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

Structured Channel Estimation Based Decision Feedback Equalizers for Sparse Multipath Channels with Applications to Digital TV Receivers

Serdar Özen, William Hillery, Michael Zoltowski

School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907-1285
{ozen, hilleryw, mikedz}@ecn.purdue.edu

S. M. Nereyanuru, Mark Fimoff

Zenith Electronics Corp.
2000 Millbrook Drive,
Lincolnshire, IL 60069
{snerayanuru, mark.fimoff}@zenith.com

Abstract

In this paper we investigate the performance of channel estimation based equalizers. We introduce two different channel estimation algorithms. Our first channel estimation scheme is a novel structured channel impulse response (CIR) estimation method for sparse multipath channels. We call this novel CIR estimation method Blended Least Squares *(BLS) which uses symbol rate sampled signals, based on blending the* least squares based channel estimation *and the* correlation *and* thresholding based channel estimation *methods. The second CIR estimation is called* Variable Thresholding *(VT), and is based on improving the output of the* correlation *and* thresholding *based estimation method. We then use these two CIR estimates to calculate the Decision Feedback Equalizer (DFE) tap weights. Simulation examples are drawn from the ATSC digital TV 8-VSB system [1]. The delay spread for digital TV systems can be as long as several hundred times the symbol duration; however digital TV channels are, in general,* sparse *where there are only a few dominant multipaths.*

1 Overview of Data Transmission Model

For the communications systems utilizing periodically transmitted training sequence, *least-squares* (LS) based channel estimation or the *correlation* based channel estimation algorithms have been the most widely used two alternatives. 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. In the sequel, although the examples following the derivations of the blended channel estimator will be drawn from the ATSC digital TV 8-VSB system [1], to the best of our knowledge it could be applied with minor modifications to any digital communication system with linear modulation which employs a training sequence.

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

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

where

$$I_k = \left\{ \begin{array}{ll} a_k, & 0 \le k \le N-1 \\ d_k, & N \le n \le N'-1, \end{array} \right\} \in \mathcal{A} \equiv \{\alpha_1, \cdots, \alpha_M\} \quad (2)$$

is the M-ary complex valued transmitted sequence, $\mathcal{A} \subset \mathbb{C}^1$, and $\{a_k \in \{\pm a \pm ja\}\}$ denote the first N symbols within a *frame* of length N' to indicate that they are the known training symbols; $\nu(t) = \eta(t) * q^*(-t)$ denotes the complex (colored) noise process after the pulse matched filter, with $\eta(t)$ being a zero-mean white Gaussian noise process with spectral density N_o 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

$$h(t) \;=\; p(t) * c(t) = \sum_{k=-K}^{L} c_k p(t - \tau_k), \qquad (3)$$

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 $T_q = N_q T$, $N_q \in \mathbb{Z}^+$. $\{c_k\} \subset \mathbb{C}^1$ 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 is slow enough that $c(t)$ can be assumed to be a static inter-symbol interference (ISI) channel, at least throughout the training period with the impulse response

$$c(t) \;=\; \sum_{k=-K}^{L} c_k \delta(t - \tau_k) \qquad (4)$$

for $0 \leq t \leq NT$, where N is the number of training symbols. The summation limits K and L denote the number of maximum anti-causal and causal multi-path delays respectively. The multi-path delays τ_k are *not* assumed to be at integer multiples of the sampling period T. Indeed it is one of the main contributions of this work that we show a robust way of recovering the pulse shape back into the composite channel estimate when the multi-path delays are not at the sampling instants.

1.1 Review of Least-Squares Channel Estimation

Without loss of generality symbol rate sampled composite CIR $h[n]$ can be written as a finite dimensional vector $\mathbf{h} = [h[-N_a], \cdots, h[0], \cdots, h[N_c]]^T$ where N_a and N_c denote the number of anti-causal and the causal taps of the channel, respectively, and $N_a + N_c + 1$ is the total memory of the channel. Based on Equation (1) 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

$$\mathbf{y} = \mathbf{A}\mathbf{h} + \boldsymbol{\nu}, \quad (5)$$

where

$$\mathbf{y} = [y[N_c], y[N_c+1], \cdots, y[N-1-N_a]]^T, \quad (6)$$

$$\mathbf{A} = \mathcal{T}\{[a_{N_c+N_a}, \cdots, a_{N-1}]^T, [a_{N_c+N_a}, \cdots, a_0]\} \quad (7)$$

where $\mathbf{A}$ 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 $\boldsymbol{\nu} = [\nu[N_c], \nu[N_c+1], \cdots, \nu[N-1-N_a]]^T$. As long as the matrix $\mathbf{A}$ is a tall matrix and of full column rank, that is **(i)** $N \geq 2(N_a + N_c) + 1$, **(ii)** rank$\{\mathbf{A}\} = N_a + N_c + 1$ then the least squares solution which minimizes the objective function $J_{LS}(\mathbf{h}) = \|\mathbf{y} - \mathbf{A}\mathbf{h}\|^2$ exists and unique, and is given by $\widehat{\mathbf{h}}_{LS} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{Y}$.

For a single antenna receiver the problems associated with the standard least squares based CIR estimation is summarized by Özen, et al[3].

2 Overview of the Proposed CIR Estimators

We will briefly overview the Variable Thresholding (VT) and Blended Least Squares (BLS). However both CIR estimation methods start from a raw channel estimate which is obtained by cross-correlating a_n (known and stored at the receiver) with the received sequence $y[n]$.

2.1 Initial Channel Estimation

Cross correlating the stored training sequence with the received sequence, which is primarily done for frame synchronization [2], yields a raw (uncleaned) channel estimate

$$\tilde{h}_u[n] = \frac{1}{r_a[0]} \sum_{k=0}^{N-1} a_k^* y[k+n], n = -N_a, \cdots, 0, \cdots, N_c \quad (8)$$

where $r_a[0] = \sum_{k=0}^{N-1} \|a_k\|^2$. Equivalently Equation (8) can be written as

$$\tilde{\boldsymbol{h}}_u = \frac{1}{r_a[0]} \tilde{\boldsymbol{A}}^H \tilde{\boldsymbol{y}}, \quad (9)$$

where

$$\tilde{\mathbf{A}} = \mathcal{T}\{[a_0, \cdots, a_{N-1}, \underbrace{0, \cdots, 0}_{N_a+N_c}]^T, [a_0, \underbrace{0, \cdots, 0}_{N_a+N_c}]\} \quad (10)$$

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 $\tilde{\boldsymbol{y}} = [y[-N_a], \cdots, y[N + N_c - 1]]^T$. We can express the observation vector $\tilde{\boldsymbol{y}}$ by

$$\tilde{\boldsymbol{y}} = \left(\tilde{\boldsymbol{A}} + \tilde{\boldsymbol{D}}\right)\boldsymbol{h} + \tilde{\boldsymbol{\nu}}, \quad (11)$$

where

$$\tilde{\boldsymbol{D}} = \mathcal{T}\{[\underbrace{0, \cdots, 0}_{N}, d_N, \cdots, d_{N_c+N_a+N-1}]^T, [0, d_{-1}, \cdots, d_{-N_c-N_a}]\}, \quad (12)$$

is a Toeplitz matrix which includes the unknown symbols before and after the training sequence, and $\tilde{\boldsymbol{\nu}} = [\nu[-N_a], \cdots, \nu[N + N_c - 1]]^T$ is the (colored) noise vector. 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

$$\boldsymbol{R}_{aa} = \frac{1}{r_a[0]} \tilde{\boldsymbol{A}}^H \tilde{\boldsymbol{A}}. \quad (13)$$

Then the *cleaned* channel estimate $\tilde{\boldsymbol{h}}_c$ is obtained from

$$\tilde{\boldsymbol{h}}_c = \boldsymbol{R}_{aa}^{-1} \tilde{\boldsymbol{h}}_u. \quad (14)$$

Substituting Equation (9) into (14) we get

$$\tilde{\boldsymbol{h}}_c = \boldsymbol{h} + \left(\tilde{\boldsymbol{A}}^H \tilde{\boldsymbol{A}}\right)^{-1} \tilde{\boldsymbol{A}}^H \left(\tilde{\boldsymbol{D}}\boldsymbol{h} + \tilde{\boldsymbol{\nu}}\right). \quad (15)$$

As can be seen from Equation (15) the channel estimate $\tilde{\boldsymbol{h}}_c$ has the contributions due to unknown symbols prior to and after the training sequence, which are elements of the matrix $\tilde{\boldsymbol{D}}$, as well as the additive channel noise; only the sidelobes due to aperiodic auto-correlation is removed.

If all the symbols involved in the correlation of Equation (9) were perfectly known then the baseline noise in the estimation vector would have been due to finite correlation of known symbols only. Then the cleaning algorithm would have cleaned this deterministic noise perfectly and we wouldn't have needed any thresholding on the cleaned estimation vector. But we always have unknown symbols involved in the correlation in most of the practical applications. So we need a thresholding algorithm after cleaning. The correlations of Equation (9) will ideally yield *peaks* at $\{D_k^a \in \mathbb{Z}^+\}$, for $k = -K, \cdots, -1, 0$, and at $\{D_k^c \in \mathbb{Z}^+\}$, for $k = 1, \cdots, L$. These delays are the sampling instants closest to the locations of the actual physical channel multipath TOAs $\{\tau_k\}$, $k = -K, \cdots, -1, 0, 1, \cdots, L$, within a symbol interval. If we apply a uniform thresholding to the vector obtained by Equation (14) which is in the form of setting the estimated channel taps to zero if they are below a certain preselected threshold; that is

$$\text{set} \quad \tilde{h}_{th}[n] = \begin{cases} 0, & \text{if } \|\tilde{h}_c[n]\| < \varepsilon \\ \tilde{h}_c[n], & \text{otherwise,} \end{cases} \tag{16}$$

for $n = -N_a, \cdots, -1, 0, 1, \cdots, N_c$, then in general we can choose the tap location with largest magnitude and denote it as the *cursor* (reference) path, and the TOAs prior to and after this reference TOA are denoted as pre- and post-cursor channel impulse responses. The relationship between the actual TOA's and the estimated TOAs is given by $D_k^a = -\text{round}\left(\frac{\tau_k}{T}\right)$, for $-K \le k \le 0$, and $D_k^c = \text{round}\left(\frac{\tau_k}{T}\right)$, for $1 \le k \le L$. It is assumed that $1 \le D_1^c < \cdots < D_L^c$, and similarly $1 \le D_1^a < \cdots < D_K^a$.

2.2 Overview of Variable Thresholding Algorithm

Though uniform thresholding of Equation (16) serves the purpose its effect is uniform through out the length of estimation vector. If we observe the noise characteristics due to unknown symbols it is obvious that the region near the cursor location in the channel estimation vector involves less number of unknown symbols than far regions. So variable thresholding algorithm is developed to threshold various regions of the estimation vector differently. At this point we have the cleaned channel estimation vector, $\tilde{\boldsymbol{h}}_c$, and thresholded channel estimation vector, $\tilde{\boldsymbol{h}}_{th}$.

1. Initial threshold, ε_{init}, is formed by the following equation

$$\varepsilon_{init}[n] = S\sigma_d\sqrt{|\mathcal{N}[n]|} \tag{17}$$

where S is the magnitude of the stored training symbol (=5 for 8-VSB case), σ_d is standard deviation of unknown data symbols (=$\sqrt{21}$ for 8-VSB case), and $\mathcal{N}[n]$ represents the number of unknown symbols appear in the channel estimation vector as a function of the location, n, in the channel estimation vector and is defined as

$$N(n) = \begin{cases} -n, & \text{if } -L_{ch} \le n < 0 \\ n, & \text{if } 0 \le n \le L_{ch} \end{cases} \tag{18}$$

where $L_{ch} = N_a + N_c + 1$ is the length of the channel, and N is the length of known (training) symbols.

2. Variable threshold ε_{vth} can be calculated as follows:

$$\varepsilon_{vth}[n] = \sum_k \varepsilon_{init}[n-k]\tilde{h}_{th}[k] \tag{19}$$

3. At the final stage variable thresholding will be done on the cleaned channel estimation vector, $\tilde{\boldsymbol{h}}_c$, by

$$\text{set } \widehat{h}_{vth}[n] = \begin{cases} 0, & \text{if } \|\tilde{h}_c[n]\| < \varepsilon_{vth}[n] \\ \tilde{h}_c[n], & \text{otherwise,} \end{cases} \tag{20}$$

for $-N_a \le n \le N_c$, and $\widehat{\boldsymbol{h}}_{vth}$ is the CIR estimate obtained by the variable thresholding algorithm.

2.3 Overview of Blended Least Squares Algorithm

The channel estimation is performed in two steps using symbol-spaced received samples after the receiver pulse matched filter. In the first step, the received samples are *correlated* with the stored training sequence, *cleaning* and *uniform thresholding* is applied, summarized by Equations (9,14,16) respectively, in order to determine the locations of the TOAs. The purpose of the second step is to incorporate the transmitted pulse shape $p(t)$ into the channel impulse response. To do this, we locate three copies of $p(t)$ shifted by one-half of a symbol period around each multipath location and estimate complex scaling factors using a modified least squares approach.

After the uniform thresholding takes place we will end up only with the peaks of the channel taps, which implies the fact that the tails of the pulse shape that are buried under the "noisy" correlation output may end up being zeroed out entirely. Now we will show to restore the pulse shape $p(t)$ into the CIR estimate. The idea is that for every multi-path we would like to approximate the shifted and scaled copies of the pulse shape $p(t)$ (shifted by τ_k and scaled by c_k) by a linear combination of three pulse shape functions shifted by half a symbol interval. More precisely

$$c_k p(nT - \tau_k) \approx \begin{cases} \sum_{l=-1}^{1} \gamma_l^{(k)} p((n + D_k^a - \frac{l}{2})T), & -K \le k \le 0 \\ \sum_{l=-1}^{1} \gamma_l^{(k)} p((n - D_k^c - \frac{l}{2})T), & 1 \le k \le L \end{cases} \tag{21}$$

where $\{\gamma_l^{(k)}, -K \le k \le L\}_{l=-1}^{1} \subset \mathbb{C}^1$. By making this approximation we can also efficiently recover the tails of the complex pulse shape $p(t)$ which are generally buried under the "noisy" output of the correlation processing, and are lost when uniform thresholding is applied. To accomplish this approximation we introduce three vectors $\boldsymbol{p}_k$, for $k = -1, 0, 1$, each containing T spaced samples of the complex pulse shape $p(t)$ shifted by $kT/2$, such that

$$\mathbf{p}_k = [p(-N_q T - \frac{kT}{2}), \cdots, p(-\frac{kT}{2}), \cdots, p(N_q T - \frac{kT}{2})]^T \tag{22}$$

for $k = -1, 0, 1$, and by concatenating these vectors side by side we define a $(2N_q + 1) \times 3$ matrix $\mathbf{P}$ by

$$\mathbf{P} = [\mathbf{p}_{-1}, \ \mathbf{p}_0, \ \mathbf{p}_1]. \quad (23)$$

Then we form the matrix denoted by Γ whose columns are composed of the shifted vectors $\mathbf{p}_k$, where the shifts represent the relative delays of the multi-paths; that is

$$\Gamma = \begin{bmatrix} \mathbf{P} & & \\ 0_{(D_K^a+D_L^c)\times 3} & \ddots & 0_{D_K^a \times 3} \\ & & \mathbf{P} \\ & 0_{D_L^c \times 3} & \ddots & 0_{(D_K^a+D_L^c)\times 3} \\ & & & \mathbf{P} \end{bmatrix} \quad (24)$$

where Γ is of dimension $(D_K^a + D_L^c + 2N_q + 1) \times 3(K + L + 1)$, and $\mathbf{0}_{m\times n}$ denotes an m by n zero matrix. Then the observation vector $\mathbf{y}$, and the convolution matrix, $\mathbf{A}$, composed only of the known training symbols are defined as in Equations (6, 7) respectively. Since it was assumed that $q(t)$ spans N_q symbol durations, it implies that $q[n]$ has $N_q + 1$ sample points, which in turn implies $p[n]$ has $2N_q + 1$ samples. Hence

$$N_a = D_K^a + N_q \quad \text{and} \quad N_c = D_L^c + N_q. \quad (25)$$

Defining $\gamma^{(k)} = [\gamma_{-1}^{(k)}, \gamma_0^{(k)}, \gamma_1^{(k)}]$ for $-K \le k \le L$, we define

$$\gamma = [\gamma^{(-K)}, \cdots, \gamma^{(0)}, \cdots, \gamma^{(L)}]^T, \quad (26)$$

as the unknown vector of the coefficients with $\{\gamma_n^{(k)}, n = -1, 0, 1; k = -K, \cdots, 0, \cdots, L\}$, of length $3(K+L+1)$. Then the vector that contains all the coefficients Then we can write the observation vector as

$$\mathbf{y} = \mathbf{A}\Gamma\gamma + \boldsymbol{\nu} \quad (27)$$

where $\boldsymbol{\nu}$ is the observation noise vector. Using the least squares arguments again, we can estimate the unknown coefficient vector γ as

$$\hat{\gamma}_{BLS} = \left(\Gamma^H \mathbf{A}^H \mathbf{A}\Gamma\right)^{-1} \Gamma^H \mathbf{A}^H \mathbf{y}. \quad (28)$$

Once the vector $\hat{\gamma}_{LS}$ is obtained, the new channel estimate $\hat{\mathbf{h}}_{new}$, where the pulse tails are recovered back, can simply be obtained by

$$\hat{\mathbf{h}}_{BLS} = \Gamma\hat{\gamma}_{BLS}. \quad (29)$$

In order to have a unique $\hat{\gamma}_{BLS}$ it is required that $\mathbf{A}\Gamma$ be a tall matrix, that is it is required that

$$N - D_K^a - D_L^c - 2N_q \ge 3(K + L + 1). \quad (30)$$

2.4 Noise Variance Estimation

Once we define the estimated channel vector as in Equation (29) we can similarly define the reconstructed observation vector $\hat{\mathbf{y}}$, based on $\hat{\mathbf{h}}_{BLS}$, by

$$\hat{\mathbf{y}} = \mathbf{A}\hat{\mathbf{h}}_{BLS} = \mathbf{A}\Gamma\hat{\gamma}_{BLS} \quad (31)$$

$$= \mathbf{A}\Gamma\left(\Gamma^H \mathbf{A}^H \mathbf{A}\Gamma\right)^{-1} \Gamma^H \mathbf{A}^H \mathbf{y}. \quad (32)$$

Then the variance of the noise samples at the output of the matched filter σ_ν^2 can be estimated using the estimator

$$\widehat{\sigma^2}_\nu = \frac{1}{(N - D_K^a - D_L^c - 2N_q)} \|\hat{\mathbf{y}} - \mathbf{y}\|^2 \quad (33)$$

which arises naturally. Note that $(N - D_K^a - D_L^c - 2N_q)$ is the length of the observation vector $\mathbf{y}$.

3 Channel Estimate-Based Decision Feedback Equalizer

The derivations of the DFE equations are based on the recent work by Zoltowski et al[4]. Based on the minimum mean squared error (MMSE) criterion at the slicer input, the optimum feed-forward and feedback taps in a DFE are directly computable from an estimate of the channel.

The equalizer is assumed to have $N_F + 1$ feedforward taps and N_B feedback taps. Assembling $N_F + 1$ consecutive samples of $y[k]$ into a vector $\boldsymbol{y}$ yields

$$\boldsymbol{y}[k] = \boldsymbol{H}\boldsymbol{s}[k] + \boldsymbol{Q}\boldsymbol{\eta}[k], \quad (34)$$

where $\boldsymbol{\eta}[k] = [\,\eta[k + L_q] \dots \eta[k - L_q - N_F]\,]^T$, $\boldsymbol{s}[k] = [\,I[k + N_a] \dots I[k - N_c - N_F]\,]^T$,

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{h}^T & 0 & \cdots & 0 \\ 0 & \boldsymbol{h}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{h}^T \end{bmatrix}_{(N_F+1)\times(N_F+N_c+N_a+1)} \quad (35)$$

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{q}^T & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}^T \end{bmatrix}_{(N_F+1)\times(N_F+N_q)} \quad (36)$$

where $\boldsymbol{q}$ is the finite length representation of the receiver matched filter with length $N_q = 2L_q + 1$:

$$\boldsymbol{q} = [\,q[-L_q], \dots, q[-1], q[0], q[1], \dots, q[L_q]\,]^T. \quad (37)$$

The symbol estimate $\hat{I}[k - \delta]$ is given by

$$\hat{I}[k - \delta] = \mathrm{Re}\{\boldsymbol{g}_F{}^H \boldsymbol{y}[k]\} + \boldsymbol{g}_B{}^T \boldsymbol{I}_B[k - \delta - 1], \quad (38)$$

where δ is the cursor location (defined below), $\boldsymbol{g}_F = [\,g_F[0], \dots, g_F[N_F]\,]^T$, $\boldsymbol{g}_B = [\,g_B[1], \dots, g_B[N_B]\,]^T$,

$s_B[k] = [I[k], \ldots, I[k+1-N_B]]^T$, and the superscript H denotes conjugate transpose. In this analysis we assume that all decisions are correct. If we adopt the convention that the subscripts R and I indicate the real and imaginary parts of quantities, then

$$\begin{aligned}\hat{I}[k-\delta] &= {g_{FR}}^T y_R[k] + {g_{FI}}^T y_I[k] \\ &\quad + {g_B}^T s_B[k-\delta-1] \\ &= {g_{FC}}^T y_C[k] + {g_B}^T s_B[k-\delta-1], \quad (39)\end{aligned}$$

where $g_{FC} = [{g_{FR}}^T \; {g_{FI}}^T]^T$ and $y_C[k] = [{y_R}^T[k] \; {y_I}^T[k]]^T$. We may now write

$$y_C[k] = H_C s[k] + Q_C \eta_C[k], \quad (40)$$

where $H_C = [{H_R}^T \; {H_I}^T]^T$, $Q_C = \begin{bmatrix} Q_R & -Q_I \\ Q_I & Q_R \end{bmatrix}$, and $\eta_C[k] = [{\eta_R}^T[k] \; {\eta_I}^T[k]]^T$.

In the decision feedback equalizer, we use the cursor to define exactly which symbol is being estimated. To motivate the cursor definition, we use the real equalizer and consider a channel containing a single path with real gain at delay zero. In this case, $h_R[k]$ is a delta function at $k=0$ and the sample $y_R[k]$ corresponds to the symbol $I[k]$ (since $y_R[k] = I[k] + \nu_R[k]$). Then the feedforward term in the symbol estimate is

$$\begin{aligned}{g_F}^T y_R[k] &= \sum_{n=0}^{N_F} g_F[n] y_R[k-n] \\ &= \sum_{n=0}^{N_F} g_F[n] \left(I[k-n] + \nu_R[k-n]\right), \quad (41)\end{aligned}$$

where we recall that g_F is real for this discussion. That is, the feedforward term in the equalizer output only depends on the symbols $I[k], I[k-1], \ldots, I[k-N_F]$. When there is multipath interference, the feedforward term will depend on symbols covering a wider time-span, but since we may encounter a channel where the multipath is negligible, we may only consider $I[k], I[k-1], \ldots, I[k-N_F]$ as candidates for the symbol to estimate. Therefore, we estimate the symbol $I[k-\delta]$, where $0 \le \delta \le N_F$, and call this symbol the *cursor*. Since there is a one-to-one correspondence between the candidate symbols and the taps in the feedforward section, we often identify the cursor by the corresponding feedforward tap. This definition of the cursor is consistent with standard DFE theory and practice.

We now find the equalizer by minimizing the mean-square error (MSE), where

$$\begin{aligned}\text{MSE} &= \text{E}\{(I[k-\delta] - \hat{I}[k-\delta])^2\} \\ &= \mathcal{E}_s - 2{g_{FC}}^T r_{sy_C} + 2{g_{FC}}^T R_{y_C s_B} g_B \\ &\quad + {g_{FC}}^T R_{y_C y_C} g_{FC} + \mathcal{E}_s {g_B}^T g_B \quad (42)\end{aligned}$$

with $\mathcal{E}_s = \text{E}\{(I[k])^2\}$, $r_{sy_C} = \text{E}\{I[k-\delta] y_C[k]\}$, $R_{y_C s_B} = \text{E}\{y_C[k] {I_B}^T[k-\delta-1]\}$, and $R_{y_C y_C} = \text{E}\{y_C[k] y_C^T[k]\}$. Here we have made use of the expressions $R_{s_B s_B} = \text{E}\{s_B[k] {s_B}^T[k]\} = \mathcal{E}_s I_{N_B}$ and $r_{ss_B} = \text{E}\{I[k-\delta] s_B[k]\} = 0$. Minimization of this expression yields

$$\begin{aligned}g_{FC} &= \left(R_{y_C y_C} - \frac{1}{\mathcal{E}_s} R_{y_C s_B} {R_{y_C s_B}}^T\right)^{-1} r_{sy_C} \quad (43) \\ g_B &= -\frac{1}{\mathcal{E}_s} {R_{y_C s_B}}^T g_{FC}, \quad (44)\end{aligned}$$

where

$$\begin{aligned}r_{sy_C} &= \mathcal{E}_s H_C \mathbf{1}_{\delta+N_a} \quad (45) \\ R_{y_C y_C} &= \mathcal{E}_s H_C {H_C}^T + N_0 Q_C {Q_C}^T \quad (46) \\ R_{y_C s_B} &= \mathcal{E}_s H_C \Delta_\delta. \quad (47)\end{aligned}$$

The vector $\mathbf{1}_{\delta+N_a}$ contains all zeros except for a one in element $\delta + N_a + 1$. The matrix Δ_δ is defined by

$$\Delta_\delta = \begin{bmatrix} 0_{(N_a+\delta+1)\times N_B} \\ I_{N_B} \\ 0_{(N_F+N_c-\delta-N_B)\times N_B} \end{bmatrix} \quad (48)$$

when $0 \le \delta \le N_c + N_F - N_B$ and

$$\Delta_\delta = \begin{bmatrix} 0_{(N_a+\delta+1)\times(N_F+N_c-\delta)} & 0_{(N_a+\delta+1)\times(\delta+N_B-N_F-N_c)} \\ I_{N_F+N_c-\delta} & 0_{(N_F+N_c-\delta)\times(\delta+N_B-N_F-N_c)} \end{bmatrix} \quad (49)$$

when $N_c + N_F - N_B < \delta \le N_c + N_F - 1$. Here I_n is the identity matrix of size $n \times n$.

The derivation of the real equalizer is similar with the result that

$$\begin{aligned}g_F &= \left(R_{y_R y_R} - \frac{1}{\mathcal{E}_s} R_{y_R s_B} {R_{y_R s_B}}^T\right)^{-1} r_{sy_R} \quad (50) \\ g_B &= -\frac{1}{\mathcal{E}_s} {R_{y_R s_B}}^T g_{FR}, \quad (51)\end{aligned}$$

where

$$\begin{aligned}r_{sy_R} &= \mathcal{E}_s H_R \mathbf{1}_{\delta+N_a} \quad (52) \\ R_{y_R y_R} &= \mathcal{E}_s H_R {H_R}^T + N_0 \left(Q_R {Q_R}^T + Q_I {Q_I}^T\right) \\ &= \mathcal{E}_s H_R {H_R}^T + N_0 I_{N_F+1} \quad (53) \\ R_{y_R s_B} &= \mathcal{E}_s H_R \Delta_\delta. \quad (54)\end{aligned}$$

4 Simulations

We compare the accuracy of the channel estimates $\widehat{h}_{vth}$ and $\widehat{h}_{BLS}$ and quantify their effectiveness by considering

Table 1: Simulated channel delays in symbol periods, relative gains

Taps	Channel 1		Channel 2		Channel 3	
k	$\{\tau_k\}$	$\{\lvert c_k\rvert\}$	$\{\tau_k\}$	$\{\lvert c_k\rvert\}$	$\{\tau_k\}$	$\{\lvert c_k\rvert\}$
−2	−23.89	0.15				
−1	−1.61	0.20	−0.95	0.72	−19.03	0.12
0	**0**	**1**	**0**	**1**	**0**	**1**
1	32.82	0.17	3.55	0.64	1.58	0.1
2	63.06	0.20	15.25	0.98	19.03	0.1
3	63.82	0.15	24.03	0.74	60.27	0.31
4			29.16	0.86	190.33	0.19

their use in the equalization of sparse channels. We considered a DFE with real valued taps, and with 256 feed-forward taps and 448 feedback taps. The cursor tap is placed at the 221st tap on the feed-forward filter.

We considered an 8-VSB [1] receiver with a single antenna. 8-VSB system has a complex raised cosine pulse shape [1]. The CIRs we considered are given in Table 1. The phase angles of individual paths for all the channels are taken to be

$$\arg\{c_k\} = \exp(-j2\pi f_c \tau_k), \quad k = 1, \cdots, 6 \quad (55)$$

where $f_c = \frac{50}{T_{sym}}$ and $T_{sym} = 92.9$nsec.

Based on each of channel estimates the optimal real valued DFE filter coefficients are obtained, and the equalization performance is compared. The DFE output (slicer input) signal-to-interference-plus-noise-ratio denoted by SINR_{DFE}, is the metric used for comparison of equalizers obtained from the channel estimates. The output of the equalizer is given by

$$\hat{I}[k] = \tilde{c}[0] * I[k] + \sum_{\forall n, n \neq 0} \tilde{c}[n] * I[k-n] + v[k] \quad (56)$$

where the first term in Equation (56) is the desired symbol with a multiplicative constant, and $\tilde{c}[k]$ is the effective channel at the output of the equalizer. $\tilde{c}[0] = 1$ only when the estimate is unbiased. The second term (the summation) in Equation (56) is the residual ISI and the last term ($v[k]$) is noise (colored). SINR_{DFE} is then defined as the variance of the desired term divided by the variance of everything else. Since the symbols and noise are uncorrelated, this turns out to be

$$\text{SINR}_{\text{DFE}} = \frac{E_s \tilde{c}[0]^2}{E_s \sum_{\forall n, n \neq 0} \tilde{c}[n]^2 + E\{v[k]^2\}}. \quad (57)$$

The SINR_{DFE} values for each channel estimate is provided for 18 to 28dB Signal-to-Noise-Ratio (SNR) values measured at the input to the receive pulse matched filter, and it is calculated by

$$\text{SNR} = E_s \left\| (c(t) * q(t))|_{t=nT} \right\|^2 / N_0, \quad (58)$$

where $E_s = 21$ is the symbol energy for 8-VSB system, and N_0 is the channel noise variance.

Figure 1 parts (a), (d), and (g) show the actual CIR's, and Figure 1 parts (b), (e), and (h) show the estimated CIRs by the VT algorithm, and Figure 1 parts (d), (f), and (i) show the estimated CIRs by the BLS algorithm, for Channels 1-3 respectively. Due to space limitations only the real parts of the CIRs have been illustrated. The VT and BLS algorithms operated at 18dB SNR. Figure 2 illustrates the SINR_{DFE} versus SNR for Channels 1–3, and for ideal CIR based DFE, for VT based CIR, and for BLS based CIR. It is clear that BLS generally outperforms the VT based channel estimation algorithm. However as in the case channel 3, it is possible for the uniform thresholding algorithm perform poorly and it can miss smaller multipaths. Then the quality of BLS may not very good; on the other hand BLS may still be as good as the variable thresholding algorithm.

Acknowledgements

This research was funded by the National Science Foundation under grant number CCR-0118842 and by the Zenith Electronics Corporation.

References

[1] ATSC Digital Television Standard, A/53, September 1995.

[2] 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, Sep. 2001.

[3] 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. 2281–84, 2002.

[4] M.D. Zoltowski, W. Hillery, S. Özen, M. Fimoff, "Conjugate-gradient-based decision feedback equalization with structured channel estimation for digital television," to be published in Proceedings of SPIE: *Digital Wireless Communications*, vol. 4740, 2002.

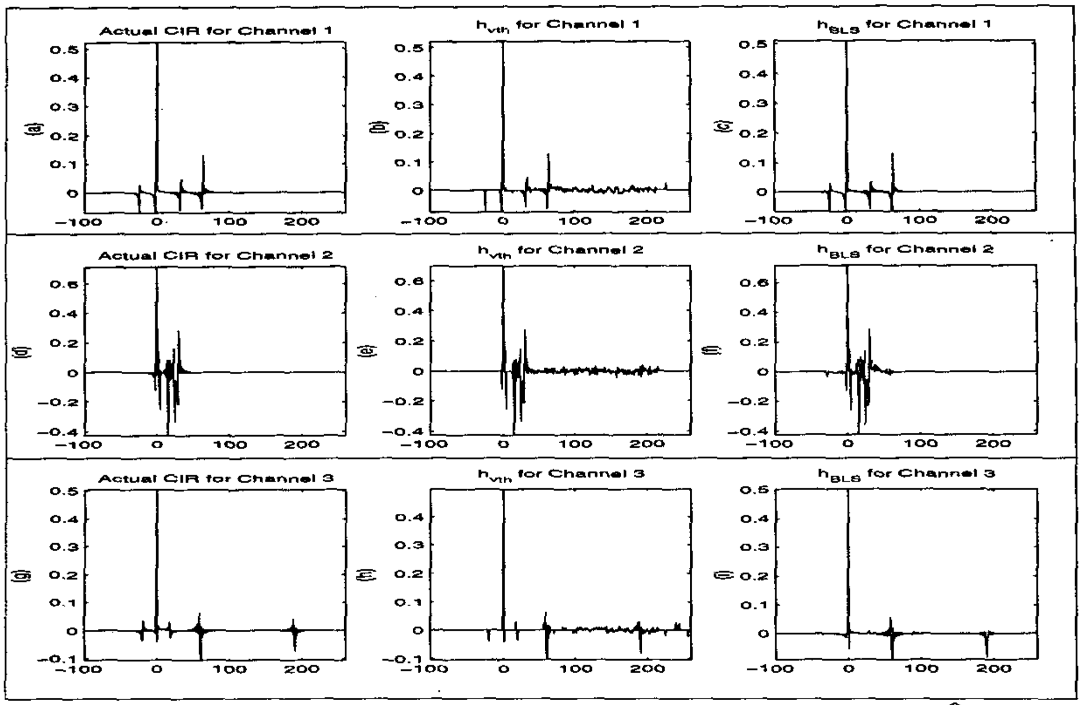

Figure 1: (a), (d), and (g) show the actual real part of the CIRs; (b), (e) and (h) show the real part of $\widehat{h}_{vth}$; (c), (f), and (i) show the real part of $\widehat{h}_{BLS}$.

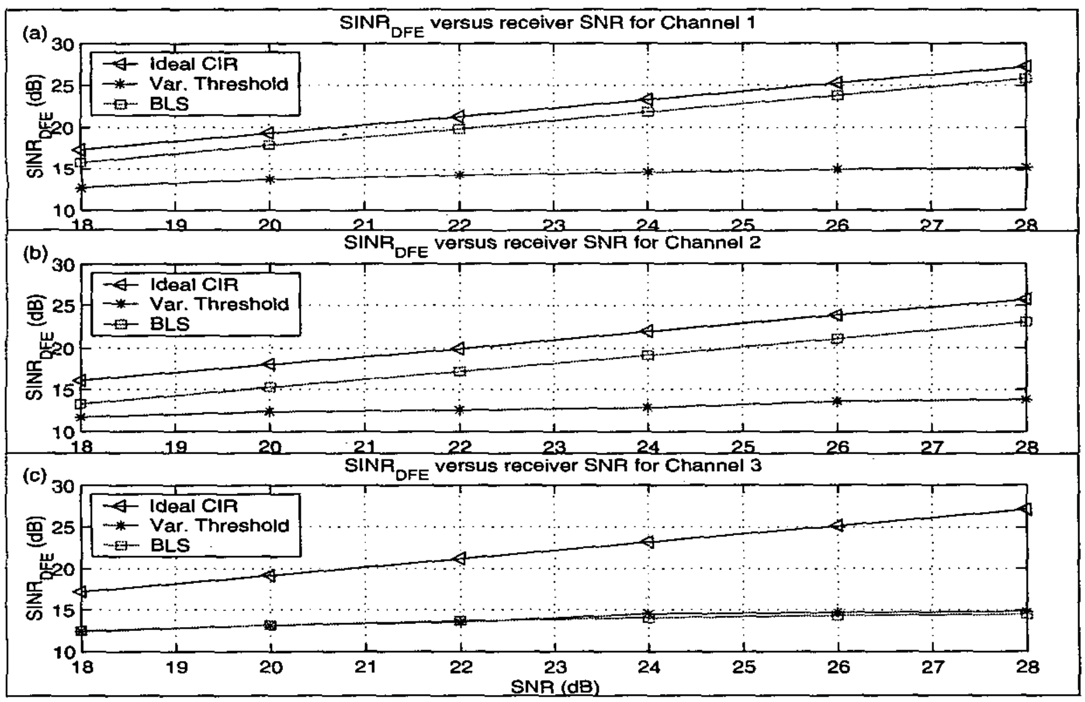

Figure 2: $SINR_{DFE}$ versus SNR for Channels 1–3, and for ideal CIR based DFE, for VT based CIR, and for BLS based CIR.