Taylor series approximation of semi-blind BLUE channel estimates with applications to DTV

Christopher Pladdy , Serdar Özen , S. M. Nerayanuru , Peilu Ding , Mark J. Fimoff & Michael Zoltowski

Taylor & Francis
Taylor & Francis Group

Taylor series approximation of semi-blind BLUE channel estimates with applications to DTV

CHRISTOPHER PLADDY*†, SERDAR ÖZEN‡, S. M. NERAYANURU§,
PEILU DING§, MARK J. FIMOFF¶ and MICHAEL ZOLTOWSKI⊥

†MITRE Corp., Leavenworth, KS, USA
‡Izmir Institute of Technology, Turkey
§Motorola Corp., IL, USA
¶Techwell Corp., Palatine, IL, USA
⊥Purdue University, West Lafayette, IN, USA

We present a low-complexity method for approximating the semi-blind best linear unbiased estimate (BLUE) of a channel impulse response (CIR) vector for a communication system, which utilizes a periodically transmitted training sequence. The BLUE, for h, for the general linear model, $y = Ah + w + n$, where w is correlated noise (dependent on the CIR, h) and the vector n is an Additive White Gaussian Noise (AWGN) process, which is uncorrelated with w is given by $h = (A^T C(h)^{-1} A)^{-1} A^T C(h)^{-1} y$. In the present work, we propose a Taylor series approximation for the function $F(h) = (A^T C(h)^{-1} A)^{-1} A^T C(h)^{-1} y$. We describe the full Taylor formula for this function and describe algorithms using, first-, second-, and third-order approximations, respectively. The algorithms give better performance than correlation channel estimates and previous approximations used, at only a slight increase in complexity. Our algorithm is derived and works within the framework imposed by the *ATSC 8-VSB DTV* transmission system, but will generalize to any communication system utilizing a training sequence embedded within data.

Keywords: Semi-blind channel estimation; Best linear unbiased estimation; Taylor series approximation; Linearization; Gauss Markoff Theorem; Training-based channel estimation

AMS 2000 Mathematics Subject Classifications: 60G35; 93E10; 93E12; 93E24; 94A99

1. Introduction and notation

In digital communication systems, the medium that relays the signal (the atmosphere, telephone lines, optical cable, etc.) is called the channel and is characterized by a vector known as the channel impulse response (CIR). The received signal is given by the (discrete) convolution of the transmitted signal and the CIR vector. Accurate estimation of the channel impulse response is often required to facilitate reliable communication

---

*Corresponding author. Email: cpladdy@hotmail.com

using equalization of the channel. Equalization is the process of compensating for unwanted channel features, which produce interference between symbols (known as intersymbol interference or ISI) that is caused by distortion introduced by the finite bandwidth and nonideal characteristics of the filters and amplifiers within the system [1,2]. When ISI is present, the desired information is observed as a function of other information symbols, as well as the CIR.

The process of equalization involves determining filters, which are given by the solution of equations that depend on the CIR vector. Hence, an accurate estimate of the CIR vector is often essential for equalization of the received signal.

With this ultimate problem of the determination of equalizing filters for digital signals in mind, the inverse problem, which we consider is that of deconvolution of a CIR vector from transmitted data, using knowledge of the received data. As described above, this is a standard deconvolution problem of digital communication.

Traditional channel estimation techniques are based on using a training sequence: that is, a periodically sent known data sequence, which is also stored at the receiver. Most communication standards utilize a training sequence to estimate the channel (e.g., GSM). Training-based channel estimation suffers from various drawbacks particular to the specific communications systems under consideration. For instance in our case, for the 8-VSB modulation system (the North American standard for High-definition television (HDTV) over-the-airwaves transmisson) the training sequence is not sufficiently long in relation to the observed possible CIR vector lengths to form an overdetermined system for reliable least-squares solution. (The training sequence length is set in the standard and so cannot be altered.)

Blind channel estimation allows for estimation of the channel based on only the statistics of the received (unknown) signal, and does not use a known training sequence.

Semi-blind algorithms exploit information used by blind methods (for example, the statistics of the unknown data symbols) as well as information from known training symbols. In general, semi-blind methods make both training-based and blind methods more robust, and performance improvements are achieved. For general references on blind and semi-blind channel estimation, see [3]. Several recent papers consider different aspects of semi-blind channel estimation. Notably [4], considers identifiability conditions, and [5–8] consider different aspects of the semi-blind approach.

Our algorithm is designed to work within the constraints imposed by the ATSC digital TV 8-VSB modulation system [9]. This standard uses a training sequence of length 728 repeated every frame of 260,416 data symbols. Channel lengths of 400–500 symbols have been inferred from the delay spreads observed in urban multipath data captures. Such channels are too long to form an overdetermined system for an accurate least-squares estimate of the channel without extending the system of equations, so that we must consider correlated noise due to unknown data. Hence, we are led to the use of the statistics of the unknown data surrounding the training sequence. This consideration makes our algorithm a semi-blind algorithm: it uses both training data and statistics of unknown data.

We describe broadly the equalizer architecture that we use. Our channel estimation algorithm is used as a fast way to initialize our channel estimate, which in turn is used to compute tap weights for equalizer filters. We can use either a decision feedback equalizer (DFE) [1], or a predictive-decision feedback equalizer (pDFE) [1]. The pDFE uses a noise cancelling feedback filter and operates in the frequency domain, where fast computation of filter tap weights and fast filtering is possible due to use of the FFT to

transform the convolution operation in the time domain into point-wise multiplication in the frequency domain. For more details see [1,10–11]. After this initial computation of equalizer tap-weights from the CIR vector estimate, the tap-weights vary adaptively (via Least Mean Square or other optimization methods [12–14]) to equalize the (possibly time-varying) channel. The training sequence can be used each time it is transmitted to reinitialize the CIR estimate, if tracking of the equalizer tap-weights is failing due to some adverse conditions. In this articles we describe an algorithm to solve the CIR estimation problem that constitutes a single "block" in the broader problem of equalization. We do not discuss the specifics of equalization beyond this point, but merely mention the topic to give the context of our problem.

The physical channel may have changed significantly from the reception of one training burst to the arrival of the next, and so we require a channel estimation method that yields a good estimate based on a single burst of training data. To achieve this, our method considers an overdetermined system, which must account for correlated noise due to transmission data. Our algorithm allows for such accurate channel estimation, even in a mobile setting. In [15] and [16], we devised a semi-blind iterative algorithm to construct the best linear unbiased estimate (BLUE) of the channel, which is given by the Gauss–Markoff Theorem ([17–19]) as

$$h_{\text{BLUE}} = (A^T C(h)^{-1} A)^{-1} A^T C(h)^{-1} y \tag{1}$$

in the case where we have the general linear model for the received data, i.e., where the noise is not white, since we consider correlated noise due to unknown data, where, in our case, the noise is correlated due to convolution with the CIR.

We describe in detail, in the following sections, the channel model, channel estimation via the correlation method, and the general linear model that we use for channel estimation.

2. Baseband data transmission model

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

$$y[n] \equiv y(t)\big|_{t=nT} = \sum_k I_k h[n-k] + v[n], \tag{2}$$

$$= \sum_k I_k h[n-k] + \sum_k \eta[k] q^*[-n+k] \tag{3}$$

where $I_k \in \{\alpha_1, \ldots, \alpha_M\} \subset C$ is the M-ary complex valued training sequence. $\{a_k\}$ denotes the first N known (training) symbols within a frame of length N' and $\{d_k\}$ denotes the remaining $N'-N$ random data within the frame. $v(t) = \eta(t) * q^*(-t)$ denotes the complex (colored) noise process at the output of 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 dynamic channel, including the pulse shaping transmit filter $q(t)$, the physical channel impulse response $c(t)$, and the receive filter $q^*(-t)$ and is given by

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

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 an integer multiple of the symbol period, T. That is, $T_q = N_q T = 2L_q$, $N_q \in Z^+$. $\{c_k\} \subset C$ denote the complex valued physical channel gains and $\{\tau_k\}$ denote the multi-path delays or the time-of-arrivals (TOA). We also note that for the 8-VSB system, the transmitter pulse shape is the Hermitian symmetric root raised cosine pulse, which implies that $q(t) = q^*(-t)$. In the sequel, we will denote both the transmit and receive filters by $q[n] \equiv q(t)|_{t=nT}$. Also the sampled matched filter output signal, $y[n]$, will be used extensively in vector form, and so we introduce the notation $y_{[n_1:n_2]} = [y[n_1], \ldots, y[n_2]]^T \in R^{n_2-n_1+1}$. Similarly $\eta_{[n_1:n_2]} = [\eta[n_1], \ldots, \eta[n_2]]^T \in R^{n_2-n_1+1}$ and $v_{[n_1:n_2]} = [v[n_1], \ldots, v[n_2]]^T \in R^{n_2-n_1+1}$.

3. Channel estimation via correlation

Some of the material contained in sections 3 and 4 is also present in [16]. We include it here for completeness. Channel estimation via correlation is typically performed by correlating the received signal with a copy of the training sequence stored at the receiver. Cox [20] initially considered this problem for outdoor channels, while Devasirvatham [21] considered indoor channels and Parsons *et al.* [22] reduced the complexity of the algorithm for implementation. Correlation used for initial synchronization is considered in [23]. Other works that consider channel estimation via correlation are [24–26].

We denote the channel estimate obtained via correlation by

$$h_{\mathrm{corr}} = [h_{\mathrm{corr}}[-N_a], \ldots, h_{\mathrm{corr}}[-1], h_{\mathrm{corr}}[0], h_{\mathrm{corr}}[1], \ldots, h_{\mathrm{corr}}[N_c]]^T$$

given by

$$h_{\mathrm{corr}}[k] = \sum_{n=k}^{N-1+k} y[n]a_{n-k}$$

Collecting together $(N_a + N_c + 1)$ of these equations, in matrix vector form the correlation channel estimate is given by

$$h_{\mathrm{corr}} = \widehat{A} y_{[-N_a:N_c+N-1]} \tag{5}$$

where

$$\widehat{A} = T\left\{\left[a_0, 0_{N_a+N_c}^T\right]^T, \left[a_0, \ldots, a_{N-1}, 0_{N_a+N_c}^T\right]\right\}$$

$$= \begin{bmatrix} a_0 & a_1 & \ldots & a_{N-1} & 0 & \ldots & 0 \\ 0 & a_0 & a_1 & \ldots & a_{N-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ldots & \ddots & 0 \\ 0 & \ldots & 0 & a_0 & a_1 & \ldots & a_{N-1} \end{bmatrix}$$

is the $(N_a + N_c + 1) \times (N_a + N_c + N)$ Toeplitz convolution matrix with first column $[a_0, 0_{N_a+N_c}^T]^T$ and first row $[a_0, \ldots, a_{N-1}, 0_{N_a+N_c}^T]$.

The received signal, $y_{[-N_a:N_c+N-1]}$, is given by

$$y_{[-N_a:N_c+N-1]} = Xh + v_{[-N_a:N_c+N-1]}$$

where $v_{[-N_a+N_c+N-1]} = \widehat{Q}\eta_{[-N_a-L_q:N_c+N-1-L_q]}$, where $\eta_{[-N_a-L_q:N_c+N-1-L_q]}$ is additive white Gaussian noise and $\widehat{Q} \in R^{(N_a+N_c+N)\times(N_a+N_c+N+2L_q)}$ is given by

$$\widehat{Q} = \begin{bmatrix} q^T & 0 & \cdots & 0 \\ 0 & q^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q^T \end{bmatrix}$$

and

$$q^T = [q[+L_q], \ldots, q[0], \ldots q[-L_q]] \in R^{2L_q+1}$$

where q denotes the symbol rate sampled receiver pulse matched filter. The matrix X is given by

$$X = T\left\{x_{[0:N_a+N_c+N-1]}, x_{[0:-N_c-N_a]}^T\right\}$$

the $(N_a+N_c+N) \times (N_a+N_c+1)$ Toeplitz matrix with first column $x_{[0:N_a+N_c+N-1]}$ and first row $x_{[0:-N_c-N_a]}^T$. The values of x_n are given by

$$x_n = \begin{cases} a_n & \text{for all } 0 \le n \le N-1 \\ d_n & \text{for all other } n \end{cases}.$$

Hence

$$X = \begin{bmatrix} a_0 & & d_{-1} & \cdots & d_{-N_c-N_a} \\ a_1 & & a_0 & \ddots & \vdots \\ \vdots & & a_1 & \ddots & d_{-1} \\ a_{N-1} & & \vdots & \ddots & a_0 \\ d_N & & a_{N-1} & \ddots & a_1 \\ \vdots & & & \ddots & \ddots & \vdots \\ d_{N_a+N_c+N-1} & \cdots & & d_N & a_{N-1} \end{bmatrix}$$
$$= (\widehat{A})^T + (\widehat{D})^T$$

where

$$(\widehat{D})^T = \begin{bmatrix} 0 & & d_{-1} & \cdots & d_{-N_c-N_a} \\ 0 & & 0 & \ddots & \vdots \\ \vdots & & 0 & \ddots & d_{-1} \\ 0 & & \vdots & \ddots & 0 \\ d_N & & 0 & \ddots & 0 \\ \vdots & & & \ddots & \ddots & \vdots \\ d_{N_a+N_c+N-1} & \cdots & & d_N & 0 \end{bmatrix}$$

Hence, using (3), we have,

$$\begin{aligned} h_{\text{corr}} &= \widehat{A} y_{[-N_a:N_c+N-1]} \\ &= \widehat{A}(\widehat{A})^T h + \widehat{A}(\widehat{D})^T h + \widehat{A} v_{[-N_a:N_c+N-1]} \end{aligned} \tag{6}$$

If the training sequence has been sent in isolation of transmission data (purely a sounding sequence), then the term $\widehat{A}(\widehat{D})^T h$ does not occur. In our case, the training sequence is embedded in transmission data and this term does appear, degrading the correlation estimate from the case where a sounding sequence alone is used as training. The general idea of the correlation estimate is that $\widehat{A}(\widehat{A})^T$ is very close to the identity matrix, due to the pseudonoise properties of the training sequence, and so we see that the correlation estimate approximates the actual channel with the addition of the noise term, $\widehat{A} v_{[-N_a:N_c+N-1]}$. The situation is worse for the case of a training sequence embedded in transmission data due to the term $\widehat{A}(\widehat{D})^T h$.

4. Matrix–vector formulation of the transmisson model

4.1. *Formulation of the least squares channel estimation problem*

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

$$h = [h[-N_a], \dots, h[-1], h[0], h[1], \dots, h[N_c]]^T$$

where N_a and N_c denote the number of anticausal and causal taps respectively. Based on (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, in vector notation, as,

$$\begin{aligned} y_{[N_c:N-N_a-1]} &= \widetilde{A} h + v_{[N_c:N-N_a-1]} \\ &= \widetilde{A} h + \widetilde{Q} \eta_{[N_c-L_q:N-N_a-1+L_q]} \end{aligned}$$

where $\widetilde{A} = T\{[a_{N_n}+N_a, \dots, a_{N-1}]^T, [a_{N_n+N_a}, \dots, a_0]\}$ is the $(N - N_a - N_c) \times (N_a + N_c + 1)$ Toeplitz convolution matrix with first column $[a_{N_n+N_a}, \dots, a_{N-1}]^T$ and first row $[a_{N_n+N_a}, \dots, a_0]$,

$$\widetilde{A} = \begin{bmatrix} a_{N_c+N_a} & a_{N_c+N_a-1} & \cdots & a_0 \\ a_{N_c+N_a+1} & a_{N_c+N_a} & \cdots & a_1 \\ \vdots & \vdots & \ddots & \vdots \\ a_{N-1} & a_{N-2} & \cdots & a_{N-1-N_c-N_a} \end{bmatrix}$$

and $v_{[N_c:N-N_a-1]} = \widetilde{Q} \eta_{[N_c-L_q:N-N_a-1+L_q]}$ is the colored noise at the receiver matched filter output, where $\widetilde{Q} \in R^{(N-N_a-N_c) \times (N-N_a-N_c+2L_q)}$ is given by

$$\widetilde{Q} = \begin{bmatrix} q^T & 0 & \cdots & 0 \\ 0 & q^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q^T \end{bmatrix}$$

with q the symbol rate sampled receiver pulse matched filter, $q^T = [q[+L_q], \ldots, q[0], \ldots, q[-L_q]] \in R^{2L_q+1}$. The problem which we encounter with this approach is that the system matrix, $\tilde{A}$, is $(N - N_a - N_c) \times (N_a + N_c + 1)$ dimensional. In our case the parameter values are such that, $N_a + N_c + 1$, which is the length of the channel is in excess of 500 tap values. This value is determined empirically from experimental data captures in the field. We have previously used the value $N_a + N_c + 1 = 576$, and we shall briefly consider this value here. The parameter N is the training sequence length, and this is fixed in the 8-*VSB* standard as 704. Hence the system is 129×576 dimensional, and is clearly badly underdefined. Simulations using this system give very bad least-squares solutions for the channel vector, h.

With the above reasoning in mind, we attempt to somehow extend the size of the linear system to give an overdetermined system.

4.2. *Formulation of the overdetermined LS channel estimation problem*

To this end, and in a similar manner to the development above, we can write the pulse matched filter output that includes *all the contributions from the known training symbols* (including output which includes contributions from adjacent unknown random data) as

$$
\begin{aligned}
y_{[-N_a:N+N_c-1]} &= (A + D)h + v_{[-N_a:N+N_c-1]} \\
&= Ah + Dh + Q\eta_{[-N_a-L_q:N+N_c-1+L_q]}
\end{aligned}
\tag{7}
$$

where

$$
A = T\{[a_0, \ldots, a_{N-1}, \underbrace{0, \ldots, 0}_{N_a+N_c}]^T, [a_0, \underbrace{0, \ldots, 0}_{N_a+N_c}]\}
$$

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

$$
D = T\{[\underbrace{0, \ldots, 0}_{N}, d_N, \ldots, d_{N_a+N_c+N-1}]^T, [0, \underbrace{d_{-1}, \ldots, d_{-N_a-N_c}}_{\text{Previous frame data}}]\}
$$

is a $(N + N_a + N_c) \times (N_a + N_c + 1)$ Toeplitz matrix which includes adjacent random infrormation symbols only, both prior to the training sequence and after the training sequence. That is

$$
A = \begin{bmatrix}
a_0 & 0 & \cdots & 0 \\
a_1 & a_0 & \ddots & \vdots \\
\vdots & a_1 & \ddots & 0 \\
a_{N-1} & \vdots & \ddots & a_0 \\
0 & a_{N-1} & \ddots & a_1 \\
\vdots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & a_{N-1}
\end{bmatrix}
$$

and

$$
D = \begin{bmatrix}
0 & d_{-1} & \cdots & d_{-N_c-N_a} \\
0 & 0 & \ddots & \vdots \\
\vdots & 0 & \ddots & d_{-1} \\
0 & \vdots & \ddots & 0 \\
d_N & 0 & \ddots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
d_{N_a+N_c+N-1} & \cdots & d_N & 0
\end{bmatrix}
$$

We shall only use the statistics of this random data, since the actual values are unknown. We use these statistics to compute the covariance matrix of the correlated noise, which is used in the solution for the BLUE estimate of h. The data symbols $d_{-1}, \ldots, d_{-N_a-N_c}$ denote the unknown information symbols transmitted at the end of the previous frame. The data symbols $d_N, \ldots, d_{N_a+N_c+n-1}$ denote the unknown information symbols transmitted after the training sequence in the present frame. The colored noise at the receiver matched filter output is $v_{[-N_a:N+N_c-1]} = Q\eta_{[-N_a-L_q:N+N_c-1+L_q]}$, where $Q \in R^{(N+N_a+N_c)\times(N+N_a+N_c+N_q)}$ is defined similarly to $\tilde{Q}$. In fact $Q \in R^{(N+N_a+N_c)\times(N+N_a+N_c+N_q)}$ is given by

$$
Q = \begin{bmatrix}
q^T & 0 & \cdots & 0 \\
0 & q^T & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & q^T
\end{bmatrix}
\tag{8}
$$

The dimensions of the system so formed are $(N + N_a + N_c) \times (N_a + N_c + 1)$ and for our values of $N = 704$, $N_a + N_c + 1 = 576$, we have a 1280×576 dimensional system that is overdetermined and suitable for solution by the method of least-squares. One complication to this approach is that the noise is no longer white. We include unknown random data into the noise term, and this gives a correlated component to the noise, where the data has been correlated by transmission through the channel. That is, the filter which correlates the independent identically distributed data as it passes through the channel is exactly the channel impulse response that we are trying to find. The AWGN term is slightly correlated due to filtering by the receiver raised cosine filter. To give the solution of this general linear problem, we will need to compute the covariance matrix of the total noise contribution.

4.3. *Covariance matrix computation*

To compute the covariance matrix for the noise contribution $Dh + Q\eta_{[-N_a-L_q:N+N_c-1+L_q]}$, to the received vector $y_{[-N_a:N+N_c-1]}$, it is advantageous to rewrite the term Dh.

We define the vector $d \in R^{N+2(N_c+N_a)}$ by

$$d = [d_{-N_c-N_a}, \ldots, d_{-1}, 0_{1 \times N}, d_N, \ldots, d_{N+N_c+N_a-1}]^T$$

and the Toeplitz channel convolution matrix, H, where

$$H = H(h) = T\{[h[-N_a], 0, \ldots, 0], [\bar{h}^T, 0, \ldots, 0]^T\}$$
$$\in R^{(N+N_a+N_c) \times (N+2(N_c+N_a))}$$

h^T is the time-reversal of the channel vector,

$$\bar{h}^T = [h[N_c], \ldots, h[1], h[0], h[-1], \ldots, h[-N_a]] \in R^{N_c+N_a+1}.$$

Specifically,

$$H = \begin{bmatrix} \bar{h}^T & 0 & \ldots & 0 \\ 0 & \bar{h}^T & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \bar{h}^T \end{bmatrix} \tag{9}$$

It is easy to verify that $Dh = Hd$ and $E[Hd(Hd)^T] = \sigma_d^2 HSH^T$, where $E[dd^T] = \sigma_d^2 S$ and the selection matrix $S \in R^{(N+2(N_c+N_a)) \times (N+2(N_c+N_a))}$ is given by

$$S = \begin{bmatrix} I_{(N_c+N_a)} & 0_{(N_c+N_a) \times N} & 0_{(N_c+N_a)} \\ 0_{N \times (N_c+N_a)} & 0_N & 0_{N \times (N_c+N_a)} \\ 0_{(N_c+N_a)} & 0_{(N_c+N_a) \times N} & I_{(N_c+N_a)} \end{bmatrix} \tag{10}$$

Since $Dh = Hd$, then we may rewrite (7) as

$$y_{[-N_a:N+N_c-1]} = Ah + Hd + Q\eta_{[-N_a-L_q:N+N_c-1+L_q]} \tag{11}$$

Here the total noise term, which combines the contribution of unknown data convolved with the composite channel, Hd, with the AWGN filtered by the receive filter, $Q\eta_{[-N_a-L_q:N+N_c-1+L_q]}$, has covariance matrix C, given by

$$\text{Cov}(w) = \frac{1}{2} E[ww^H]$$
$$= \frac{\sigma_d^2}{2} HSH^H + \frac{\sigma_\eta^2}{2} QQ^H$$
$$= \frac{\sigma_d^2}{2} \left(HSH^H + \frac{\sigma_\eta^2}{\sigma_d^2} QQ^H \right)$$

since the data symbols, $d[.]$ and the AWGN term $\eta[.]$ are uncorrelated. Using the fact that due to the choice of the transmitter and receiver pulse shaping, the product QQ^H is very nearly equal to $\|q\|^2 I$, and $\sigma_v^2 = \sigma_\eta^2 \|q\|^2$, then we can give the further approximation

$$\text{Cov}(w) \simeq \frac{\sigma_d^2}{2} \left(HSH^H + \frac{\sigma_v^2}{\sigma_d^2} I \right)$$

where I is the $(N + N_a + N_c) \times (N + N_a + N_c)$ identity matrix. This approximation is often useful in simulations. Since the vector q is the root raised cosine function sampled at zero and its zero crossings, then the matrix product QQ^H is a diagonal matrix with diagonal entries $\|q\|^2$ and with off-diagonal entries, which are the values of the autocorrelation function of q evaluated at n, $n \neq 0$. These values are theoretically zero, but due to the finite length of the vector q, may actually turn out to be small but nonzero. This gives rise to the product being close to the identity, but not exactly equal to it.

We note that since $H = H(h)$ from (9), then $\text{Cov}(w)$ is a function of h, i.e., $\text{Cov}(w)$ is a function of the channel impulse response vector that we are attempting to estimate. We shall define

$$C(h) = HSH^H + \frac{\sigma_\eta^2}{\sigma_d^2} QQ^H \tag{12}$$

so that $\text{Cov}(w) = (\sigma_d^2/2)C(h)$. We often use the approximation

$$C(h) \simeq HSH^H + \frac{\sigma_v^2}{\sigma_d^2} I$$

in simulations. Using the approximation $QQ^H \simeq \|q\|^2 I$ in practice is a judgement call as to whether to added precision but higher complexity of using the exact value QQ^H is preferable to use the approximation $\|q\|^2 I$, which offers a reduction in complexity, but at a loss in precision. For hardware implementations, the approximation might likely be used.

5. Problem statement

5.1. *Problem description*

The solution of the general linear model, (11), is given by the Gauss–Markoff Theorem, [17–19], as

$$h = (A^T(\text{Cov}(w))^{-1}A)^{-1}A^T(\text{Cov}(w))^{-1}y \tag{13}$$

It is straightforward to check that the factor $(\sigma_d^2/2)$ contained in the expression for $\text{Cov}(w)$ actually cancels from (13), and we obtain

$$h = (A^T C(h)^{-1}A)^{-1}A^T C(h)^{-1}y \tag{14}$$

where $C(h)$ is given by (12). We note that in solving (14) we are looking for a fixed point, denoted by h_0, of the mapping $h \mapsto F(h)$, where for each fixed vector of received symbols, y, and each fixed convolution matrix of known transmitted training symbols, A, then $F: R^L \to R^L$ is given by (15).

5.2. *Previous work*

Our own previous approaches to finding the semi-blind BLUE channel estimate have encompassed:

Case 1 In [15] and [16], an initial thresholded approximation denoted $h^{(0)}$ to h was obtained via correlation of the training sequence with a stored copy of the training

sequence at the receiver. (This processing already exists for timing acquisition.) Then the iteration

$$h^{(k+1)} = (A^T C(h^{(k)})^{-1} A)^{-1} A^T C(h^{(k)})^{-1} y, \quad k = 0, 1, 2, \dots$$

was used to generate a sequence of approximations to the fixed point, h_{BLUE}. Numerical simulations indicated that for the problems which we were concerned with of channel identification for HDTV transmission, two or three iterations were sufficient for an error of $\|h^{(k)} - h_{\text{BLUE}}\| \sim 10^{-6}$ (here $h \in R^{512}$). Theoretically, convergence of this iteration to the unique fixed point, h_{BLUE}, of the function $F(h) = F_{y,A}(h)$ is guaranteed if $\|J_F(h)\| < 1$, where $J_F(h)$ denotes the Jacobian matrix of F [16]. This condition can be satisifed if the initial approximation is sufficiently close to the fixed point.

Case 2 In [27], an approximate linear system was derived to give an approximate solution to (14). This approximate linear system was derived by replacing $C(h)$ by $C(h_{\text{ideal}})$, on the right-hand side of (14) where $h_i = [0, \dots, 0, 1, 0 \dots, 0]^T \in R^L$, here the 1 appears in the 64th position in our case, to correspond to the position of the cursor in the decision feedback equalizer (DFE) that we use. This approach has the added advantage that the matrix $(A^T C(h_{\text{ideal}})^{-1} A)^{-1} A^T C(h_{\text{ideal}})^{-1}$ may be computed offline and stored at the receiver.

5.3. *Present work*

In the present work, we take the point of view that we may expand the vector-valued function $F(h) = F_{y,A}(h) = (A^T C(h)^{-1} A)^{-1} A^T C(h)^{-1} y$ of the vector variable, h, using Taylor's formula [28,29] about a fixed vector, h_{ideal}. The approximation to $F_{y,A}(h)$ given by $F_{y,A}(h_{\text{ideal}})$ in Case 2, above, is the zero-order Taylor approximation. We then derive explicitly first-, second-, and third-order Taylor approximations to the BLUE CIR estimate. The methods are attractive, since computation of the matrices involved may be done offline and stored at the receiver. The only processing, which will be needed at the reciever is that of computing $(h_i - h_i^{id})$, where h_i is not available, but is estimated as the *i*th entry of the approximate channel vector obtained by correlation, and $h_i^{id} = 1$, if we use the "ideal" vector $h_{id} = [0, \dots, 0, \underbrace{1}_{i\text{th}}, 0, \dots, 0]^T$, and then to form the matrices used in obtaining the vectors given in (17) and (18).

5.4. *Other semi-blind channel estimation approaches*

Some other approaches to the problem of semi-blind channel estimation have encompassed [5–8].

Medles *et al.* [5] uses a linear prediction for the received symbols, combined with the observation (measurement) equation that is available. The linear prediction of the received symbols gives the blind criterion, and a least-squares problem constitutes the training-based part of the approach.

In [7], a semi-blind algorithm is used which is the regularized version of a least-squares solution. Regularization (see, for example, [30,31]) is typically used as a means to better condition ill-conditioned problems. In [7], the penalty added to the least-squares cost function which constitutes the regularization term uses *a priori* information about the channel. It is essentially a Bayesian approach, more

accurately termed an "empirical Bayesian" approach by the authors as it fits the parameters of the prior distribution using the empirical data itself. If our approach could be adapted to be a Bayesian approach, this would add extra complexity to the solution, as an additional term, C_h^{-1}, the inverse of the covariance matrix of the CIR vector, which is now viewed as a random variable would need to be computed to give the Bayesian analoge to the least-squares solution [31],

$$h_{\text{Bayes}} = (A^T C(h)^{-1} A + C_h^{-1})^{-1} A^T C(h)^{-1} y$$

In [8], a novel use of the subchannel response matching criterion is used [32]. This technique uses "cross relations" between subchannel output pairs.

In [6] the approach given in [7] of linearly combining the training-based criterion and the blind criterion, in the manner which is identical to the standard regularization method, is extended. The authors comment that in [7] an asymptotic argument was used to "fine tune" the regularization parameter by minimizing the channel estimation error. This is done in [7] for the single-input multiple-output (SIMO) case. In [6] the results are extended to the case of the Multiple-input multiple-output channel in the context of a synchronized Code Division Multiple Access uplink (channel estimation for the multiple channels in this case). Since, in this case, there are several channels to be estimated in the uplink, then there is a vector of regularization parameters (or tuning parameters) to be estimated. Lasaulce *et al.* [6] addresses this problem.

Our algorithm is a semi-blind algorithm, using training data and transmission data statistics, although it does not use a "regularization approach" which uses a linear combination of training and blind cost functions, nor does it use a linear prediction for the channel impulse response, nor do we use a subspace matching approach. Our algorithm is a novel semi-blind approach, which uses an appropriately weighted least-squares solution of an overdetermined least-squares system. This system has been extended to be overdetermined and the solution incorporates knowledge of the statistics of unknown transmission data.

6. Overview of Taylor series and approximations

For each fixed matrix of training data, A, and each vector of received values, y, we define the function

$$F(h) = F_{y,A}(h) = (A^T C(h)^{-1} A)^{-1} A^T C(h)^{-1} y \tag{15}$$

In [27], an approximate version of the iterative algorithm of [15] and [16] is described. In the present work, we propose a more general framework within which the approximation given in [27] would be the zero-order Taylor series approximation of the function $F(h)$, i.e., a constant approximation, $F(h_{id})$, to the function $F(h)$.

We consider the real part of the received data, denoted as vector y, and the real part of the CIR vector, denoted as h. We use the vector-valued function of a real vector variable version of the Taylor series [28,29], to develop a series approximation to the function $F(h)$ that gives the BLUE estimate for h, as given in (1).

Note that although our proposed algorithm does not fit within a Kalman filtering framework, the process of linearization used in deriving the extended Kalman filter

(EKF) is similar to that which we propose here, and the higher order approximations that we use are similar to those used in obtaining higher order Kalman filter approximations [33].

In fact, the use of the Taylor series is a standard tool of approximation, see, for example, [34–36] to see instances of its use in applied engineering problems.

We describe the full Taylor formula for this function,

$$F(h) = F(h_{id}) + \sum_{|\alpha| \geq 1} (h - h_{id})^\alpha (\partial/\partial h)^\alpha F(h_{id}) \tag{16}$$

(where $h = [h_1, \ldots, h_L]^T$, $h_{id} = [h_1^{id}, \ldots, h_L^{id}]^T$ and α is a multi-index. The multi-index notation is defined by

$$h^\alpha = \prod_{i=1}^{L} h_i^{\alpha_i}$$

where $\alpha = [\alpha_1, \ldots, \alpha_L]^T \in R^L$. $|\alpha|$ denotes the $1 - norm$ of α, that is $|\alpha| = \sum_{i=1}^{L} |\alpha_i|$ and the derivative $(\partial/\partial h)^\alpha F(h_{id})$ is defined by

$$(\partial/\partial h)^\alpha F(h_{id}) = \left(\prod_{i=1}^{L} (\partial/\partial h)^{\alpha_i} \right) F(h) \Bigg|_{h=h_{id}}$$

We shall use either a first (denoted $\nabla^1 F(h_{id})$) or second (denoted $\nabla^2 F(h_{id})$) or third (denoted $\nabla^3 F(h_{id})$) order approximation. We do not describe the third-order approximation explicitly in the interests of space. The first- and second-order approximations are most easily implementable in practice, considering the tradeoff between complexity and performance. We have,

$$F(h) \approx \nabla^1 F(h_{id}) \equiv F(h_{id}) + (h_i - h_i^{id})(\partial/\partial h_i)F(h_{id}) \tag{17}$$

$$F(h) \approx \nabla^2 F(h_{id}) \equiv F(h_{id}) + (h_i - h_i^{id})(\partial/\partial h_i)F(h_{id}) + (h_i - h_i^{id})^2 (\partial^2/\partial h_i^2)F(h_{id}) \tag{18}$$

Note that the *full* linear approximation contains L first-order derivative terms. That is

$$F(h) \approx F(h_{id}) + \sum_{i=1}^{L} (h_i - h_i^{id})(\partial/\partial h_i)F(h_{id})$$

Similarly, the full second-order derivative approximation contains L first-order derivative terms and $(L^2 + L)/2$ second-order derivative terms. That is

$$F(h) \approx F(h_{id}) + \sum_{i=1}^{L} (h_i - h_i^{id})(\partial/\partial h_i)F(h_{id}) + \sum_{|\alpha|=2} (h_i - h_i^{id})^\alpha (\partial/\partial h_i)^\alpha F(h_{id})$$

We emphasize that the first- and second-order approximations (FOA and SOA, respectively) that we use in our algorithms are *not* the full FOA and SOA as described here, but merely a first-order approximation consisting of a single first-order derivative with respect to the main tap weight and similarly a SOA, which contains only a single first-order derivative and a single second-order derivative each with respect to the main tap weight. The main tap weight will be described later, but it denotes the position of

the main signal in the CIR vector. This insight is based partly on the desire to reduce the complexity of the algorithms (by using fewer terms) and also on the empirical observation that the performance of the algorithms improves when only a *single dominant tap-weight* is used in the expansions. The main tap weight of the CIR vector is identified in practice from the correlation estimate. The processing necessary to compute the correlation estimate already exists for the purposes of timing synchronization in the system, and so no extra complexity is added by using this.

7. Derivation of the linear and quadratic approximations

We define the matrix

$$A_C(h) = \left(A^T C(h)^{-1} A\right)^{-1} A^T C(h)^{-1} \in R^{L \times (N+L-1)}. \tag{19}$$

Then $F\colon R^L \to R^L$ defined by $F\colon h \mapsto F(h)$ as defined in (15) is given by

$$F(h) = A_C(h)y \in R^L.$$

We state the following Propositions, and give brief proofs.

PROPOSITION 1 *For any matrix $B = B(h) \in R^{n \times n}$, which depends on a vector parameter $h = [h_1, \ldots, h_K] \in R^K$ such that there exists an open set $U \subset R^K$, such that B is nonsingular and differentiable on U, then, we have that for any i, $1 \leq i \leq K$, and for any $h \in U$,*

$$\frac{\partial B(h)^{-1}}{\partial h_i} = -B(h)^{-1} \frac{\partial B(h)}{\partial h_i} B(h)^{-1} \tag{20}$$

Proof Differentiate with respect to h_i both sides of the identity $B(h)^{-1} B(h) = I$, using the product rule. ∎

Concerning $C(h)$ we have,

PROPOSITION 2 *$C(h)$ is a differentiable function of h for all $h \in R^{N_a+N_c+1}$. $C(h)$ is a positive definite matrix for all $h \in R^{N_a+N_c+1}$, and consequently $C(h)$ is invertible for all $h \in R^{N_a+N_c+1}$.*

Proof The fact that $C(h)$ is a differentiable function of h for all $h \in R^{N_a+N_c+1}$ is clear from the definition (12). $C(h)$ is a positive definite matrix for all $h \in R^{N_a+N_c+1}$ since for all $h \in R^{N_a+N_c+1}$ and for all nonzero $x \in R^{N_a+N_c+N}$ we have

$$\langle C(h)x, x \rangle = \langle HSH^H x, x \rangle + \frac{\sigma_\eta^2}{\sigma_d^2} \langle QQ^H x, x \rangle$$

$$= \|SH^H x\|^2 + \frac{\sigma_\eta^2}{\sigma_d^2} \|Q^H x\|^2$$

$$\geq \frac{\sigma_\eta^2}{\sigma_d^2} \|Q^H x\|^2$$

$$> 0 \quad \text{for all } x \in R^{N_a+N_c+N}, x \neq 0$$

follows since Q^H is of full column rank, and provided that $\sigma_\eta^2 \neq 0$. We have used the fact that $S = S^T S$. Hence, $C(h)$ is a positive definite matrix for all $h \in R^{N_a+N_c+1}$ and consequently is nonsingular for all $h \in R^{N_a+N_c+1}$. $C^{-1}(h)$ is differentiable for all $h \in R^{N_a+N_c+1}$ follows from Lemma 3. ∎

Concerning differentiability of F we have

PROPOSITION 3 *The function F given by (15) is differentiable for all $h \in R^{N_a+N_c+1}$.*

Proof Since $C(h)$ is nonsingular and differentiable for all $h \in R^{N_a+N_c+1}$, then $A^t C(h)^{-1} A$ is nonsingular and differentiable for all $h \in R^{N_a+N_c+1}$, and so $(A^t C(h)^{-1} A)^{-1}$ is differentiable by Proposition 1. Hence, F is a product of differentiable functions and is differentiable for all $h \in R^{N_a+N_c+1}$. ∎

The partial derivatives of the function F are given by

PROPOSITION 4 *For $1 \leq i \leq L$, we have*

$$\frac{\partial F(h)}{\partial h_i} = A_C(h) \frac{\partial C(h)}{\partial h_i} C(h)^{-1} \{AA_C(h) - I\} y \tag{21}$$

Proof Differentiate, using the product rule, with respect to h_i, the expression for $Ac(h)y$ and use Proposition 1 three times. The details are given in the Appendix. ∎

PROPOSITION 5 *For $1 \leq i, j \leq L$, we have*

$$\frac{\partial^2 F(h)}{\partial h_i \partial h_j} = \left\{ \frac{\partial A_C(h)}{\partial h_j} \frac{\partial C(h)}{\partial h_i} + A_C(h) \left[\frac{\partial^2 C(h)}{\partial h_i \partial h_j} - \frac{\partial C(h)}{\partial h_i} C(h)^{-1} \frac{\partial C(h)}{\partial h_j} \right] \right\}$$
$$\times C(h)^{-1} \{AA_C(h) - I\} y + A_C(h) \frac{\partial C(h)}{\partial h_i} C(h)^{-1} \left\{ A \frac{\partial A_C(h)}{\partial h_j} \right\} y \tag{22}$$

Proof Differentiate (21) with respect to h_j, using the product rule and employing Proposition 1. We omit the details. ∎

We omit the expression for the third derivatives, but it is straightforward to compute from (22) above.

PROPOSITION 6 *For $1 \leq i \leq L$, we have*

$$\frac{\partial C(h)}{\partial h_i} = \sigma_d^2 H(h) S \left(\frac{\partial H(h)^T}{\partial h_i} \right) + \sigma_d^2 \left(\frac{\partial H(h)}{\partial h_i} \right) S H(h)^T \tag{23}$$

Proof Differentiating (12) with rect to h_i gives the required result. ∎

PROPOSITION 7 *For $1 \leq i, j \leq L$ we have*

$$\frac{\partial^2 C(h)}{\partial h_i \partial h_j} = \sigma_d^2 \left(\frac{\partial H(h)}{\partial h_j} \right) S \left(\frac{\partial H(h)^T}{\partial h_i} \right) + \sigma_d^2 \left(\frac{\partial H(h)}{\partial h_i} \right) S \left(\frac{\partial H(h)^T}{\partial h_j} \right) \tag{24}$$

Proof Differentiating (23) with rect to h_j, and noting that

$$\partial^2 H(h)^T / \partial h_i \partial h_j = 0$$

for any i, j, gives the required result. ∎

PROPOSITION 8   *For all derivatives of order* 3 *and higher, we have*

$$\left(\frac{\partial}{\partial h}\right)^{\alpha} C(h) \equiv 0_{(N+L-1)\times(N+L-1)} \quad \forall \alpha, |\alpha| \geq 3$$

Proof   Differentiating (24) with repect to h_k, and noting that

$$\frac{\partial^2 H(h)^T}{\partial h_i \partial h_j} = 0$$

for any *i, j*, gives the required result.   ∎

We use approximations (17) and (18) where we do *not use a full linear approximation*, including all terms in the linear Taylor series approximation. As commented before, we only use a linear approximation using the dominant term, which for us is a linear approximation where we have differentiated with respect to the *main tap weight*. A similar comment holds for both the quadratic and cubic approximations: we only use a single second-order term and a single third-order term in the SOA and third-order approximations.

8. Algorithm

8.1. *Input*

Received vector *y*;
   Training sequence to form data matrix *A*;
   Identify the position of the main tap weight of the CIR vector from the correlation estimate (processing exists to compute the correlation estimate for timing synchronization).

8.2. *Stored at receiver*

Zero order approximation (ZOA) matrix (19)

$$A_C(h_{id}) = (A^T C(h_{id})^{-1} A)^{-1} A^T C(h_{id})^{-1} \in R^{L\times(N+L-1)} \tag{25}$$

First-order approximation matrix $\in R^{L\times(N+L-1)}$ for first-order algorithm (see (21) and note that $\partial F(h)/\partial h_i = ((\partial A_C(h_{id})/\partial h_i)y)$

$$\frac{\partial A_C(h_{id})}{\partial h_i} = A_C(h_{id})\frac{\partial C(h_{id})}{\partial h_i} C(h_{id})^{-1}\{AA_C(h_{id}) - I\} \tag{26}$$

Second-order approximation matrix $\in R^{L\times(N+L-1)}$, if second-order algorithm is used (see (22) and note that $(\partial^2 F(h)/\partial h_i \partial h_j) = ((\partial^2 A_C(h)/\partial h_i \partial h_j)y)$

$$\frac{\partial^2 A_C(h_{id})}{\partial h_i^2} = \left\{\frac{\partial A_C(h_{id})}{\partial h_i}\frac{\partial C(h_{id})}{\partial h_i} + A_C(h_{id})\left[\frac{\partial^2 C(h_{id})}{\partial h_i^2} - \frac{\partial C(h_{id})}{\partial h_i}C(h_{id})^{-1}\frac{\partial C(h_{id})}{\partial h_i}\right]\right\}$$
$$\times C(h_{id})^{-1}\{AA_C(h_{id}) - I\} + A_C(h_{id})\frac{\partial C(h_{id})}{\partial h_i}C(h_{id})^{-1}\left\{A\frac{\partial A_C(h_{id})}{\partial h_i}\right\}$$
$$\in R^{L\times(N+L-1)} \tag{27}$$

Third-order approximation matrix if third-order algorithm is used (not given here).

8.3. *Real-time processing*

Let $h_{\text{corr}} = [h_1^{\text{corr}}, h_2^{\text{corr}}, \ldots h_L^{\text{corr}}]^T \in R^L$ denote the correlation approximation to the channel which is available from timing acquisition. Then compute the scalar value $h_i^{\text{corr}} - h_i^{id} = h_i^{\text{corr}} - 1$ when the ideal channel used is $h_{id} = [0, \ldots, 0, \underbrace{1}_{i\text{th}^{\text{position}}}, 0, \ldots, 0]^T \in R^L$. Use this, together with (25) and (26), to form the matrix

$$(\nabla^1 A_C)(h_{id}) = A_C(h_{id}) + (h_i^{\text{corr}} - h_i^{id})\frac{\partial A_C(h_{id})}{\partial h_i} \tag{28}$$

to be used in the first-order algorithm. (Complexity is $L \times (N+L-1)$ multiplications and $L \times (N+L-1)$ additions.) Similarly, using (25), (26) and (27), form the matrix

$$(\nabla^1 A_C)(h_{id}) = A_C(h_{id}) + (h_i^{\text{corr}} - h_i^{id})\frac{\partial A_C(h_{id})}{\partial h_i} + (h_i^{\text{corr}} - h_i^{id})^2 \frac{\partial^2 A_C(h_{id})}{\partial h_i^2} \tag{29}$$

(Complexity is $2 \times L \times (N+L-1)$ multiplications and $2 \times L \times (N+L-1)$ additions.) We omit the third-order approximation matrix in the interests of space.

Compute the approximations to h_{BLUE}

$$\nabla^0 F(h_{id}) = (\nabla^0 A_C(h_{id}))y \quad \text{ZOA,} \quad \text{identical to that in [27]}$$

$$\nabla^1 F(h_{id}) = (\nabla^1 A_C(h_{id}))y \quad \text{FOA using (28)}$$

$$\nabla^2 F(h_{id}) = (\nabla^2 A_C(h_{id}))y \quad \text{SOA using (29)}$$

$$\nabla^3 F(h_{id}) = (\nabla^3 A_C(h_{id}))y \quad \text{third-order approximation,}$$

$$\text{not explicitly given here}$$

(Each takes $(N + N_a + N_c) \times (N_a + N_c + 1)$ multiplications.)

9. Simulation results

The channels used are known as Brazil channels 2, 3, 5, 7, 8, and 11. They appear in the HDTV literature [9], and originate in field test data gathered in Brazil. All of the channels are pictured in figure 1. Channels 7 and 8 are described fully in table 1, giving delays in symbols and the relative path gains. In the interests of space, we only describe channels 7 and 8 in this manner.

In tables 2 and 3, we show results for the channels 2, 3, 7, 8, and 11 at 18 and 28 dB and contrast the results of our approximations with the correlation estimate. We use simulated 8-VSB data passed through the channels at 18 dB and 28 dB AWGN level. The table gives the 2-norm of the error, $\|h - \hat{h}\|$ where $\hat{h}$ is the channel estimate and h is the true channel. The error in the correlation estimate is given in the last column, labeled as "Correlation". The errors in the ZOA, FOA and SOA, are given in the previous columns. The ZOA is the same as the approximation considered in [27]. There is often improvement of the FOA, and SOA over the ZOA and over the correlation estimate. This is particularly true for the more "challenging" channels, 7 and 8 and to a
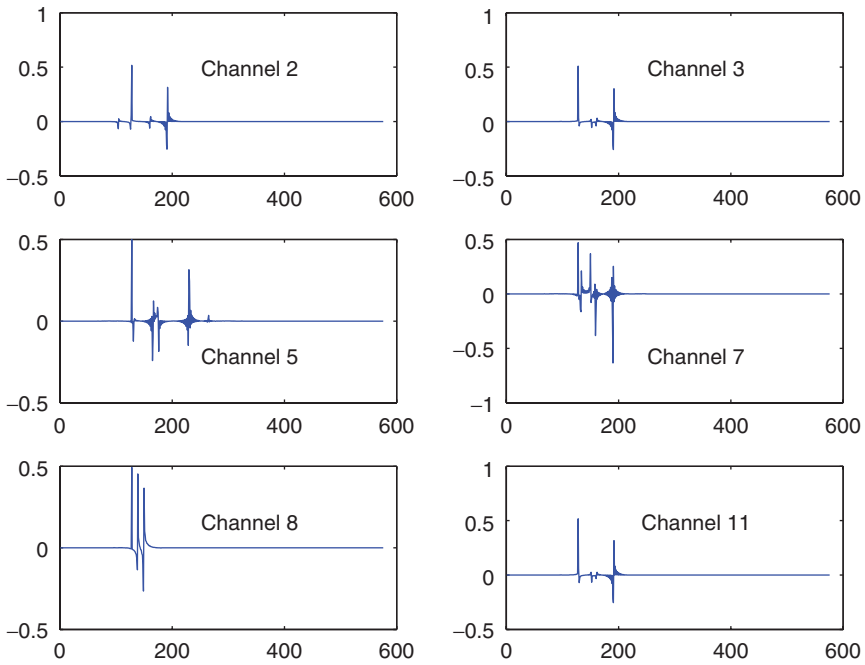
Figure 1.   Channels 2, 3, 5, 7, 8, and 11.

Table 1.   Brazil channels 7 and 8.

Name	Delays (in symbols)	Path gain (relative)
Brazil-7	0	1
	5.16587424	0.65575
	22.27783266	0.75697
	31.2104902	0.87482
	61.45237898	1.01565
	62.20573564	0.7379
Brazil-8	0	1
	10.762238	1
	21.524476	1

Table 2.   2-norm of error.

Channel (dB)	Exact	Zero	First	Second	Correlation
2 18	0.1293	0.1776	0.1580	0.1570	0.8967
2 28	0.0865	0.1206	0.1179	0.1211	0.8748
3 18	0.1206	0.1232	0.1224	0.1255	0.8838
3 28	0.0818	0.0766	0.0869	0.0928	0.8678
7 18	0.2406	0.4839	0.4070	0.3861	0.8007
7 28	0.1021	0.4331	0.3598	0.3398	0.7837
8 18	0.1737	0.2830	0.2664	0.2665	0.8805
8 28	0.0781	0.2289	0.2211	0.2244	0.8669
11 18	0.1264	0.1365	0.1353	0.1380	0.8827
11 28	0.0837	0.0956	0.1024	0.1069	0.8680

Table 3.   2-norm of error.

Channel (dB)	Exact	Zero	First	Second
2 20	0.1139	0.1578	0.1434	0.1438
2 22	0.1029	0.1434	0.1332	0.1346
2 24	0.0952	0.1331	0.1261	0.1283
2 26	0.0900	0.1258	0.1212	0.1240
3 20	0.1064	0.1050	0.1083	0.1126
3 22	0.0963	0.0925	0.0989	0.1040
3 24	0.0894	0.0844	0.0928	0.0984
3 26	0.0848	0.0794	0.0891	0.0949
5 20	0.1243	0.1984	0.1774	0.1750
5 22	0.1079	0.1869	0.1681	0.1664
5 24	0.0960	0.1794	0.1622	0.1609
5 26	0.0875	0.1746	0.1585	0.1575
7 20	0.1967	0.4627	0.3877	0.3674
7 22	0.1627	0.4493	0.3753	0.3553
7 24	0.1366	0.4410	0.3675	0.3476
7 26	0.1168	0.4360	0.3627	0.3428
8 20	0.1404	0.2612	0.2484	0.2499
8 22	0.1157	0.2469	0.2366	0.2390
8 24	0.0980	0.2378	0.2289	0.2319
8 26	0.0859	0.2322	0.2241	0.2273
11 20	0.1109	0.1195	0.1218	0.1255
11 22	0.0999	0.1083	0.1130	0.1172
11 24	0.0923	0.1015	0.1075	0.1120
11 26	0.0871	0.0976	0.1042	0.1088

lesser extent 5. The exact solution is computed using prior knowledge of the true CIR vector to form the covariance matrix $C(h)$ and compute the vector

$$h_{\text{Blue}} = \left(A^T C(h)^{-1} A\right)^{-1} A^T C(h)^{-1} y$$

This, of course, is the CIR vector which we are trying to estimate, and so knowledge of it is not available. We include the exact solution as a benchmark for the error to approach for the other methods of estimation.

We also show the norm of the error, $\|h - \hat{h}\|$ where $\hat{h}$ is the channel estimate and h is the true channel for channels 2, 3, 5, 7, 8, and 11 at 20, 22, 24, and 26 dB input SNR. We use simulated 8-VSB symbols passed through the channel at 20, 22, 24, and 26 dB AWGN level. We use the approximations as given in (17) and (18) where $i = 64$, which is the position of the main tap in the DFE that we use (this corresponds to zero delay in symbols for Brazil-D in table 2), where $\partial F(h)/\partial h_i$ is given by (21) and $\partial^2 F(h)/\partial h_i^2$ is given by (22), with $i = j$.

9.1. *Complexity*

We give the number of multiplications needed for each of our algorithms; that is the ZOA, FOA and SOA. For ZOA, this is the numbers of multiplications needed to form the product of the matrix on line (25) with the vector y of received data. For FOA and SOA, this is the number of multiplications needed to form the matrices on lines (28)

Table 4. Complexity of the algorithms.

	Complexity
ZOA	$O(L(N+L-1))$
FOA	$O(2L(N+L-1))$
SOA	$O(3L(N+L-1))$
Correlation	$O(L(N+L-1))$
Exact solution	$O((N+L-1)^3+L^3)$

(FOA), and (29) (SOA) and the number of multiplications to compute the matrix vector product of these matrices with the vector y of received data.

We contrast this with the complexity of computing the Correlation estimate, as given by equation (5) and computing the exact solution, as given by equation (14).

Recall that $L = N_a + N_c + 1$ (used from section 6 onwards) is the predetermined channel length ($L = 512$ is commonly used in HDTV applications), and that N is the length of the training data ($N = 704$ or 728 for the 8-VSB modulation system for HDTV applications.) Recall from (5) that $h_{\mathrm{corr}} = \hat{A} y_{[-N_a:N_c+N-1]}$ which takes $L \times (L+N-1)$ multiplicaitons to compute.

For the exact solution, we must solve (13), or more accurately, solve the equation (11) that preceded this, either by explicit matrix inversion or more likely, by some indirect (iterative) method or direct method that does not explicitly compute the inverse matrix that appears on the right-hand side of (13). We are even simplifying the situation which we encounter in (13), since we are overlooking the fact that the co-variance matrix Cov(w) is not available to us. This covariance matrix must be approximated somehow to give the exact solution. Overlooking this fact for now, the most naive approach to the issue of complexity in solving the exact problem needs two matrix inversions (each are $O((N+L-1)^3)$ and $O((L)^3)$) and then a further $L(N+L-1)$ multiplications to form the matrix vector product on the right-hand side of (13). Hence computing the exact solution, overlooking the fact that the true covariance matrix Cov(w) is not available, is $O((N+L-1)^3+L^3)$. Iterative methods to solve the system could be devised, but it is not trivial to do due to the structure of the equation (11), and the complexity is likely to still be $O(L^2)$ or worse for methods such as the conjugate gradient method (table 4). However, this is an aspect of the problem which could be worked on further.

10. Conclusions

Our algorithm gives an approximation to the BLUE CIR which is attained at a great reduction in computational complexity over iteratively computing the BLUE CIR for the specific channel. The extent of received data used in the data vector is sufficiently large for the purpose of forming an overdetermined system, that the effects of unknown data convolved with the CIR must be incorporated into the data covariance matrix used to weight the LS solution appropriately. We approximate the nonlinear function of the CIR that gives the BLUE CIR. The approximations that we use are given by the Taylor series representation of the function. We use a FOA (linear), SOA (quadratic), and third-order (cubic) approximation. Each approximation used is not the full

approximation given by the Taylor series, but is an approximation around the single dominant tap in the CIR vector. This approximation is determined by empirical testing, and is preferable to full approximations in each case, as the complexity is reduced.

Simulation results show improvements in the estimates over the correlation estimate (for example, see [20–26] for its use), and the ZOA, which has been previously used [27].

References

[1] John, G.P., 2001, *Digital Communications*, 4th Edn (Boston: McGraw Hill).

[2] Sergio, B. and Ezio, B., 1999, *Principles of Digital Transmission with Wireless Applications* (New York: Kluwer Academic/Plenum Publishers).

[3] Giannakis, G.B., *et al.*, 2001, *Signal Processing Advances in Wireless and Mobile Communications: Trends in Channel Estimation and Equalization* (Upper Saddle River, NJ: Prentice Hall).

[4] de Carvalho, E. and Slock, D.T.M., 2004, Blind and semi-blind FIR multichannel estimation: (global) identifiability conditions. *IEEE Transactions on Signal Processing*, **52**(4), 1053–1064.

[5] Medles, A., Slock, D.T.M. and De Carvalho, E., 2001, Linear prediction based semi-blind estimation of MIMO FIR channels. In: *Proceeding of the Third IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications*, Taiwan, Taoyuan, pp. 20–23.

[6] Lasaulce, S., Loubaton, P. and Moulines, E., 2003, A semi-blind channel estimation technique based on second-order blind method for CDMA systems. *IEEE Transactions on Signal Processing*, **51**(7), 1894–1904.

[7] Buchoux, V., *et al.*, 2004, On the performance of semi-blind subspace based channel estimation. *IEEE Transactions on Signal Processing*, **48**(6), 1750–59.

[8] Rontogiannis, A., *et al.*, 2002, Semiblind estimation of multi-path channel parameters via a separable least-squares approach. In: *DSP 2002 Proceedings of 14th International Conference on Digital Signal Processing*, 1, 49–52, 1–3 July.

[9] Report of the SET/ABERT DTV Committee, January 2000.

[10] Belfiore, C.A. and Park Jr, J.H., 1979, Decision-feedback equalization. *Proceedings IEEE*, **67**, 1143–1156.

[11] Belfiore, C.A., 1976, Decision-feedback equalization – a unified approach, PhD dissertation, University of Minnesota, Minneapolis, March.

[12] Haykin, S., 1996, *Adaptive Filter Theory* (Upper Saddle River, NJ: Prentice-Hall Information and Systems Science Series).

[13] Farhang-Boroujeny, B., 1998, *Adaptive Filters: Theory and Applications* (New York: John Wiley).

[14] Chong, E.K.P. and Zak, S., 1996, *An Introduction to Optimization* (New York: John Wiley).

[15] Pladdy, C., *et al.*, 2003, Best linear unbiased channel estimation for frequency selective multipath channels with long delay spreads, *Vehicular Technology Conference*, Orlando, FL, Fall 2003.

[16] Pladdy, C., *et al.*, 2006, Semiblind BLUE channel estimation with applications to digital television. *IEEE Transactions on Vehicular Technology*, **55**(6), 1812–1823.

[17] Kay, S.L., 1993, *Fundamentals of Statistical Signal Processing: Volume 1, Estimation Theory* (Upper Saddle River, NJ: Prentice-Hall Signal Processing Series).

[18] Mendel, J.M., 1995, *Lessons in Estimation Theory for Signal Processing, Communications and Control* (Upper Saddle River, NJ: Prentice-Hall Signal Processing Series).

[19] Bjorck, A., 1996, *Numerical Methods for Least Squares Problems* (Philadelphia, PA: Society for Industrial and Applied Mathematics).

[20] Cox, D.C., 1972, Delay Doppler characteristics of multipath propagation at 910 MHz in a suburban mobile radio environment. *IEEE Transactions on Antennas and Propagation*, **AP-20**, 625–635.

[21] Devasirvatham, D., 1986, Time delay spread and signal level measurements of 850 MHz radio waves in building environment. *IEEE Transactions on Antennas and Propagation*, **AP-34**, 1300–1305.

[22] Parsons, J., Demery, D. and Turkmani, A., 1991, Sounding techniques for wideband mobile radio channels: a review. *IEEE Proceedings-I*, **138**, 437–446.

[23] Jibrail, W.W. and Houmadi, A.-R., 1991, Acquisition of direct sequence spread spectrum signals using sliding correlators. *International Journal of Electronics*, **71**(5), 733–743.

[24] Ruprecht, J., 1989, Maximum likelihood estimation of multipath channels. PhD thesis, Swiss Federal Institute of Technology, Zurich.

[25] Kostić, Z., Sezan, M. and Titlebaum, E., 1992, Estimation of the parameters of a multipath channel using set-theoretic convolution. *IEEE Transactions on Communications*, **40**, 1006–1011.

[26] Montemayor, C.A. and Flikkema, P.G., 1998, Near-optimum iterative channel estimation of dispersive multipath channels, Conference Record of the Vehicular Technology Conference.

[27] Özen, S., 2003, Approximate best linear unbiased channel estimation for frequency selective multipath channels with long delay spreads, Asilomar Conference on Signals and Systems, Monterey, CA.
[28] Browder, A., 1996, *Mathematical Analysis: An Introduction* (New York: Springer Verlag).
[29] Lang, S., 1997, *Undergraduate Analysis*, 2nd Edn (New York: Springer Verlag).
[30] Engl, H.W., Hanke, M. and Neubauer, A., 1996, *Regularization of Inverse Problems* (Dordrecht, The Netherlands: Kluwer Academic Publishers).
[31] Vogel, C.R., 2002, *Computational Methods for Inverse Problems* (Philadelphia, PA: Society for Industrial and Applied Mathematics).
[32] Xu, G., *et al.*, 1995, A least-squares approach to blind channel identification. *IEEE Transactions on Signal Processing*, **43**(12), 2982–2993.
[33] Gelb, A., *et al.*, 1974, *Applied Optimal Estimation* (Cambridge, MA: MIT Press).
[34] Chung, H.-Y. and Sun, Y.-Y., 1988, Analysis and parameter estimation of nonlinear systems with Hammerstein model using Taylor series approach. *IEEE Transactions on Circuits and Systems*, **35**(2), 1539–1541.
[35] Heydt, G.T., Bakroun, M. and Inan, A., 2003, Voltage flicker estimation based on linearization and L_p norms. *IEEE Transactions on Power Delivery*, **18**(4), 1564–1566.
[36] Tsatsanis, M.K. and Giannakis, G.B., 1997, Subspace methods for blind estimation of time-varying FIR channels. *IEEE Transactions on Signal Processing*, **45**(12), 3084–3093.

Appendix

Proof of Proposition 5

Differentiate, using the product rule, with respect to h_i, the expression for $A_C(h)y$ and use Proposition 1 three times. That is, we have,

$$
\begin{aligned}
\frac{\partial F(h)}{\partial h_i} &= \frac{\partial}{\partial h_i}\big((A^T C(h)^{-1} A)^{-1}\big) A^T C(h)^{-1} y + (A^T C(h)^{-1} A)^{-1} A^T \frac{\partial}{\partial h_i}(C(h)^{-1}) y \\
&= (A^T C(h)^{-1} A)^{-1} A^T C(h)^{-1} \left(\frac{\partial}{\partial h_i} C(h)\right) C(h)^{-1} A (A^T C(h)^{-1} A)^{-1} \times A^T C(h)^{-1} y \\
&\quad - (A^T C(h)^{-1} A)^{-1} A^T C(h)^{-1} \times \left(\frac{\partial}{\partial h_i} C(h)\right) C(h)^{-1} y
\end{aligned}
$$

which gives the result.