Department of Computer Engineering & Informatics

Signal Processing and Communications Lab

Wireless and Mobile Communications

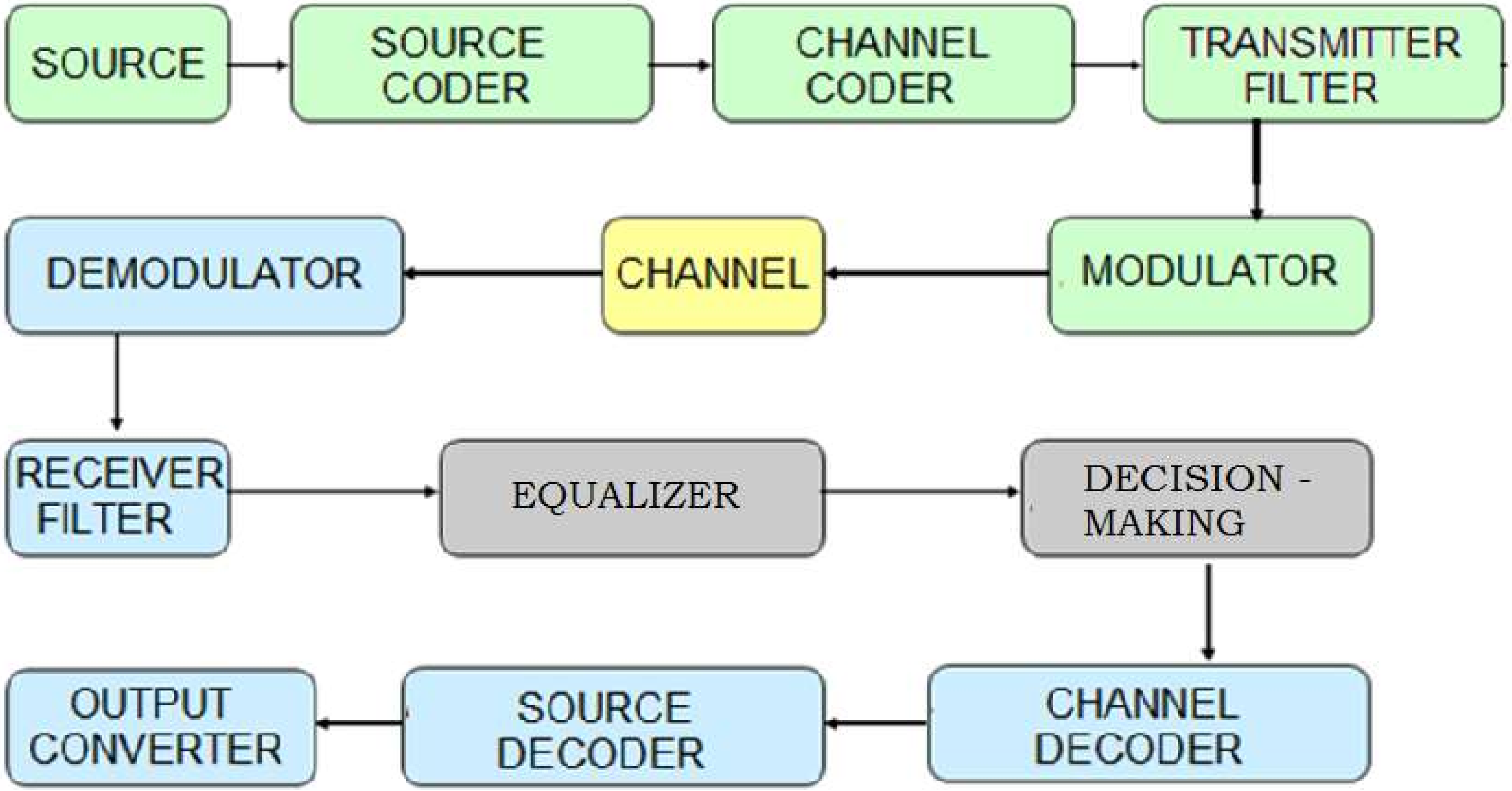
Key Technologies:

Channel Equalization

Presentation outline

- Intersymbol Interference (ISI): definitions, importance
- Equalization: a well-established approach to combat ISI
- Equalization criteria and associated techniques
- Adaptive Equalization
- Linear and non-Linear Equalizers

Block diagram of a Digital Com System



ISI and Equalization

- As already mentioned in previous lectures, the multipath phenomenon appearing in mobile communication networks is the main cause of the so-called Intersymbol Interference (**ISI**).
- The received signal is written as: $y(t) = x(t) * f(t) + n(t)$
- where $f(t)$ is the impulse response of the total discrete channel (i.e., transmitter filter – channel - receiver filter): $f(t) = g_T(t) * c(t) * g_R(t)$
- If the channel $c(t)$ was fixed then proper design of transmitter and receiver filters (e.g., using raised cosine) could theoretically eliminate ISI.
- However, the channel is (ussually) changing, so the above solution is not effective.
- To this end, a special subsystem, called equalizer, is needed at the receiver.
- Equalization, in a broad sense, is any signal processing function that reduces intersymbol interference (i.e., it eliminates the effects of the channel).

ISI and Equalization cont.

- The output of the equalizer has the form of:

$$\hat{d}(t) = x(t) * f(t) * h_{eq}(t) + n_b(t) * h_{eq}(t)$$

- Where $f(t)$ includes the chain of systems:

  Transmitter Filter - Modulator - Channel - Demodulator - Receiver Filter

- **Aim:** design $h_{eq}(t)$, so that the output $d(t)$ of the equalizer tends to $x(t)$.
- In many applications the equalizer must be time-varying to monitor channel changes

Different Equalization approaches

Based on the optimization criterion:

- Maximum Likelihood Criterion - ML: The detection is performed on a symbol by symbol basis so that the probability of correct decision is maximized, given the value of the received signal. They are optimal equalizers, but of high complexity.
- Zero-forcing Criterion - ZF: The equalizer is designed by forcing the ISI to be equal to zero. It's very simple to implement criterion but when noise is present (apart from ISI) it exhibits very poor performance.
- Minimum Mean Square Error Criterion - MMSE: It aims to minimize the mean square error between the equalizer output and the corresponding transmitted sequence. It takes into account both the inter-symbol interference and the added noise.

Different Equalization approaches cont.

Based on the temporal variation of the equalizer:

- Fixed or preset equalizers: The coefficients are calculated once at the beginning of their operation and remain constant for a specified time interval.
- Adaptive equalizers: The coefficients are constantly changing so as to track the time variations of the channel.

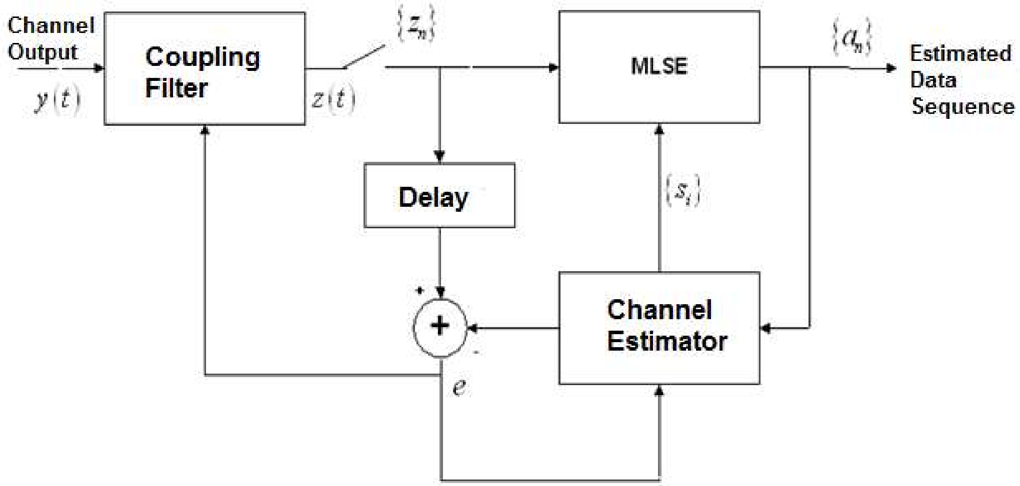
Based on their structure as systems:

- Linear equalizers: The output is a linear function of their input.
- Non-Linear equalizers : The output is not a linear function of their input (e.g., equalizers based on the ML criterion or using NN models).

Maximum Likelihood (ML) criterion

Structure of MLSE equalizer

(Maximum Likelihood Sequence Estimation)



$$max P\{\boldsymbol{a}_m | \boldsymbol{r}\} \equiv max f\{\boldsymbol{r} | \boldsymbol{a}_m\}$$

- Checks all possible data sequences and selects the one with the maximum probability to produce the received signal.
- Requires knowledge of the transmission channel and noise distribution.
- Very high complexity. However, the use of the Viterbi algorithm drastically reduces the computations and allows the application of MLSE in short channels.
- Complexity: from $O(M^N)$ to $O(NM^L)$, where: L the channel length, M the alphabet cardinality and N the number of symbols.

MLSE

- Example: let us consider a binary symbol sequence (e.g., 2-PAM $\alpha_m = +1, -1$) transmitted through a channel of length $L = 2$, $\{f(0), f(1)\}$
- If we ignore noise, then the received value is:

$$y(m) = f(0)a_m + f(1)a_{m-1}$$

- Depending on the symbols sent, we can get the following combinations:

$$y_1(m) = f(0)(+1) + f(1)(+1)$$

$$y_2(m) = f(0)(+1) + f(1)(-1)$$

$$y_3(m) = f(0)(-1) + f(1)(+1)$$

$$y_4(m) = f(0)(-1) + f(1)(-1)$$

MLSE (continued)

- Since:
  - $y(m)$ is known
  - and somehow, we have estimated the channel, that is, we know the $f(0), f(1)$
- Now, we can:
  - calculate all possible $y_i(m)$
  - to see which is closer to the received $y(m)$
  - and decide which symbols were sent
- This process is equivalent to maximizing the cost function: $\log[f(y|a)]$
- Complexity: for sequence of N symbols requires M^N
- We assume that the channel is known or can be estimated
- Note that: The next time (when we receive $y(m+1)$), the a_m symbol is again involved, which is certainly exploitable.

MLSE with Viterbi Algorithm

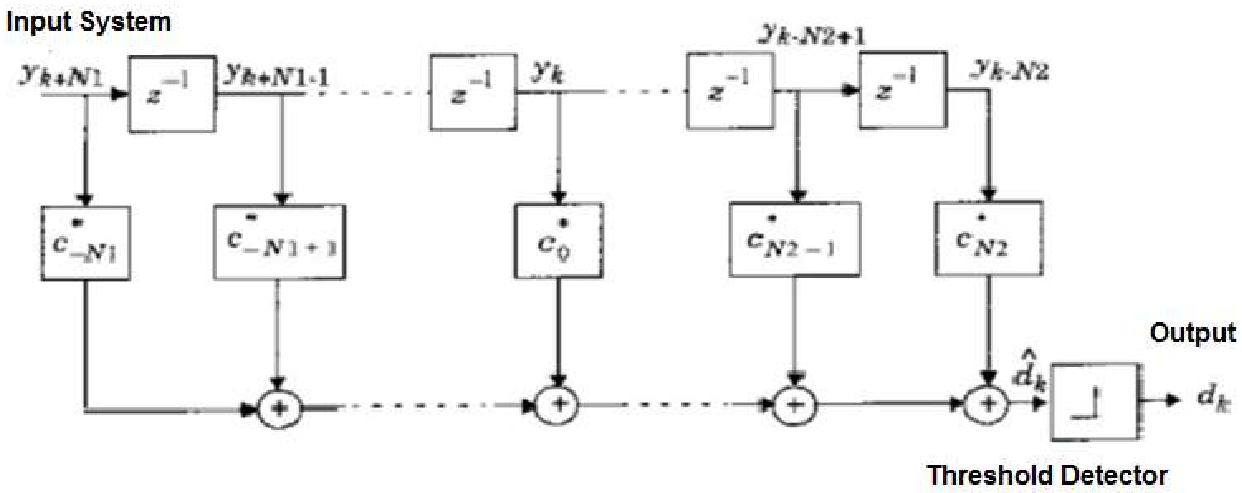
- MLSE: is optimal estimator in case of symbol sequences:
  - Eliminates the ISI
  - What it remains is the added random noise (AWGN)
- Viterbi Algorithm: offers a very efficient implementation of MLSE
- The complexity of the Viterbi algorithm is $\boldsymbol{O}(\boldsymbol{N}\boldsymbol{M}^{\boldsymbol{L}})$, still high compared to other sub-optimal methods
- Due to its exponential complexity, MLSE (via Viterbi) is practically used only in cases
  - of small M, L (e.g., mobile communication systems with relatively low data rates [$M = 2{:}\,4, L = 2{:}\,5$])
- For large M and L, other sub-optimal methods are preferable
- MLSE is used though as a benchmark.

Linear Equalizers

- Instead of MLSE, we can use a linear filter to reduce ISI
- Actually, the purpose of the filter is to “undo” the effects of the channel and the distortions it has caused
- Its parameters are defined based on the characteristics of the channel
- This filter is called channel equalizer or (simply) equalizer

Linear Equalization

Basic structure of a linear transverse equalizer



- The input to the decision device is a linear combination of the inputs of the equalizer in the current and previous time steps, i.e.,

$$\hat{d}_k = \sum_{n=-N_1}^{N_2} (c_n^*) y_{k-n}$$

- the weights c_n are determined by either using the ZF or the MMSE criterion.

Zero Forcing (ZF) Criterion

- The equalizer is designed so as to eliminate inter-symbol interference (required: channel estimation)

$$y(t) = x(t) * h_{ch}(t) + n_b(t)$$

$$\hat{d}(t) = x(t) * h_{ch}(t) * h_{eq}(t) + n_b(t) * h_{eq}(t)$$

$$h_{eq}(t) * h_{ch}(t) = \delta(t) \quad \text{ή} \quad H_{eq}(f) H_{ch}(f) = 1$$

- That is, an infinite length ZF equalizer is actually an inverse filter for the channel system:

$$H_{eq}(f) = \frac{1}{H_{ch}(f)}$$

- Main disadvantage:

It doesn't take into account the added noise which may be considerably amplified at those frequencies where the frequency response of the channel shows large dips.

Minimum Mean Squared Error (MMSE) Criterion (1/3)

- ZF suffers from a main drawback, i.e., it amplifies the added noise in the case of channels with large dips in their frequency response
- One solution to this problem is to relax the "zero forcing" process
- Select the equalizer so that the combined power:
  - of the introduced ISI
  - and the added noise

  at its output to be minimized in the sense of the MMSE criterion
- The resulting equalizer is called Minimum Mean Square Error (MMSE) Equalizer.

Minimum Mean Squared Error (MMSE) Criterion (2/3)

- The MMSE equalizer is designed to minimize the average square error at its output:

$$y(t) = x(t) * h_{ch}(t) + n_b(t)$$

$$\hat{d}(t) = x(t) * h_{ch}(t) * h_{eq}(t) + n_b(t) * h_{eq}(t)$$

$$\min\{E(|e(t)|^2)\} = \min\left\{\left|\hat{d}(t) - x(t)\right|^2\right\}$$

- Advantage: minimizes the sum of both ISI and added noise thus attaining a lower symbol error rate.

Minimum Mean Squared Error (MMSE) Criterion (3/3)

- The MMSE equalizer may be computed as follows:
  - the derivatives of the MSE cost function with respect to $\boldsymbol{c}$ (where $\boldsymbol{c}$ is a vector with the equalizer filter coefficients) are set equal to zero
  - then a linear system is formed, with dimensions $L \times L$, where L the number of equalizer coefficients

$$\boldsymbol{R}_y \boldsymbol{c} = \boldsymbol{r}_{xy}$$

- $L = 2N + 1$ is the number of the received signal samples

$$\boldsymbol{y}_k = [y(kT + N\tau) \quad \cdots \quad y(kT) \quad \cdots \quad y(kT - N\tau)]^T$$

which enter as input to the equalizer filter $\boldsymbol{c}$

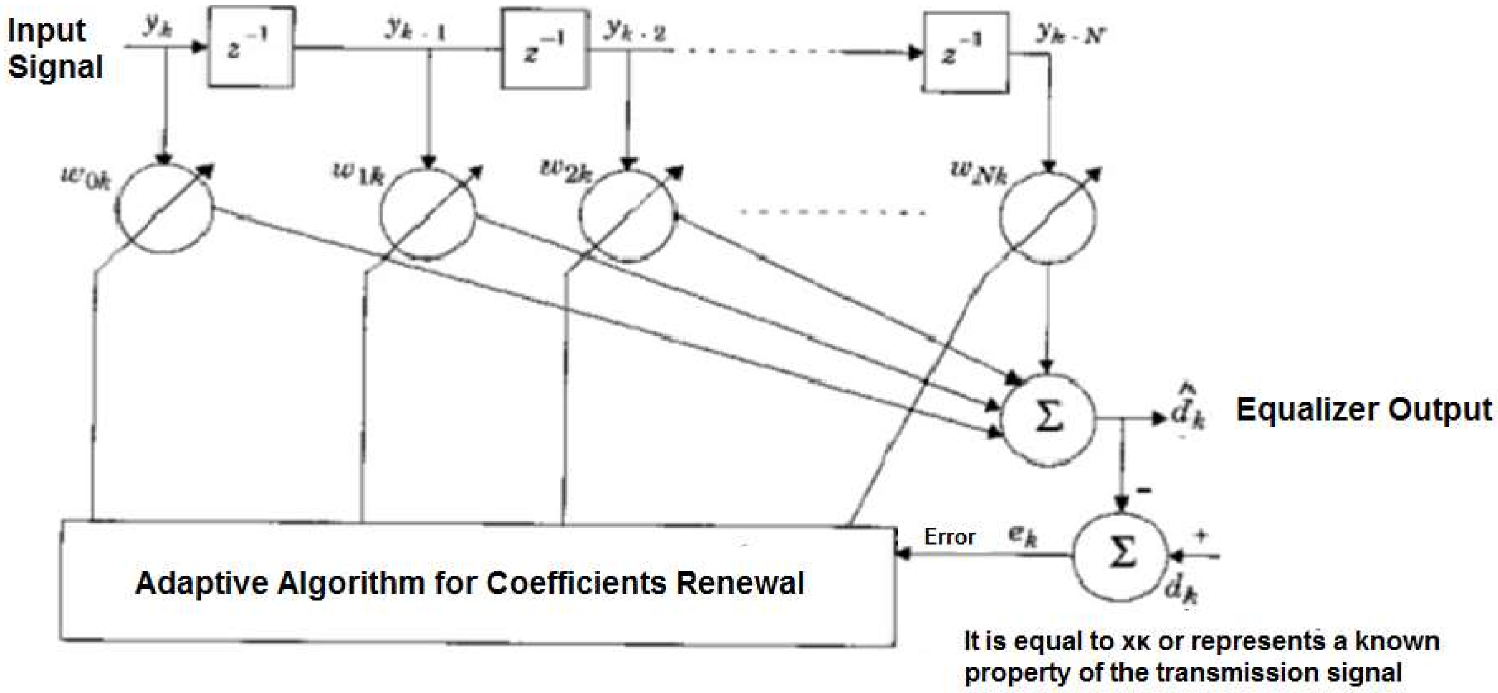
- Since the statistical quantities of autocorrelation and cross-correlation are usually unknown, they are estimated by time averages. To estimate the time average of the quantity $\boldsymbol{r}_{xy}$ we need a number of training symbols.

- Ideally, if the equalizer has infinite length and the noise is AWG:

$$H_{eq}(f) = \frac{1}{H_{ch}(f) + N_0}$$

Adaptive Equalization: Basic structure

Adaptive Linear Equalizer



- When the channel characteristics are time-varying then some kind of adaptive channel equalization would be needed
- There are various equalization algorithms which compute adaptively the filter coefficients. In the following we focus on a very widely used algorithm based on the MMSE criterion.

Adaptive Equalization: main concepts

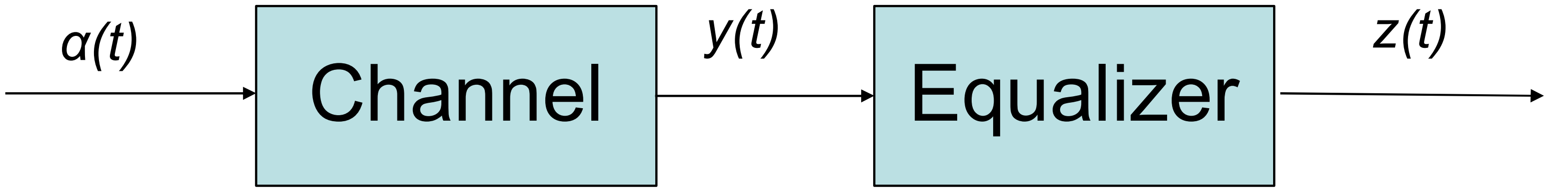
- Usually before the information sequence is transmitted, a training sequence is sent that is used at the receiver's end to initialize the equalizer coefficients.

- Convergence procedure: based on the error signal e_k, the equalizer coefficients are constantly updated, and the minimization function is repeatedly reduced.

- After convergence, the algorithm either "freezes" the coefficients (until it receives a new training sequence) or switches to the so-called decision-directed mode (where the symbol decisions are used to form the equalization error that drives the algorithm).

- Blind adaptive algorithms: a class of algorithms that utilize the statistical characteristics of the transmitted signal and do not require a training sequence.

From fixed to adaptive equalization



- Therefore, to calculate the coefficients of the MMSE equalizer, we need to solve an $L\times L$ linear system $\boldsymbol{R}_y\boldsymbol{c} = \boldsymbol{r}_{\boldsymbol{ay}}$
- The solution of the system is

$$\boldsymbol{c}_{opt} = \boldsymbol{R}_y^{-1}\boldsymbol{r}_{\boldsymbol{ay}}$$

- Often in practical equalizer applications:
  - » we try to avoid the highly complex inversion of $\boldsymbol{R}_y$
  - » and we apply an iterative procedure for computing filter $\boldsymbol{c}$
- As we will see later on, the iterative procedure concept will eventually lead to adaptive equalizers
- Adaptive equalizers based on the ZF criterion can be designed in a similar way

Iterative solution of the linear system:

- We start from an arbitrary point $\boldsymbol{c}_0$ of the coefficient vector lying on the surface of the cost function.
  - Since we are using MMSE criterion we have a 2nd degree (quadratic) surface in the $(2N + 1)$-dimensional space
- At each iteration k, we calculate the derivative of the cost function in terms of factors
  - the derivative of MSE cost function is:

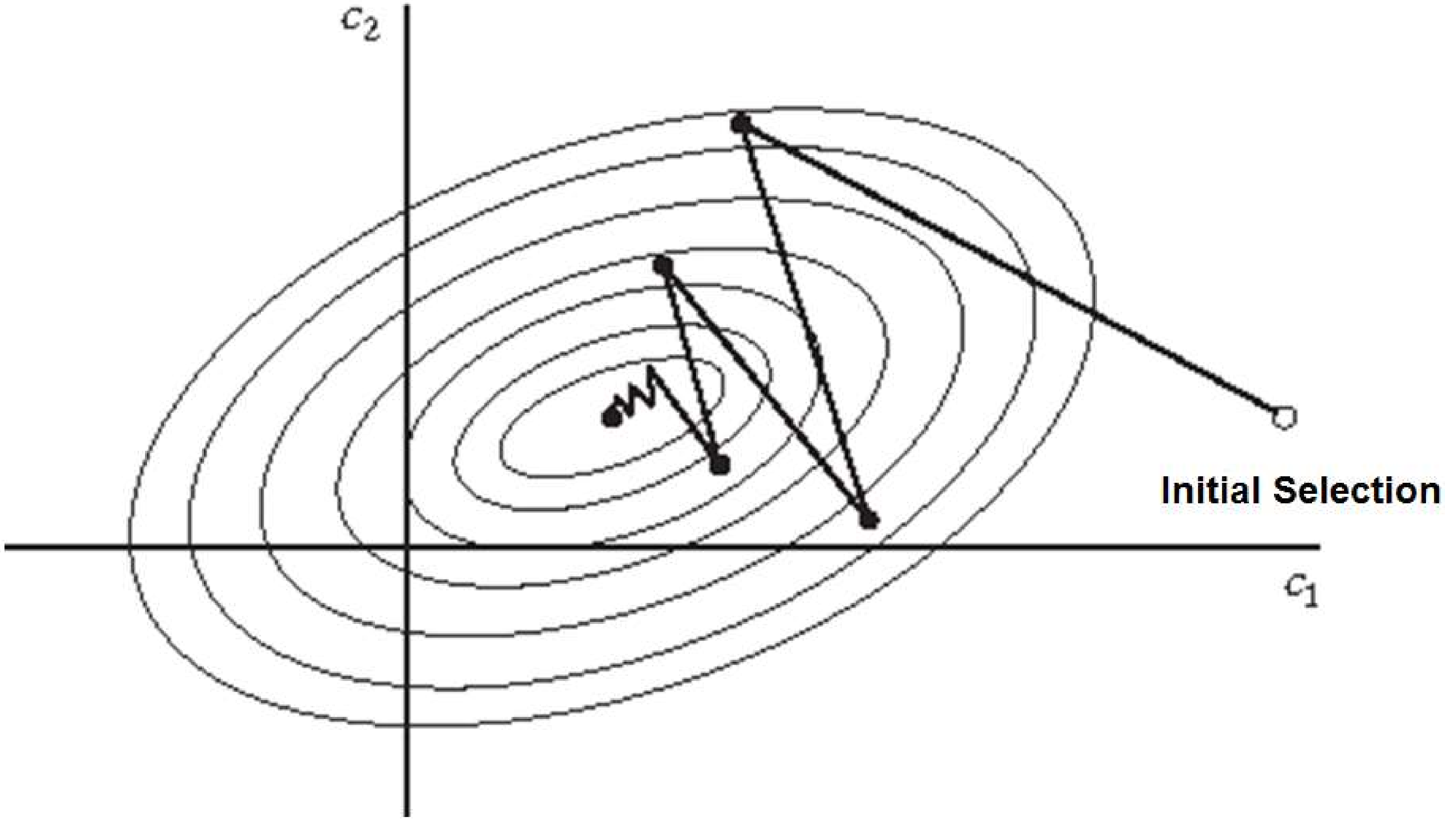
$$\boldsymbol{g}_k = \boldsymbol{R}_y \boldsymbol{c}_k - \boldsymbol{r}_{ay}$$

- Then the filter vector $\boldsymbol{c}_k$ will changes in a direction opposite to the that of the gradient vector

$$\boldsymbol{c}_{k+1} = \boldsymbol{c}_k - \Delta \boldsymbol{g}_k$$

where Δ, is the parameter «step-size»

Steepest descent convergence



Convergence of steepest descent algorithm
on the surface of the cost function
(Two-Dimensional Space)

Steepest descent method

- To attain convergence, parameter Δ should take a small positive value (within the convergence region)
- If $k \to \infty$, then
  - $\boldsymbol{g}_k \to 0$ (the gradient vector tends to zero)
  - $\boldsymbol{c}_k \to \boldsymbol{c}_0$ (coefficients tend to be optimal)
- Convergence to the optimal value $\boldsymbol{c}_0$ requires theoretically an infinite number of iterations
- The quantities $\boldsymbol{R}_{\boldsymbol{y}}$ and $\boldsymbol{r}_{\boldsymbol{ay}}$ are calculated once at the start of the adaptive process (by taking into account all available data) and then the iterations begin.
- In practice, the optimal solution is closely approached within a few hundreds of iterations
- The complexity is $O(N^2)$ per iteration

Adaptive equalization based on MMSE criterion

- If the channel is time variant,
- Then,
  - the introduced ISI is time-varying
  - the optimal solution $\boldsymbol{c}_0$ moves in the L-D space
- The aim now is to solve adaptively the *LxL* linear system
  $\boldsymbol{R}_y\boldsymbol{c} = \boldsymbol{r}_{\boldsymbol{ay}}$
- An estimate of the gradient vector can be used to "correct" the equalizer

$$\hat{\boldsymbol{c}}_{k+1} = \hat{\boldsymbol{c}}_k - \Delta\hat{\boldsymbol{g}}_k$$

- For the MMSE criterion, it applies

$$\boldsymbol{g}_k = -E[e_k\boldsymbol{y}_k]$$

- A simple estimate of $\boldsymbol{g}_k$ is its instantaneous value

$$\hat{\boldsymbol{g}}_k = -e_k\boldsymbol{y}_k$$

LMS Equalization Algorithm

- Employing the instantaneous estimation of $\boldsymbol{g}_k$ results in the Least Mean Squares – LMS adaptive algorithm
- The LMS is based on the MSE minimization criterion
- Since we use a gradient vector estimate, it is also known as stochastic gradient algorithm
- Each time a new sample is received, we have an iteration step (i.e. iteration step coincides with time step)
- At time step (iteration) k, the equalizer filter has as input the following $2N+1$ received samples

$$\boldsymbol{y}_k = [y(kT+N\tau) \quad \cdots \quad y(kT) \quad \cdots \quad y(kT-N\tau)]^T$$

LMS Equalization Algorithm cont.

$$\boldsymbol{c}_0 = [0 \quad \cdots \quad 0 \quad 1 \quad 0 \quad \cdots \quad 0]^T$$

for $k = 0,1,\ldots$

$$z_k = \boldsymbol{c}_k^H \boldsymbol{y}_k$$

$$e_k = \tilde{a}_k - z_k$$

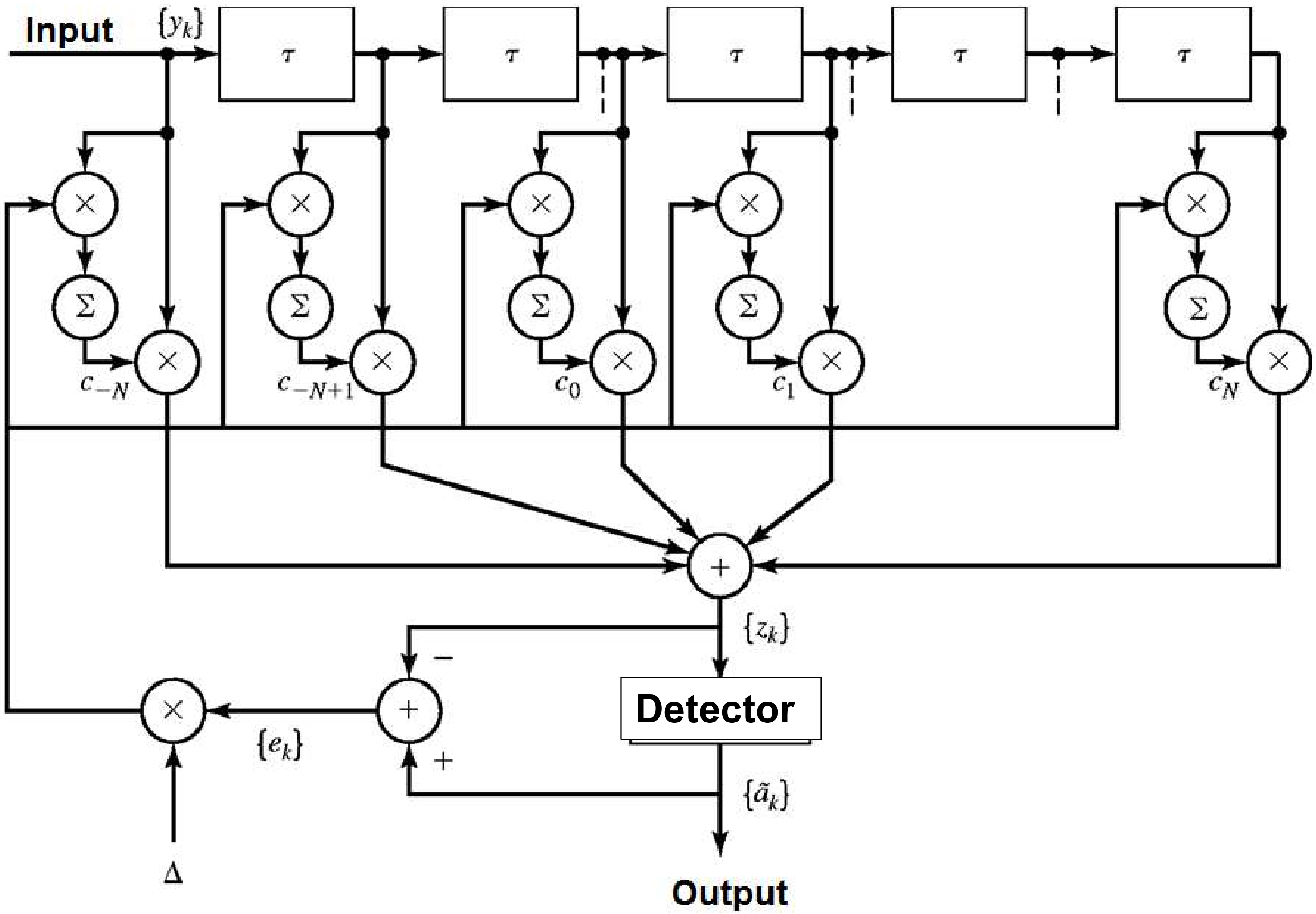
$$\boldsymbol{c}_{k+1} = \boldsymbol{c}_k + \Delta \boldsymbol{y}_k e_k^*$$

- Training Mode
  - first, a training sequence, i.e., a sequence of known symbols, is sent by the transmitter
- Decision-directed mode
  - then the information data are transmitted
  - the decisions of the equalizers are assumed correct and used as desirable symbols

$$\tilde{a}_k = Q(z_k)$$

$$e_k = \tilde{a}_k - z_k$$

LMS Equalization Algorithm cont.



Linear adaptive equalizer based on the MMSE criterion

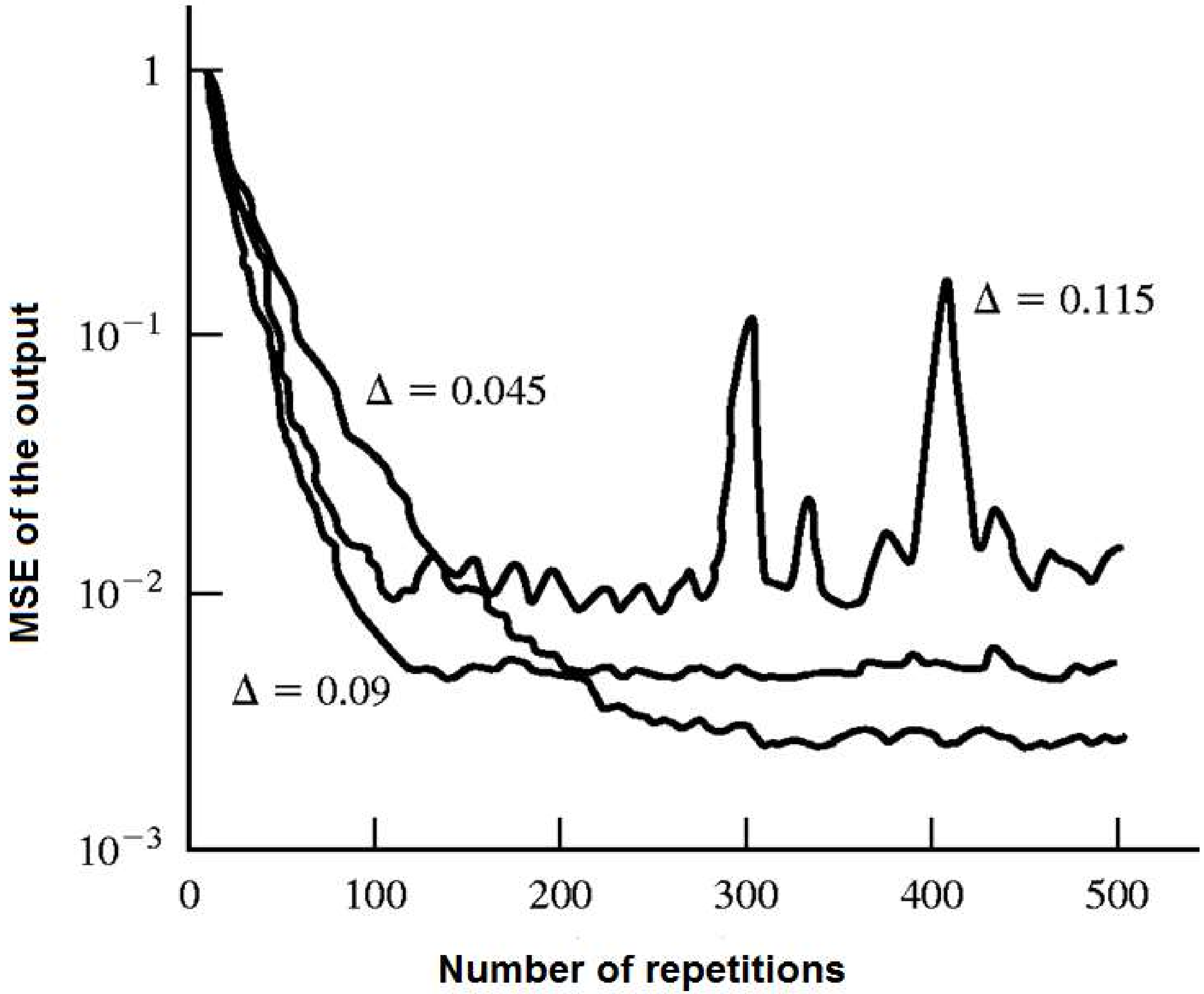
LMS Algorithm: Step size selection

- The involved step size plays a very important role in the LMS algorithm's operation and performance
- If Δ is small, then the algorithm:
  - converges slowly,
  - but converges closer to the optimum value.
- If Δ is large, then the algorithm:
  - converges faster,
  - but away from the optimal value.
- If Δ is too large, then the algorithm diverges.
- An empirical rule for acceptable convergence and good tracking in slow variant channels is

$$0 < \Delta < \frac{2}{\sum_{k=1}^{2N+1} \lambda_k(R)} \quad \text{ή} \quad \Delta = \frac{1}{5(2N+1)P_R}$$

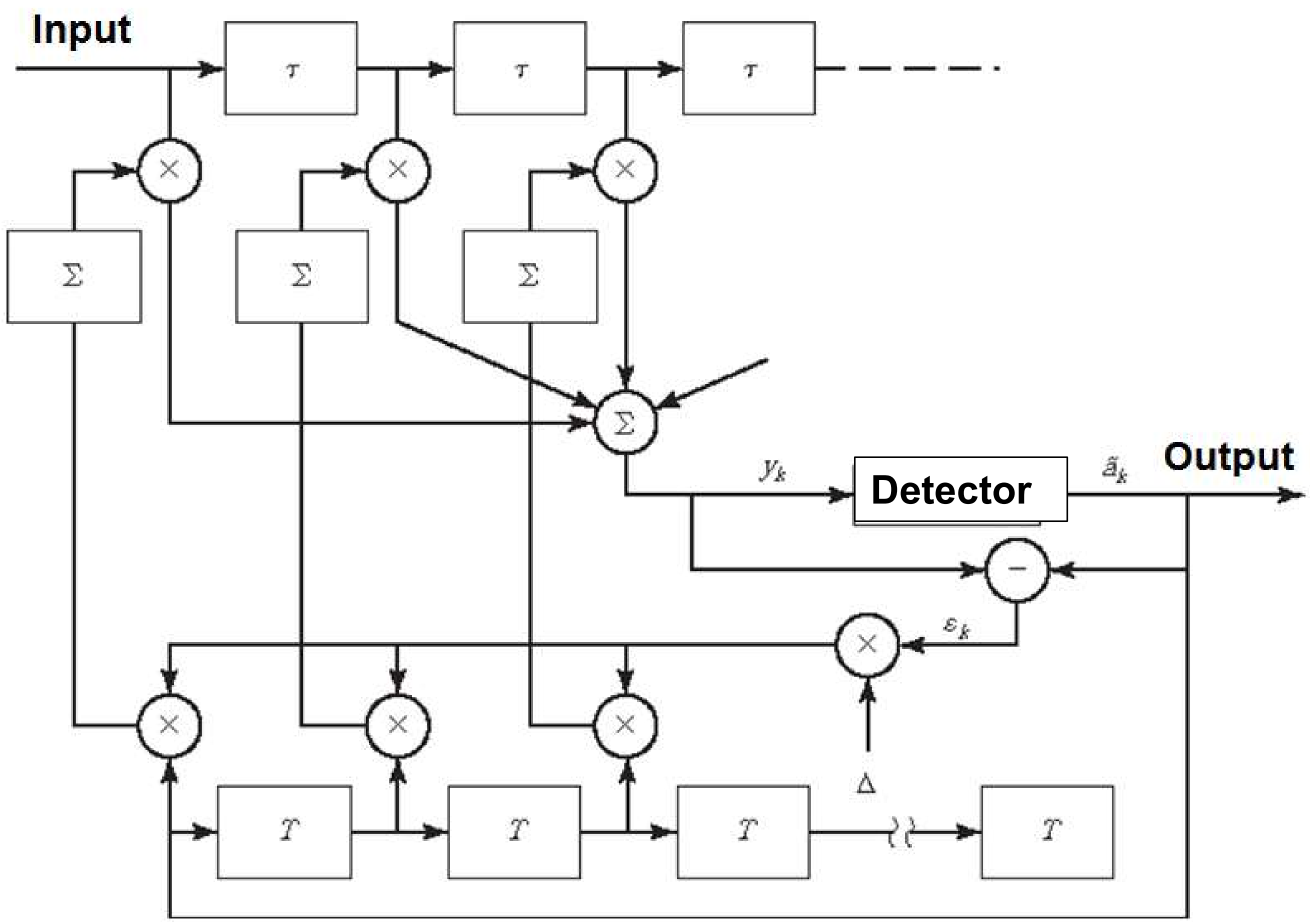
where P_R is the power of the input signal

LMS convergence: examples



Convergence of the LMS algorithm
for different step sizes

Adaptive ZF Equalizer (OXI)



Zero forcing adaptive equalizer:

- a ZF equalizer can be adaptively implemented in a similar way as the MMSE equalizer

Criteria for evaluating/selecting adaptive algorithms

- Steady-state error
- Convergence speed
- Tracking properties
- Complexity
- Processing delay
- Robustness (related to stability)
- Other numerical properties (such as solution accuracy)
- Versatility regarding efficient implementations (DSP, parallel, pipeline)

Criteria for evaluating/selecting adaptive algorithms cont.

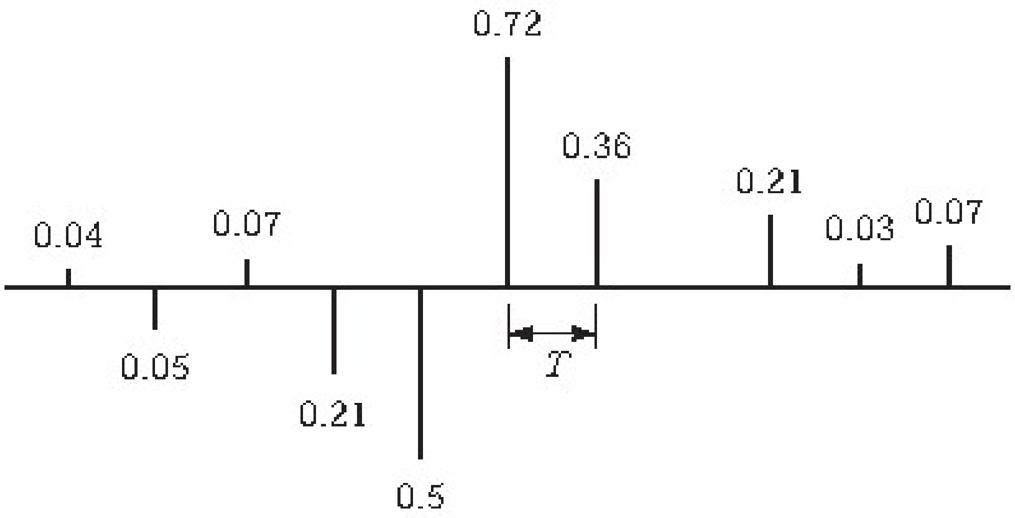
Further criteria with great importance in mobile communications:

- Cost of computing platform.
- Power consumption.
- Data rate and movement speed (affect the characteristics of the channel, thus determining the specifications of the equalizer).
- Maximum expected time spread of the channel (dictates the required number of equalizer coefficients, thus affecting its cost, processing time, etc.).

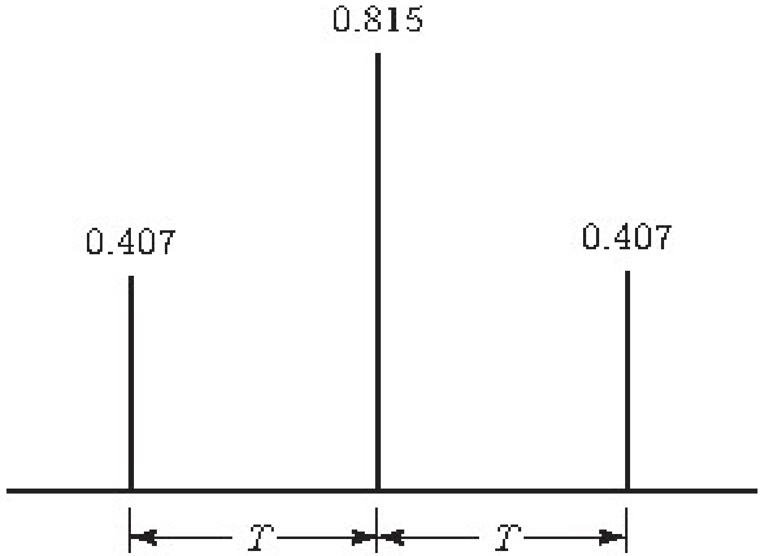
Intensity of ISI

- The intensity of the ISI is related
  - mainly with the spectral characteristics of the channel
  - and secondarily with the extent of the ISI (channel length)
- When the channel frequency response has large spectral dips, then severe ISI is introduced
- In these cases, the linear equalizers try to apply the inverse filter, thus significantly amplifying the noise in this area (even when we have an MMSE criterion)
- Conclusion: linear equalizers may not be suitable for channels with large spectral dips
- Some typical examples follow

Impulse responses of two example channels

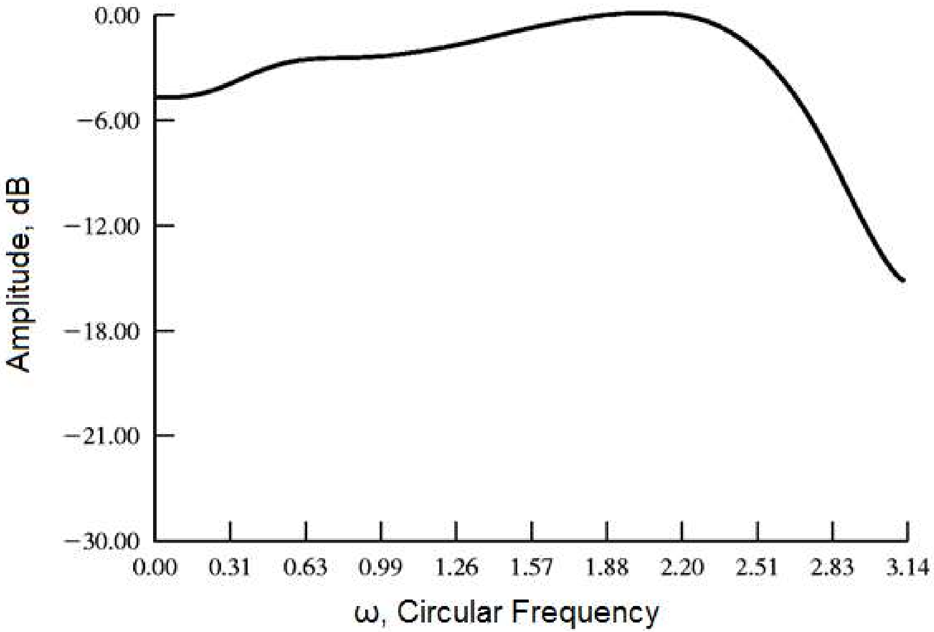


Channel A

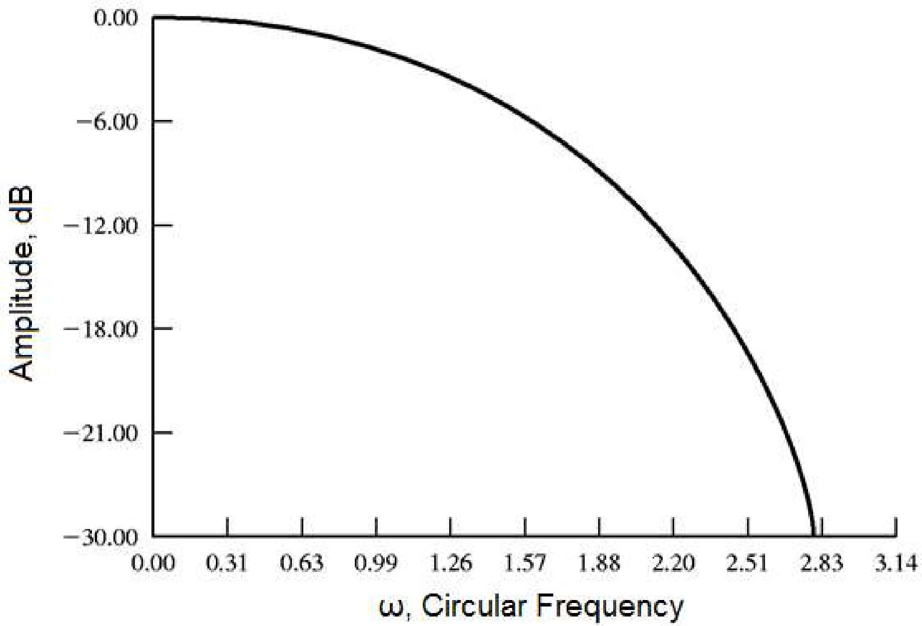


Channel B

Frequency responses of the example channels

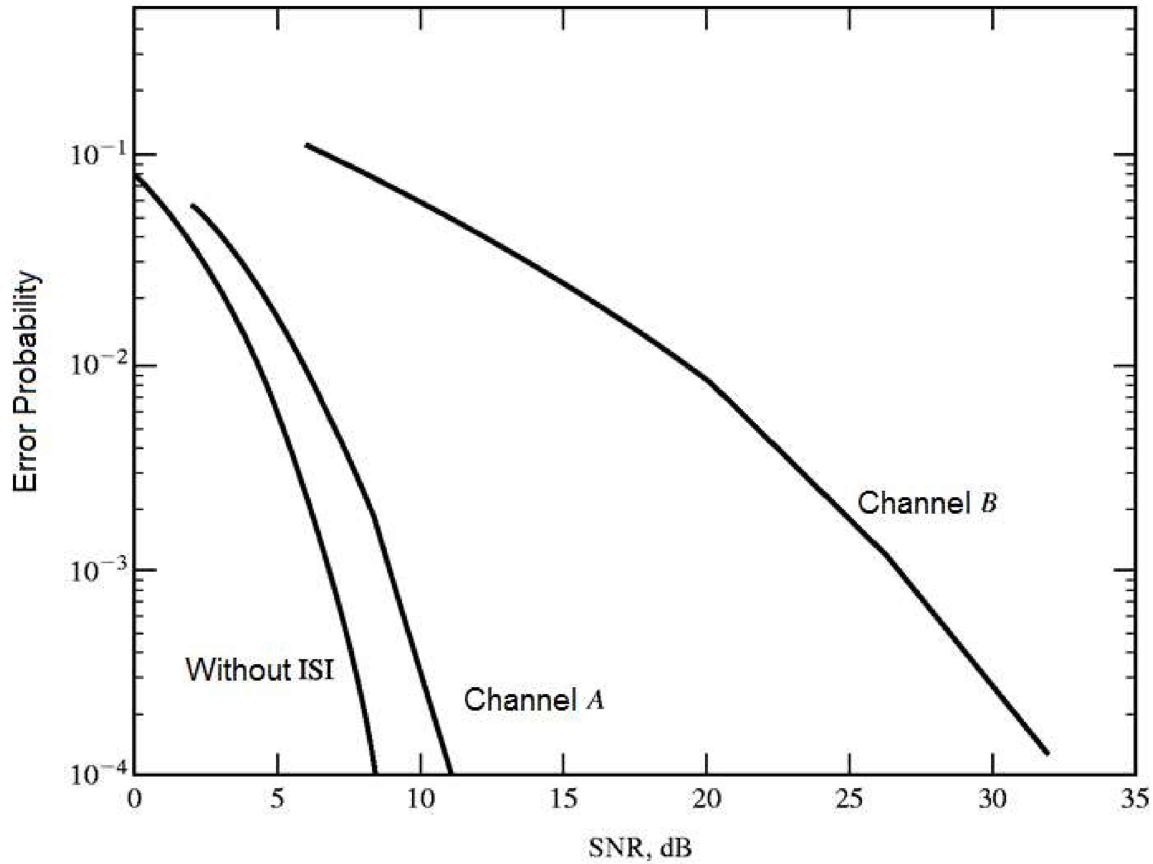


Channel A



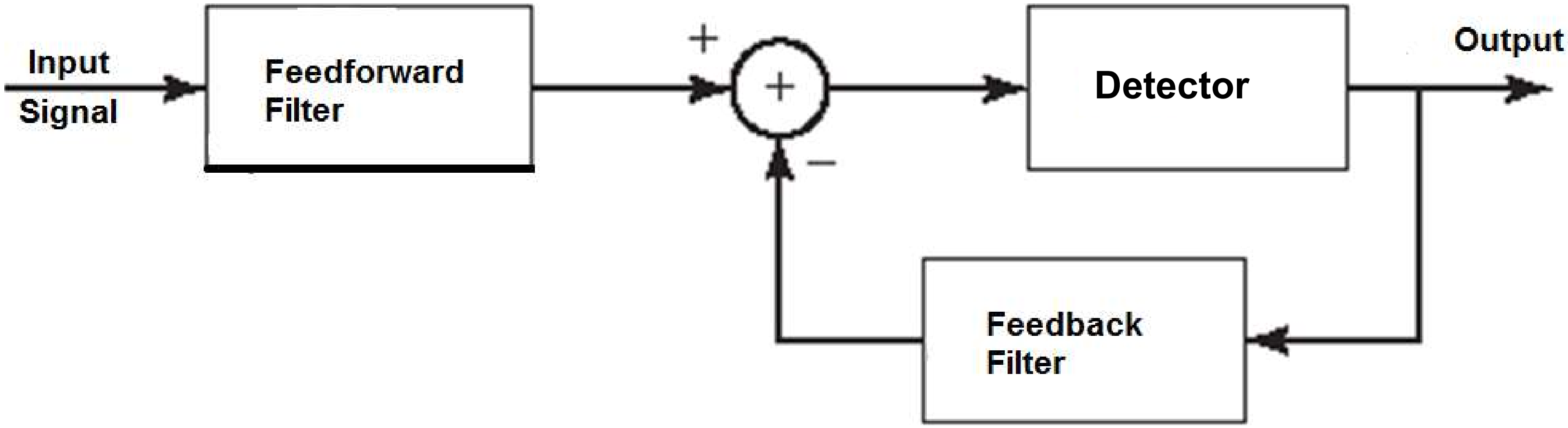
Channel B

Channel A & B: MMSE equalizer performance



Error Rate Performance (BER) of Linear MMSE Equalizer

DFE Equalizer (1/2)



- DFE: Decision Feedback Equalizer
  - Non-Linear Equalizer structure
  - The feed-forward filter in series with the preceding overall discrete channel (TX Filter – Channel – RX Filter) constitutes a causal system (that is, a system in which the ISI comes from previous symbols only)
  - The feedback filter uses the Detector's decisions for previous symbols in order to eliminate the (causal) ISI that they inserted in the current symbol
  - DFE generally performs better than linear equalizers.

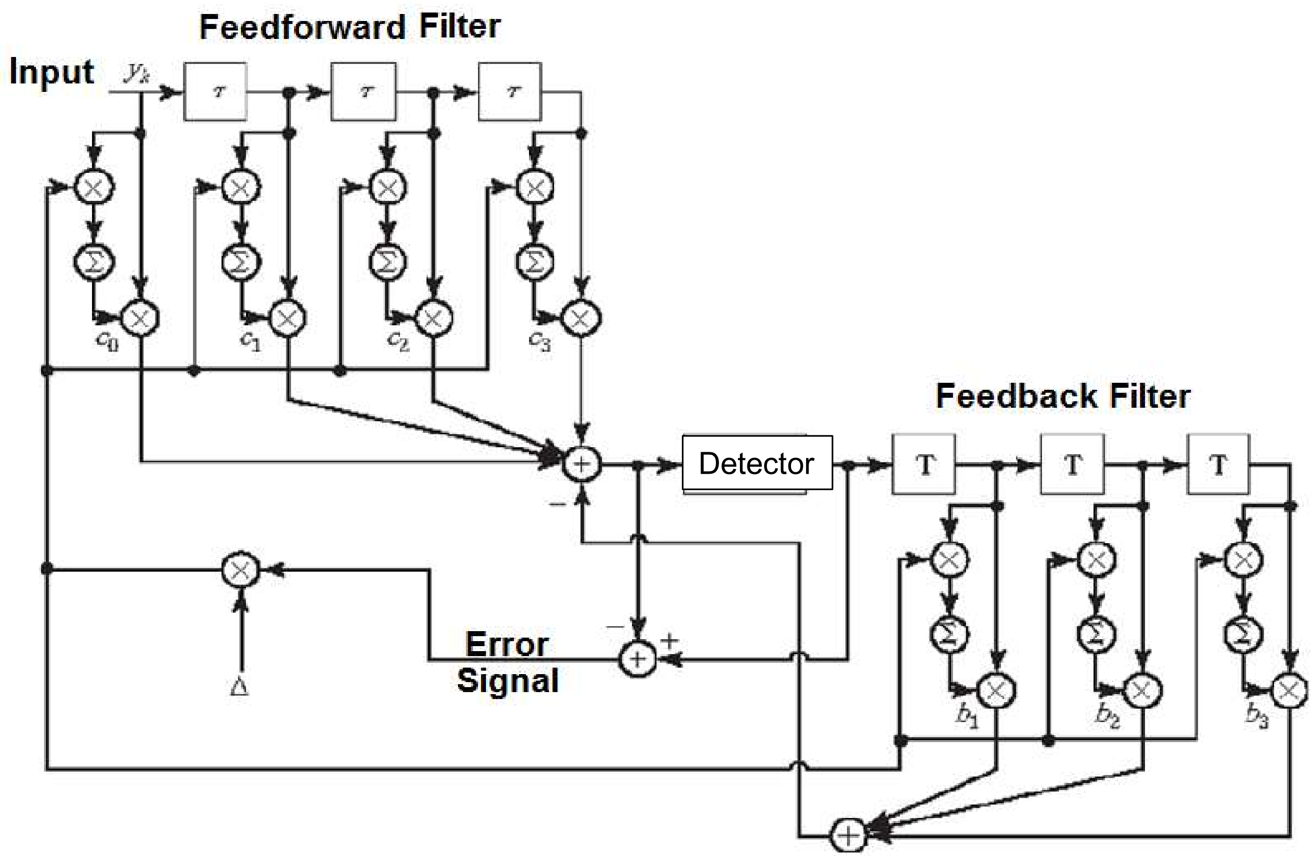
DFE Equalizer (2/2)

- Feedforward Filter, FF
  - usually fractionally-spaced
  - adaptive or fixed
  - equivalent to linear equalizers
  - N_1 coefficients, $\{c_n\}$
- Feedback filter, FB
  - symbol-spaced
  - adaptive or fixed
  - its input are the decisions made by the detector
  - N_2 coefficients, $\{b_n\}$
- **DFE Output**

$$z_m = \sum_{n=1}^{N_1} c_n y\big((m+N_1)T - nT\big) + \sum_{n=1}^{N_2} b_n \tilde{a}_{m-n} = \boldsymbol{c}^H \boldsymbol{y}_m + \boldsymbol{b}^H \tilde{\boldsymbol{a}}_m,$$

where $\tilde{\alpha}_m = Q(z_m)$

DFE Adaptive Equalizer (1/2)



- The MMSE criterion is commonly used
- and some stochastic algorithm (e.g., LMS)

DFE Adaptive Equalizer DFE (2/2)

- The LMS algorithm in the case of the DFE equalizer

$$\boldsymbol{c}_0 = [0 \quad \cdots \quad 0 \quad 1]^T$$

for $m = 0,1,\ldots$

$$z_m = \boldsymbol{c}^H \boldsymbol{y}_m + \boldsymbol{b}^H \tilde{\boldsymbol{a}}_m$$

$$e_k = Q(z_m) - z_m$$

$$\begin{bmatrix} \boldsymbol{c}_{m+1} \\ \boldsymbol{b}_{m+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{c}_m \\ \boldsymbol{b}_m \end{bmatrix} + \Delta e_m^* \begin{bmatrix} \boldsymbol{y}_m \\ \tilde{\boldsymbol{a}}_m \end{bmatrix}$$

Error Propagation of DFE Equalizer

- If the previous decisions are correct and the length of the feedback filter is long enough, complete elimination of the introduced ISI is achieved.

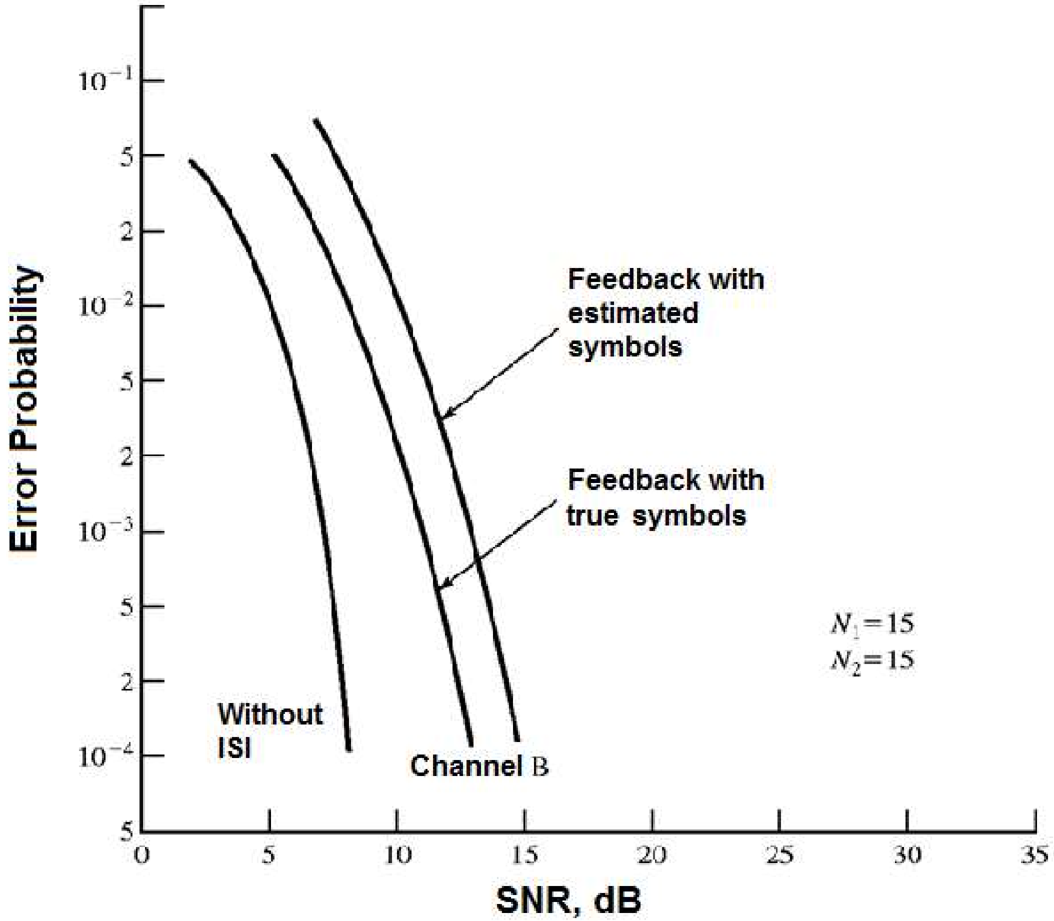
- However, if the detector decides incorrectly, then the erroneous symbol is fed back through the FB filer and affects the detection of subsequent symbols

- That is, due to the FB filter, any decision error is spread out to future decisions and may cause new errors

- It turns out that the effect of this "error propagation" phenomenon is not usually catastrophic

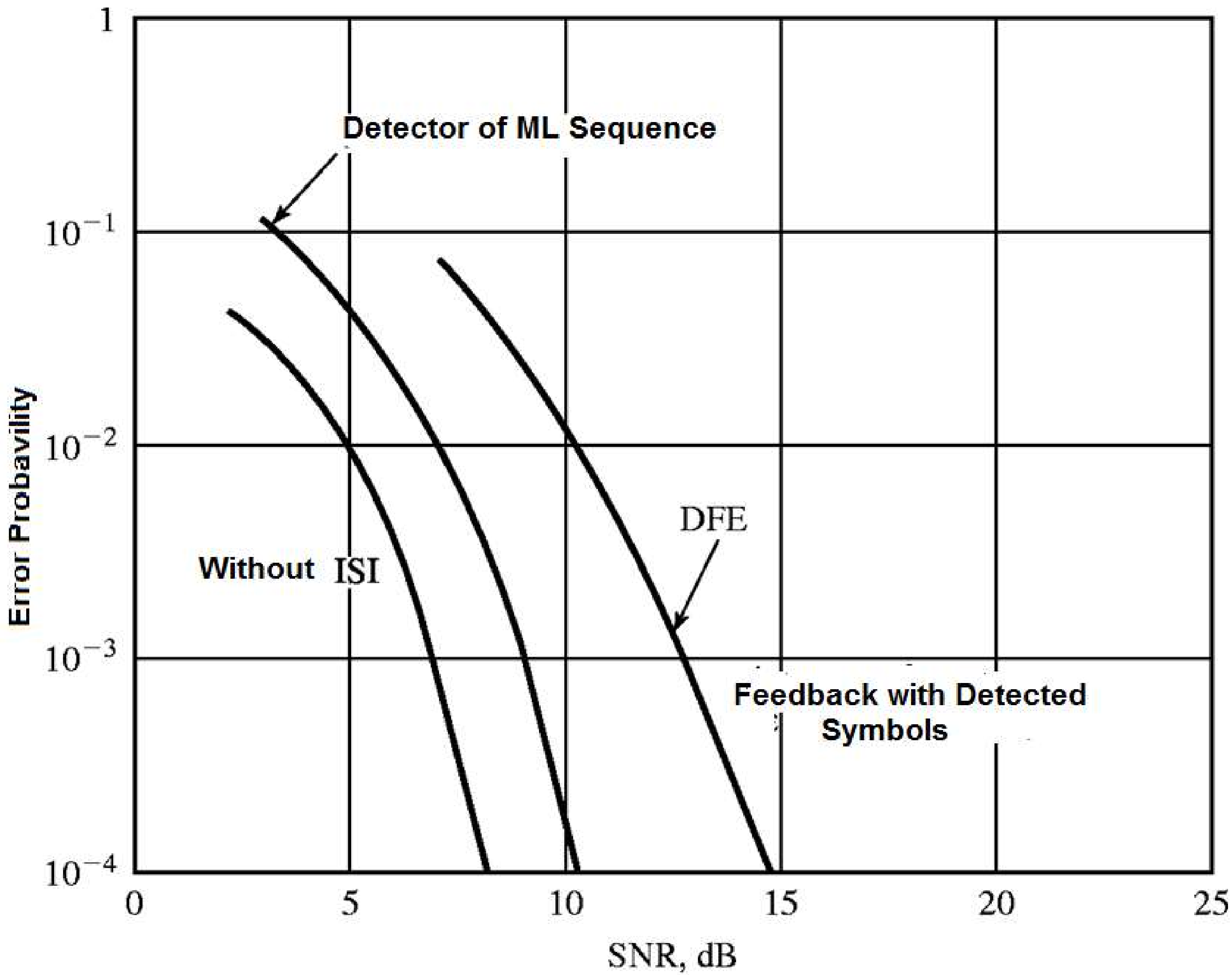
- On average it causes a $1-2\ dB$ loss of performance for $BER < 10^{-2}$

Example: Error propagation and DFE equalizer performance



DFE performance with and without error propagation for channel B and $N_1 = N_2 = 15$

Comparison of Viterbi and DFE (Proakis, Channel B)



Example of another adaptive equalization algorithm

RLS Algorithm (Recursive Least Squares)

- Minimizes the time average of the error:

$$J(n) = \sum_{i=1}^{n} \lambda^{n-i} e^{*}(i,n) e(i,n)$$

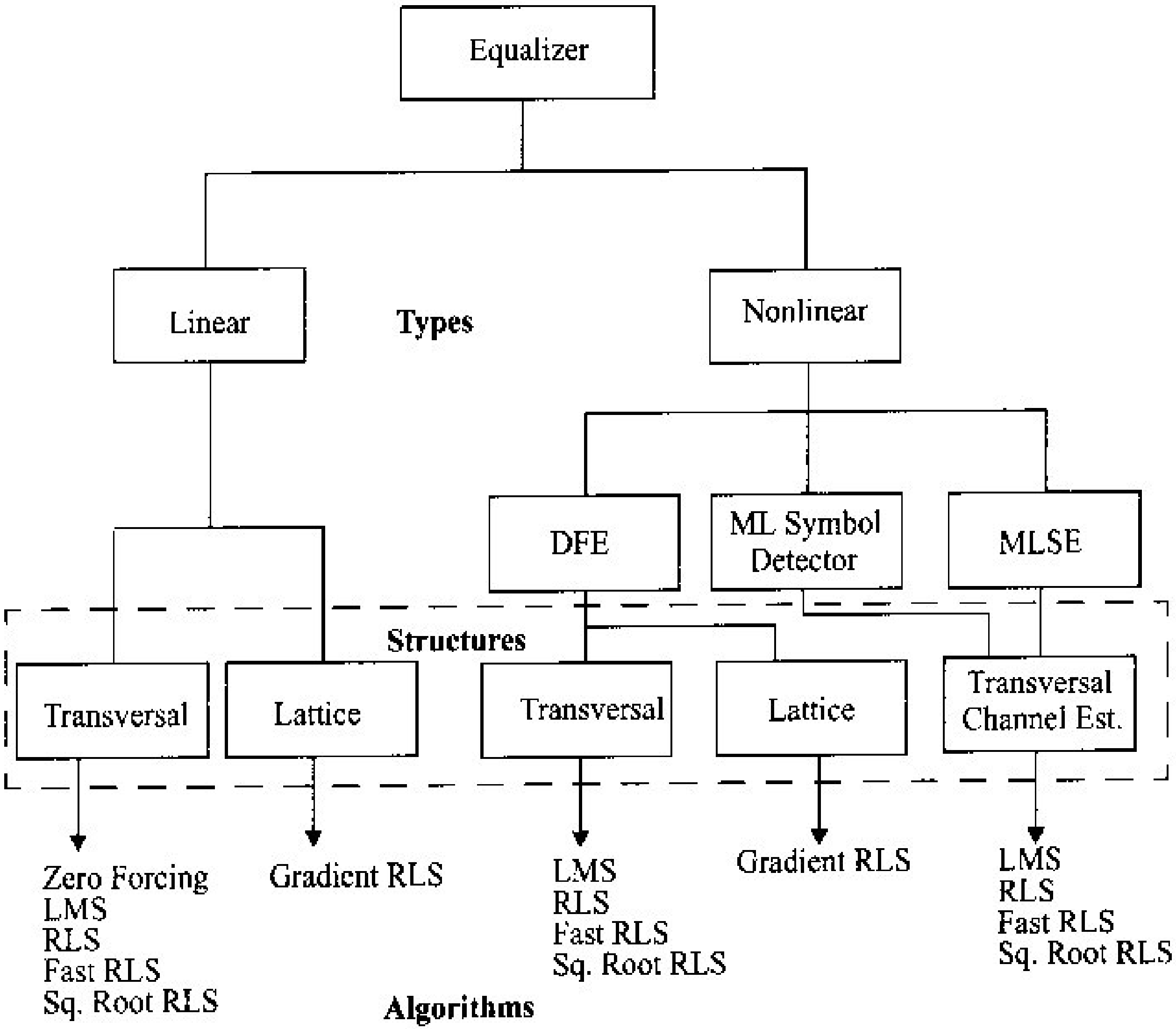
$$\hat{d}(n) = \boldsymbol{w}^{T}(n-1)\boldsymbol{y}(n), \quad e(n) = x(n) - \hat{d}(n)$$

$$\boldsymbol{k}(n) = \frac{\boldsymbol{R}^{-1}(n-1)\boldsymbol{y}(n)}{\lambda + \boldsymbol{y}^{T}(n)\boldsymbol{R}^{-1}(n-1)\boldsymbol{y}(n)}, \boldsymbol{R}^{-1}(n) = \frac{1}{\lambda}[\boldsymbol{R}^{-1}(n-1) - \boldsymbol{k}(n)\boldsymbol{y}^{T}(n)\boldsymbol{R}^{-1}(n-1)]$$

$$\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \boldsymbol{k}(n)e^{*}(n)$$

- Parameter λ determines the ability to monitor changes.
- The convergence rate is determined by Table $\boldsymbol{R}$.
- Faster convergence than LMS, but also greater complexity $(2.5N^{2} + 4.5N)$

Summary of equalization techniques



Special topics

- Equalization of Non-Linear Channels:

$$y(t) = NL\{a_n\}$$, where NL non-linear operator

e.g., such nonlinearities occur due to non-linear amplification

Possible treatment methods:

- Non-Linear Models (Volterra Series Expansion)
- Neural Networks (Highly Non-Linear Mapping)
- MLSE (Viterbi) (requires non-linear channel estimation)

- Blind (and Semi-Blind) Equalization:

- Equalization without use (or with minimal use) of training sequence.

- Interference management in MIMO and distributed MIMO (modern research field)