ECE 5654 Lecture 22
DFE & Adaptive Equalization

Ravi Tandon

Virginia Tech

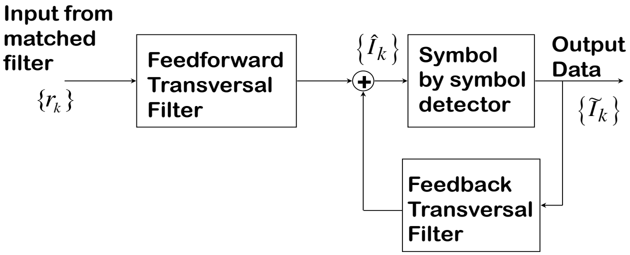
April 23, 2014

Learning Objectives

At the end of this lecture, the student should be able to:

- Describe Decision Feedback equalization
- Describe Adaptive Equalization Schemes

Decision Feedback Equalization



- Once the output symbols are estimated, they can be used to help estimate the channel impulse response
- DFE can result in complete elimination of inter-symbol interference provided decisions are correct and the feedback filter is long enough to compensate for the longest ISI
- Feedforward filter can be ZF/MMSE

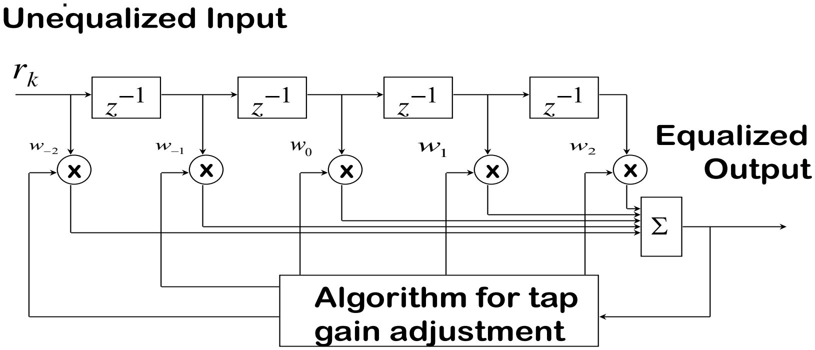
More on DFE

- Input to Feed-forward filter is the received signal sequence
- Feedback filter takes the previously detected symbols as input
- Feedback filter removes that part of ISI caused by the previously detected symbols
- Since the detector feeds hard decisions, DFE is a non-linear equalizer

Adaptive Equalization

- Previously we examined equalizers which can improve performance in ISI channels
- In time varying channels, the optimal equalizer coefficients will very with time and thus must be constantly estimated and updated
- This update technique is called an adaptive algorithm
- The algorithm used depends both on the minimization criterion and the update approach

Linear Transversal Filter



$$r_k = \sum_n I_n h_{k-n} + \nu_k$$

Linear Transversal Filter Equalizer

- Complex samples of the received signal input into the equalizer, i.e., r_k
- Each element z^{-1} represents one sample delay
- $2K+1$ taps on the equalizer (i.e., $2K+1$ complex coefficients w_k)
- Equalized output samples: $\hat{I}_k = \sum_{j=-K}^{K} w_j r_{k-j}$
- Decision is made based on $\hat{I}_k$

Distortion Criteria

- Choice of tap coefficients depends on choice of decision criteria
- Peak distortion criteria leads to zero-forcing equalizer
- Mean Squared Error (MSE) criteria is much more widely used: $E[|(I_k - \hat{I}_k)|^2]$

Summary of Non Adaptive MMSE Solution

- MMSE equalizer: select $\mathbf{w}$ to minimize $E[|(I_k - \sum_{j=-K}^{K} w_j r_{k-j})|^2]$
- The optimal MMSE equalizer selects $\mathbf{w} = [w_{-K} \ldots w_0 \ldots w_K]^T$ as:

$$\mathbf{w} = \mathbf{R}_{rr}^{-1}\mathbf{d} \tag{1}$$

where

- $\mathbf{R}_{rr}$ is the covariance matrix of the input data, i.e.,

$$\mathbf{R}_{rr} = E[\mathbf{r}\mathbf{r}^T]$$

- $\mathbf{d}$ is the correlation between the received signal and the information sequence

$$\mathbf{d} = E[\mathbf{r} I_k] = \mathbf{h}$$

More on MMSE

- Covariance matrix:

$$\begin{aligned}\mathbf{R}_{rr} &= E[\mathbf{r}\mathbf{r}^T] \\ &= \left[\begin{pmatrix} \sum_{i=0}^{L-1} h_i I_{k-K-i} + \nu_{k-K} \\ \vdots \\ \sum_{i=0}^{L-1} h_i I_{k+K-i} + \nu_{k+K} \end{pmatrix} \begin{pmatrix} \sum_{i=0}^{L-1} h_i I_{k-K-i} + \nu_{k-K} \\ \vdots \\ \sum_{i=0}^{L-1} h_i I_{k+K-i} + \nu_{k+K} \end{pmatrix}^T \right] \\ &= \mathbf{H}\mathbf{H}^T + \frac{N_0}{2}\mathbf{I}\end{aligned}$$

- Cross-correlation vector:

$$\mathbf{d} = \begin{bmatrix} h_{K+1} \\ \vdots \\ h_0 \\ \vdots \\ 0 \end{bmatrix}$$

Problems with Non-Adaptive Solution

- The direct solution requires estimation of $\mathbf{R}_{rr}$ and $\mathbf{d}$ followed by inversion of $\mathbf{R}_{rr}$ and multiplication with $\mathbf{d}$
- This approach is also sometimes called DMI (direct matrix inversion)
- Requires direct computation of a full matrix inversion with every update
- Explicitly makes use of channel parameters which must be estimated
- Combined, this results in a very computationally intense approach, especially in a time varying environment
- Solution: Real-time Adaptive Algorithms
- Least Mean Square (LMS) (Gradient search technique)
- Recursive Least Squares (RLS) (Estimates matrix inverse)

Least Mean Squares (LMS) Algorithm

- We begin with an initial guess for the solution $\mathbf{w}_0$
- We form a sequence of estimates $\mathbf{w}_0, \mathbf{w}_1, \mathbf{w}_2, \ldots$, of the solution to the equation $\mathbf{R}_{rr}\mathbf{w} = \mathbf{d}$
- The estimates of solution are found using the steepest descent method
- Estimates are updated iteratively:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \Delta \cdot \mathbf{G}_k$$

where

- Δ is a small constant to control the step size of the update
- $\mathbf{G}_k$ is the gradient vector of the error (i.e., the derivative of the MSE wrt the coefficients)

$$\mathbf{G}_k = -E[e_k \mathbf{r}_k^*]$$

Gradient Vector in LMS Algorithm

- Calculation of the gradient vector requires knowledge of covariance matrix $\mathbf{R}_{rr}$ and cross-correlation vector $\mathbf{d}$
- In practice, we use the actual samples as the unbiased estimate of the true gradient
- This is the stochastic version of the method of steepest descent
- The LMS Algorithm:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \Delta \cdot e_k \mathbf{r}_k^*$$

- Note that $e_k \mathbf{r}_k^*$ is the unbiased estimate of the gradient

Interpretation of LMS Algorithm

- Coefficients are adjusted in small steps to reduce the error in the response of the equalizer to the current set of input samples
- Coefficients can be updated after each new sample
- Note that this algorithm implicitly makes use of channel parameters, but only makes use of the received signal samples which we can directly measure
- Error calculation (e_k) requires knowledge of the information symbols I_k, which are in general unknown.
- Thus, we must send a training sequence periodically.

Role of Step Size in LMS Algorithm

- Larger Δ means faster convergence to optimum
- However, Δ cannot be arbitrarily large
- $\Delta < \frac{2}{\lambda_{max}}$, to guarantee convergence of algorithm, where λ_{max} is the largest eigenvalue of the covariance matrix $\mathbf{R}_{rr}$
- Larger Δ also means that the algorithm will be less likely to finely converge to the optimal solution at the end - sometimes called "excess MSE"
- Too small a choice for Δ may prevent the LMS algorithm from adapting to a rapidly-varying channel

Key Weakness of the LMS Algorithm

- Clearly, a good choice of Δ must be made to avoid too slow initial convergence or poor final convergence and instability
- The same step size Δ must be used for the update of all coefficients, even though some may be more sensitive to instability and others may be more important to update rapidly
- Solution: Recursive Least Squares Algorithm

Recursive Least Squares (RLS) Algorithm

- The RLS algorithm implements the optimal solution $\mathbf{w} = \mathbf{R}_{rr}^{-1}\mathbf{d}$ by iteratively estimating $mathbfR_{rr}$ and $\mathbf{d}$ and then using the matrix inversion lemma
- Stochastic estimate of $\mathbf{R}_{rr}$:

$$\mathbf{R}_k = \alpha \mathbf{R}_{k-1} + \mathbf{r}_k \mathbf{r}_k^{\dagger}$$

- Matrix Inversion Lemma:

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}$$

- Stochastiv inverse using matrix inversion lemma gives:

$$\mathbf{R}_k^{-1} = \frac{1}{\alpha}\left(\mathbf{R}_{k-1}^{-1} - \frac{\mathbf{R}_{k-1}^{-1}\mathbf{r}_k\mathbf{r}_k^{\dagger}\mathbf{R}_{k-}^{-1}}{\alpha + \mathbf{r}_k^{\dagger}\mathbf{R}_{k-1}^{-1}\mathbf{r}_k}\right)$$

Recursive Least Squares (RLS) Algorithm

- We then update the weights according to:

$$\begin{aligned}
\mathbf{w}_{k+1} &= \mathbf{R}_{k+1}^{-1}\mathbf{d}_{k+1} \\
&= \frac{1}{\alpha}\left(\mathbf{R}_k^{-1} - \mathbf{K}_{k+1}\mathbf{r}_{k+1}^{\dagger}\mathbf{R}_k^{-1}\right)\left(\alpha\mathbf{d}_k + I_k\mathbf{r}_{k+1}^{*}\right) \\
&= \mathbf{w}_k + \mathbf{K}_{k+1}\left(I_k - \mathbf{r}_{k+1}^{\dagger}\mathbf{w}_k\right) \\
&= \mathbf{w}_k + \mathbf{K}_{k+1}e_k
\end{aligned}$$

- Here, we have defined

$$\mathbf{K}_{k+1} = \frac{\mathbf{R}_k^{-1}\mathbf{r}_{k+1}}{\alpha + \mathbf{r}_{k+1}\mathbf{R}_k^{-1}\mathbf{r}_{k+1}^{\dagger}} = \frac{\mathbf{P}_k\mathbf{r}_{k+1}}{\alpha + \mathbf{r}_{k+1}\mathbf{P}_k\mathbf{r}_{k+1}^{\dagger}}$$

- This is also known as the Kalman Gain vector
- $\mathbf{P}_k$ is the recursively updated inverse of the correlation matrix

RLS Algorithm

- The recursive method of computing the matrix inversion is at the heart of the RLS algorithm
- The coefficient $0 < \alpha < 1$ is a scalar which we will discuss in a moment
- Rather than minimizing the average mean-square error like the LMS, the RLS algorithm is minimizing the actual squared error for the given data sequence
- The RLS algorithm is an implementation of a more general technique known as Kalman Filtering
- The Kalman gain replaces the fixed step size in updating the equalizer coefficients

Role of the Weighting Factor α

- The weighting factor $0 < \alpha < 1$ determines how quickly older data is discarded from the updated estimate
- Large α means that older data included in the estimate for a longer period of time
- Small α means that older data is discounted quickly

RLS vs LMS

- The RLS algorithm converges faster than the LMS algorithm
- The RLS algorithm is more computationally complex than the LMS algorithm
- Note that the recursive nature of the RLS algorithm is still simpler than a full matrix inversion operation
- Note that the RLS algorithm operates directly on signal samples

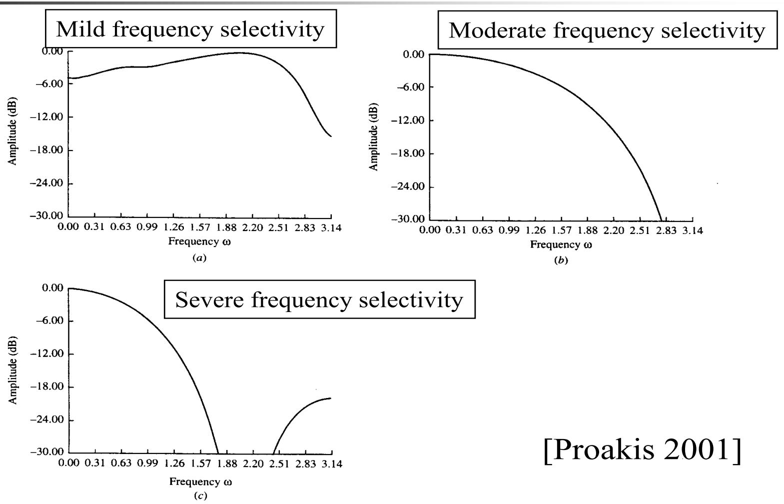
Trained vs. Blind Equalization

- Note that the computation of the error $e_k = I_k - \hat{I}_k$ is required for implementation of these algorithms
- Many systems use initial training sequences for which I_k is known to both transmitter and receiver
- Example: GSM transmits periodic training sequences as part of each signal frame
- Recently, there has been interest in "Blind Equalization" techniques, which require no training at all

Summary of Adaptive Algorithms

- Two adaptive algorithms were presented to adapt equalizer coefficients
- The LMS algorithm is less complex
- The RLS algorithm converges faster
- User-specified parameters determine the speed of convergence for each of these algorithms

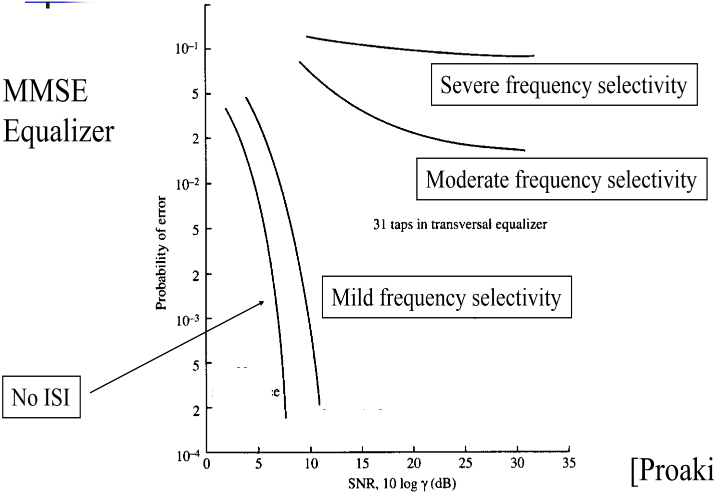
Three Frequency Selective Channels



[Proakis 2001]

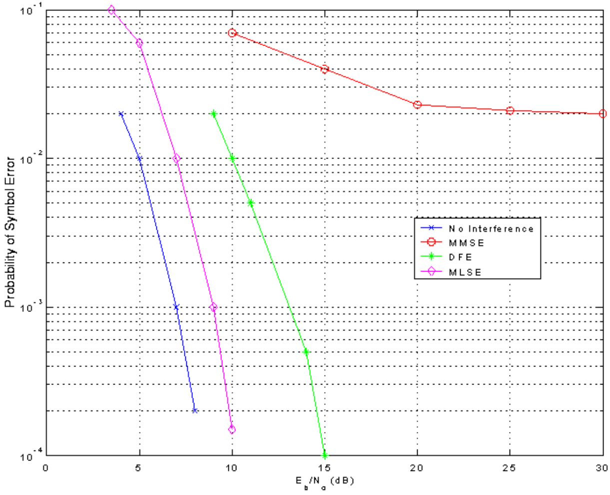
Performance of Linear Transversal Filter Equalizer

MMSE Equalizer



[Proakis 2001]

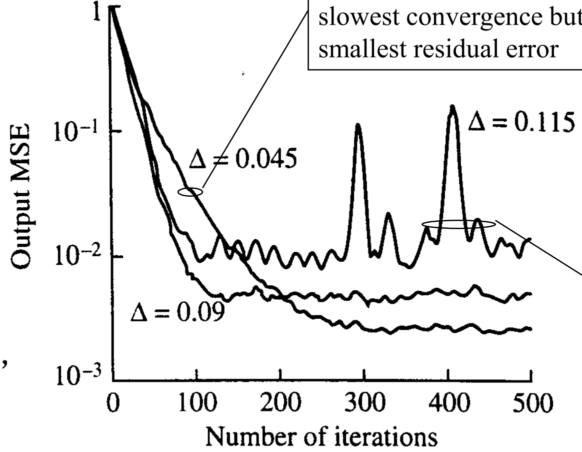
Performance of different Equalizers



Moderate Frequency Selectivity

DFE can provide substantial gains

Effect of Step Size on LMS Algorithm



Smallest step size results in slowest convergence but smallest residual error

Large step size results in fast convergence but high residual error and undesirable instability

[Proakis 2001]

Convergence Rate of LMS (Gradient) vs RLS (Kalman)