NLS} = \frac{\|\boldsymbol{h}-\hat{\boldsymbol{h}}\|^2}{N_a+N_c+1}$.

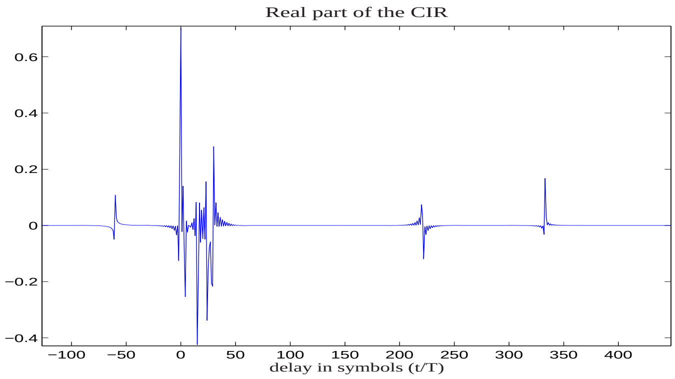

Fig. 1. The real part of the actual CIR where the time delays and gains are given in Table I.

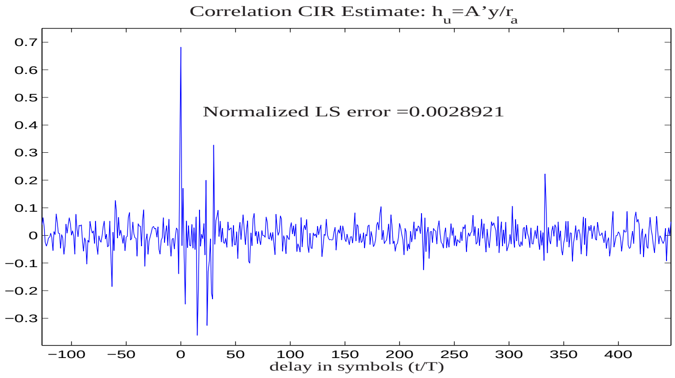

Fig. 2. The correlation based CIR estimate $\hat{\boldsymbol{h}}_u$ of Equation (29).

ACKNOWLEDGMENTS

This research was funded in part by the National Science Foundation under grant number CCR-0118842.

REFERENCES

[1] H. Arslan, G. E. Bottomley, "Channel estimation in narrowband wireless communication systems," Wireless Comunications and Mobile Computing, vol. 1, pp. 201-219, 2001.

[2] ATSC Digital Television Standard, A/53, September 1995. (available from http://www.atsc.org/standards.html)

[3] M. Fimoff, S. Özen, S.M. Nereyanuru, M.D. Zoltowski, W. Hillery, "Using 8-VSB Training Sequence Correlation as a Channel Estimate for DFE Tap Initialization," Proceedings of the 39th Annual Allerton Conference in Communications, Control and Computing, September 2001.

[4] S. Özen, "Topics on channel estimation and equalization for sparse channels with applications to digital TV systems," Ph.D. Dissertation, Purdue University, School of Electrical and Computer Engineering, West Lafayette, Indiana, May 2003.

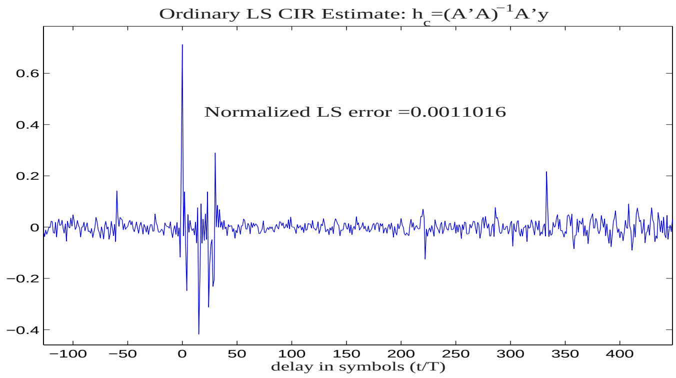

Fig. 3. LS based CIR estimate $\hat{\boldsymbol{h}}_c$ of Equation (33).

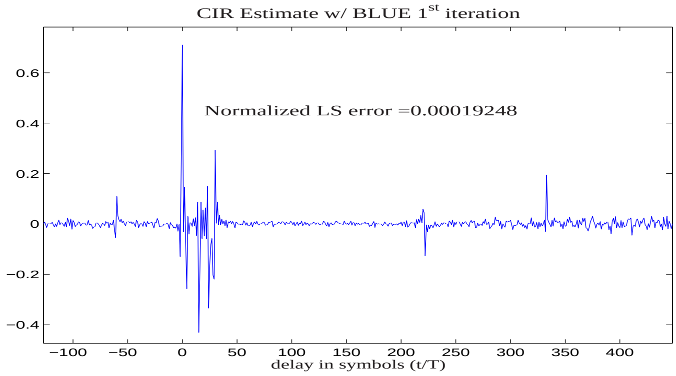

Fig. 4. BLUE based CIR estimate of Algorithm 1, $\hat{\boldsymbol{h}}_K[1]$, after the first iteration only.

[5] S. Özen, M. D. Zoltowski, M. Fimoff, "A Novel Channel Estimation Method: Blending Correlation and Least-Squares Based Approaches," Proceedings of ICASSP, v. 3, pp. 2281-2284, 2002.

[6] S. Özen, W. Hillery, M. D. Zoltowski, S. M. Nerayanuru, M. Fimoff, "Structured Channel Estimation Based Decision Feedback Equalizers for Sparse Multipath Channels with Applications to Digital TV Receivers," Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, vol. 1, pp. 558-564, Nov. 3-6, 2002.

[7] G. A. F. Seber, "*Linear Regression Analysis*," John-Wiley and Sons, 1977.

[8] H-K Song, "A channel estimation using sliding window aproach and tuning algorithm for MLSE," IEEE Communications Letters, vol. 3, pp. 211-213, 1999.

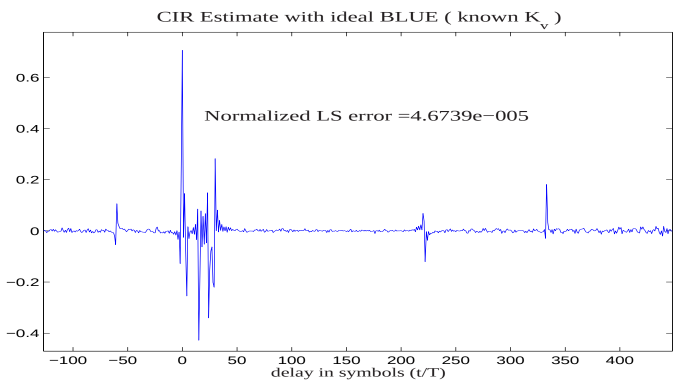

Fig. 5. BLUE CIR estimate with known $\boldsymbol{K_v}$. This plot serves as a bound for BLUE iterations.