Revision of Lecture 4

- We have discussed all basic components of MODEM
  - Pulse shaping Tx/Rx filter pair
  - Modulator/demodulator
  - Bits $\overset{map}{\leftrightarrow}$ symbols
- Discussions assume ideal AWGN channel, i.e. channel is non-dispersive (no memory)
- Dispersive channel causes ISI, and results no longer valid

MODEM components

pulse shaping Tx/Rx filter pair

modulator/demodulator

bits $\overset{map}{\leftrightarrow}$ symbols

equalisation (distorting channel)

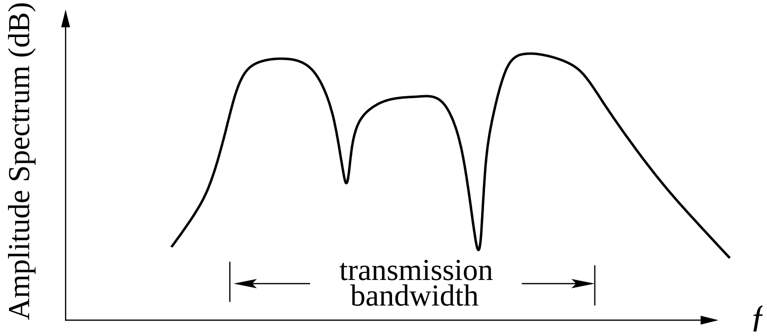
bit error rate and other issues

The problem: the combined impulse response of Tx filter, channel and Rx filter will lose desired property of regular zero crossings at symbol spacing

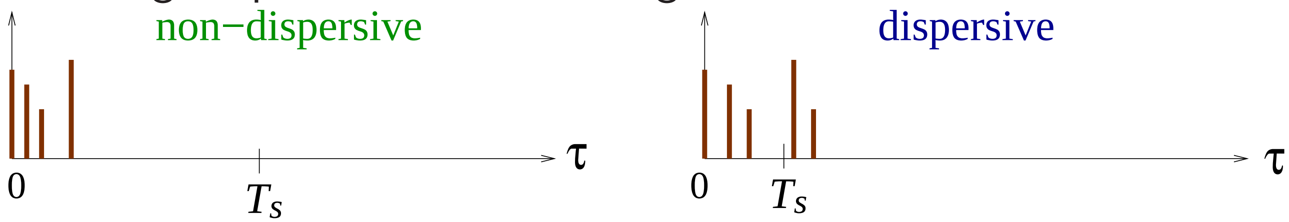
This lecture: **equalisation**

Dispersive Channel

- Channel for communication is at RF passband, but we consider its equivalent baseband channel
  – We design Tx and Rx filter pair $G_{Rx}(f)G_{Tx}(f)$ to be a Nyquist system, whose impulse response has regular zero crossings at symbol-rate spacing
  – If channel $G_c(f)$ is non-ideal, the combined $G_{Rx}(f)G_c(f)G_{Tx}(f)$ is not a Nyquist system, causing intersymbol interference
- Non-ideal channel has memory, i.e. is dispersive, which can be caused by
  1. Restricted bandwidth, i.e. channel bandwidth is insufficient for the required transmission rate



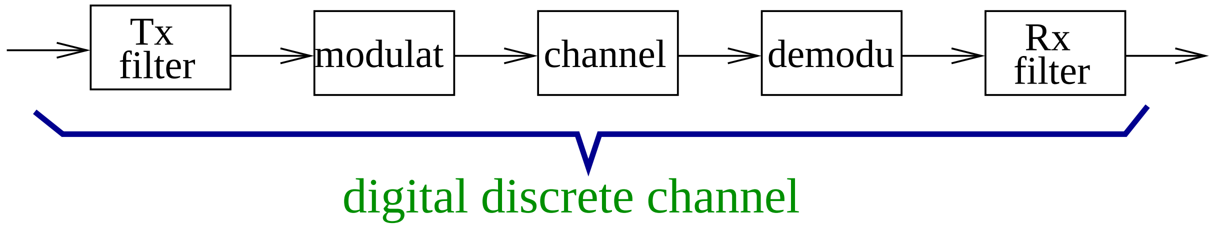
  2. Multipath distorting: copies of transmitted signal arrive at receiver with different excess delays

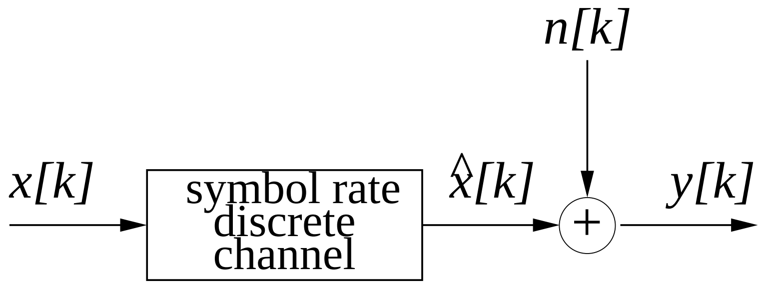


  – If excess delay is small compared with symbol period T_s, channel is non-dispersive, i.e. ideal
  – If excess delay is big small compared with T_s, channel is dispersive, i.e. having memory

Discrete Channel Model

- Recall slide **16**, examine the **combined channel model** between $x[k]$ and $\hat{x}[k]$:





- If physical transmission channel is ideal, $y[k]$ is a noise corrupted delayed $x[k]$:

$$y[k] = x[k - k_d] + n[k]$$

- If physical channel is **dispersive** (note ISI):

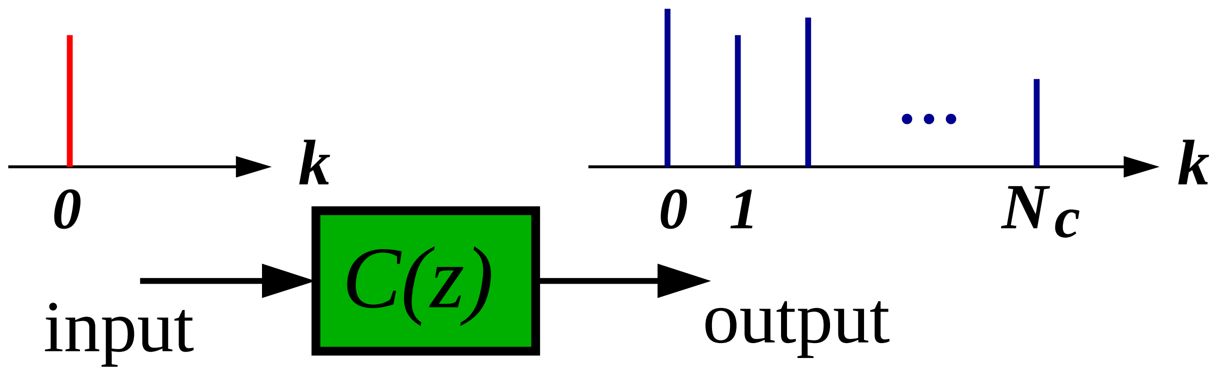
$$y[k] = \sum_{i=0}^{N_c} c_i \cdot x[k - i] + n[k]$$

$\{c_i\}$ are the channel impulse response (CIR) taps, and N_c the length of CIR

Channel Impulse Response

- Continuous-time signal/system $\rightarrow$ Fourier transform
- Discrete-time signal/system $\rightarrow$ z-transform
- Discrete channel with channel impulse response $\{c_0, c_1, \cdots, c_{N_c}\}$

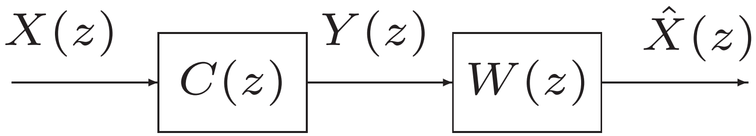
$$C(z) = \sum_{i=0}^{N_c} c_i z^{-i}$$



- In practice, real signal/system are real-valued, but we can use equivalent baseband signal/system (as in QAM system) which are complex-valued

Equalisation — Solution

- The system $C(z)$ is the z-transform of the discrete baseband channel model (including Tx and Rx filters, modulation, physical transmission channel, demodulation, and sampling)
- If the channel has severe amplitude and phase distortion, equalisation is required:



- We want to find an equalisation filter $W(z)$ such that the recovered symbols $\hat{X}(z)$ are only delayed versions of the transmitted signal, $\hat{X}(z) = z^{-k_d} \cdot X(z)$
- The optimal solution for the noise-free case is (zero-forcing equalisation):

$$W(z) \cdot C(z) = z^{-k_d} \qquad \text{or} \qquad W(z) = z^{-k_d} \cdot C^{-1}(z)$$

  – Since $C(z) = \sum_{i=0}^{N_C} c_i z^{-i}$ is a finite-duration impulse response (FIR) filter, $z^{-k_d} \cdot C^{-1}(z)$ is an infinite-duration impulse response (IIR) filter

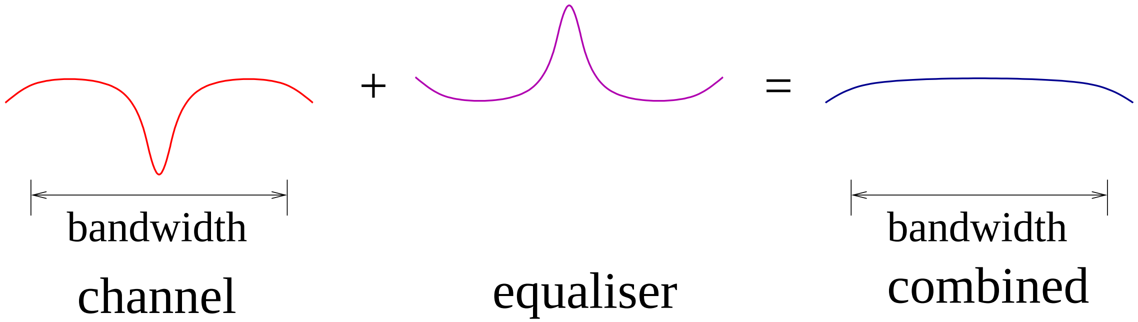
  – In practice we can only truncate $W(z)$ to a sufficiently long but finite-duration filter

$$W(z) = \sum_{i=0}^{N_e} w_i z^{-i} \approx z^{-k_d} \cdot C^{-1}(z)$$

- Another popular optimal equalisation solution is called minimum mean square error (MMSE) solution

Equalisation — Issues

- Equaliser: aims to make the combined channel/equaliser a Nyquist system again
  - Zero-forcing equalisation will completely remove ISI



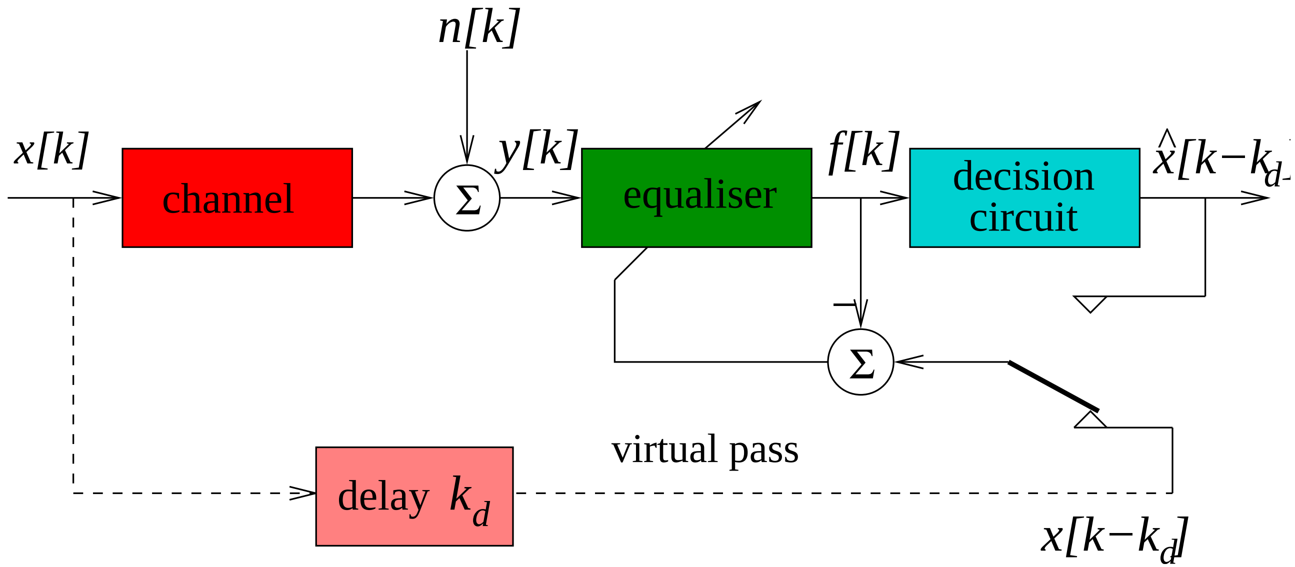
  - But the noise is amplified by the equaliser, and in high noisy condition, ZF equalisation may enhance the noise to unacceptable level $(N(z) \cdot C^{-1}(z))$

- Design of equaliser is a **trade off** between eliminating ISI and not enhancing noise too much
  - MMSE equalisation provides better trade off between eliminating ISI and enhancing noise

- Also the channel can be time-varying, hence *adaptive* equalisation is needed
  - Channel $\{c_i\}_{i=0}^{N_c}$ may change, and equaliser $\{w_i\}_{i=0}^{N_e}$ have to follow

Adaptive Equalisation — Architecture

- The generic framework of **adaptive** equalisation:



- Equaliser sets its coefficients w_i to 'match' channel characteristics
  - *Training mode*: Tx transmits a prefixed sequence known to Rx. The equaliser uses locally generated symbols $x[k]$ as the desired response to adapt w_i
    - ∗ As though, training data $\{x[k]\}$ were sent to receiver via a virtual pass
  - *Decision-directed mode*: the equaliser assumes the decisions $\hat{x}[k-k_d]$ are correct and uses them to substitute for $x[k-k_d]$ as the desired response

Adaptive Equalisation — Arrangement

- For fixed (time-invariant) channel, equalisation is done once during link set up
  - During link set up, a prefixed training sequence is sent, and equaliser is trained based on locally generated this training sequence

- For time-varying channel, equalisation must be performed periodically, transmission is organized in time frames, a small part of each frame contains training symbols
  - e.g. GSM mobile phone, middle of each Tx frame contains 26 training symbols

data	training	data

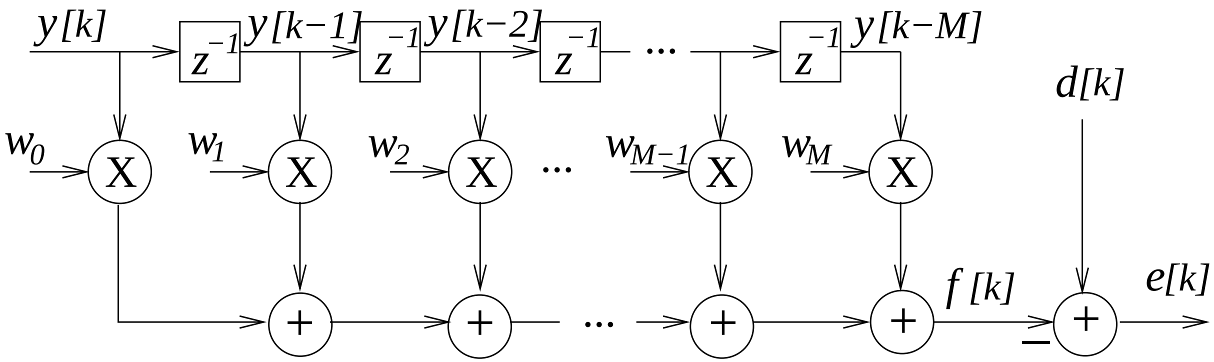
Frame structure

  - Receiver uses locally generated training symbols for training equaliser, and the trained equaliser then detects the data in the frame

- Blind equalisation: perform equalisation based on Rx signal $\{y[k]\}$ without access to training symbols $\{x[k]\}$, e.g. multipoint network, digital TV, etc
  - Note training causes extra bandwidth, thus blind equalisation is attractive but is more difficult

Linear Equaliser

- The setup of generic linear equaliser with length $N_e = M$ and filter coefficients w_i:



- The aim of the equaliser is to set its coefficients w_i to produce an output $f[k]$:

$$f[k] = \sum_{i=0}^{M} w_i^* \cdot y[k-i]$$

  – that is as close as possible to the desired signal $d[k]$:

$$d[k] = \begin{cases} x[k-k_d], & \text{training} \\ \hat{x}[k-k_d], & \text{decision directed} \end{cases}$$

  – Conventionally, conjugate w_i^* of w_i is used in producing equaliser output

  – Equaliser length M should be sufficiently long to cancel channel induced ISI, but not too long as to amplify noise too much

  – Equaliser decision delay k_d depends the zero locations of the channel transfer function $C(z)$: for minimum phase $C(z)$, $k_d = 0$; otherwise, $k_d > 0$

Mean Square Error

- The formulation of error signal $e[k]$:

$$e[k] = d[k] - f[k] = d[k] - \sum_{i=0}^{M} w_i^* \cdot y[k-i] = d[k] - \boldsymbol{w}^{\mathrm{H}} \cdot \boldsymbol{y}_k$$

with definitions $\boldsymbol{w}^{\mathrm{H}} = \left[w_0^* \; w_1^* \cdots w_M^* \right]$ and $\boldsymbol{y}_k = \left[y[k] \; y[k-1] \cdots y[k-M] \right]^{\mathrm{T}}$

- The **mean square error** formulation:

$$\begin{aligned} \mathcal{E}\left\{ |e[k]|^2 \right\} &= \mathcal{E}\left\{ \left| d[k] - \boldsymbol{w}^{\mathrm{H}} \cdot \boldsymbol{y}_k \right|^2 \right\} \\ &= \mathcal{E}\{d[k] \cdot d^*[k]\} - \boldsymbol{w}^{\mathrm{T}} \cdot \mathcal{E}\{d[k] \cdot \boldsymbol{y}_k^*\} - \boldsymbol{w}^{\mathrm{H}} \cdot \mathcal{E}\{\boldsymbol{y}_k \cdot d^*[k]\} + \boldsymbol{w}^{\mathrm{H}} \cdot \mathcal{E}\left\{ \boldsymbol{y}_k \cdot \boldsymbol{y}_k^{\mathrm{H}} \right\} \cdot \boldsymbol{w} \\ &= \sigma_d^2 - \boldsymbol{w}^{\mathrm{T}} \cdot \boldsymbol{p}^* - \boldsymbol{w}^{\mathrm{H}} \cdot \boldsymbol{p} + \boldsymbol{w}^{\mathrm{H}} \cdot \boldsymbol{R} \cdot \boldsymbol{w} \end{aligned}$$

  – desired signal power $\sigma_d^2 = \mathcal{E}\left\{ |x[k]|^2 \right\}$; cross-correlation vector $\boldsymbol{p} = \mathcal{E}\{\boldsymbol{y}_k \cdot d^*[k]\}$; autocorrelation matrix $\boldsymbol{R} = \mathcal{E}\{\boldsymbol{y}_k \cdot \boldsymbol{y}_k^{\mathrm{H}}\}$

- A standard optimisation procedure to achieve the minimum MSE yields Wiener-Hopf equation:

$$\frac{\partial}{\partial \boldsymbol{w}} \mathcal{E}\left\{ |e[k]|^2 \right\} = \mathbf{0} \;\Rightarrow\; -\boldsymbol{p} + \boldsymbol{R} \cdot \boldsymbol{w} = \mathbf{0}$$

Minimum Mean Square Error

- If $\boldsymbol{R}$ is invertible, then the optimum filter coefficients $\boldsymbol{w}_{\text{opt}}$ are given by Wiener-Hopf equation as:

$$\boldsymbol{w}_{\text{opt}} = \boldsymbol{R}^{-1} \cdot \boldsymbol{p}$$

  - The **MMSE solution** $\boldsymbol{w}_{\text{opt}}$ is unique and is also called the **Wiener solution**
  - The **minimum MSE** (MMSE) value is $\mathcal{E}\left\{|e[k]|^2\right\}|_{\boldsymbol{w}_{\text{opt}}} = \sigma_d^2 - \boldsymbol{p}^{\text{H}} \cdot \boldsymbol{R}^{-1} \cdot \boldsymbol{p}$

- Recall channel model in slide **64**, equaliser input vector $\boldsymbol{y}_k$ is expressed as

$$\boldsymbol{y}_k = \boldsymbol{C} \cdot \boldsymbol{x}_k + \boldsymbol{n}_k$$

  - $\boldsymbol{x}_k = \left[x[k]\ \ x[k-1] \cdots x[k-L]\right]^{\text{T}}$ with length $L = N_c + M$ and symbol power $\mathcal{E}\left\{|x[k]|^2\right\} = \sigma_d^2$
  - $\boldsymbol{n}_k = \left[n[k]\ n[k-1] \cdots n[k-M]\right]^{\text{T}}$ with noise power $\mathcal{E}\left\{|n[k]|^2\right\} = 2\sigma_n^2$
  - $(L+1) \times (L+1)$ CIR convolution matrix has Toeplitz form

$$\boldsymbol{C} = \left[\begin{array}{ccccccc} c_0 & c_1 & \cdots & c_{N_c} & 0 & \cdots & 0 \\ 0 & c_0 & c_1 & \cdots & c_{N_c} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & c_0 & c_1 & \cdots & c_{N_c} \end{array}\right] = \left[\boldsymbol{c}_0\ \boldsymbol{c}_1 \cdots \boldsymbol{c}_{k_d} \cdots \boldsymbol{c}_L\right]$$

- The MMSE equalisation solution is given by

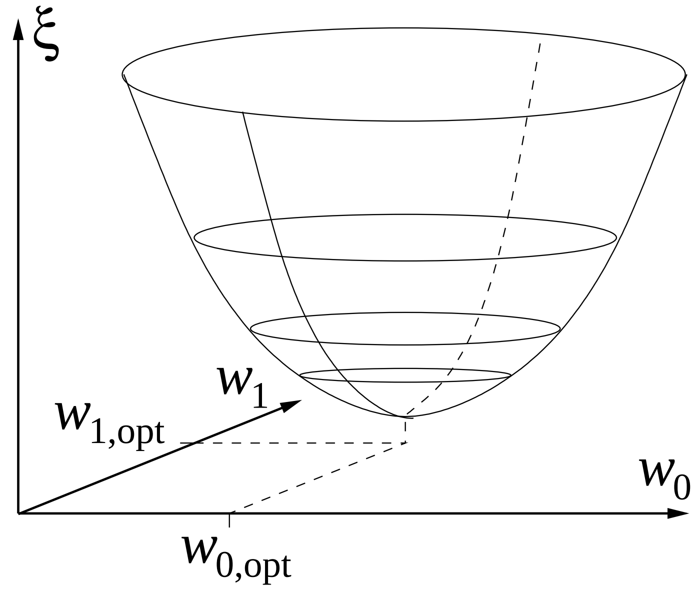
$$\boldsymbol{w}_{\text{opt}} = \left(\boldsymbol{C} \cdot \boldsymbol{C}^{\text{H}} + \frac{2\sigma_n^2}{\sigma_d^2} \cdot \boldsymbol{I}_{L+1}\right)^{-1} \cdot \boldsymbol{c}_{k_d}$$

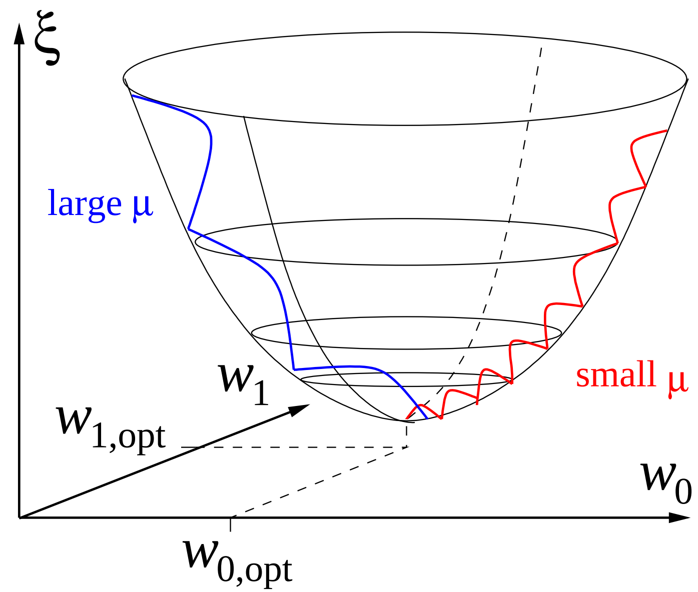
MSE Surface and Iterative Solution

- Example of mean square error surface for $2(M+1)=2$ real coefficients:
- Being **quadratic** in the filter coefficients $\boldsymbol{w}$, the MSE surface $\xi = \mathcal{E}\{|e[k]|^2\}$ is a hyperparabola in $2(M+1)+1$ dimensional real space
- **Unique** MMSE solution is $\boldsymbol{w}_{\text{opt}} = \boldsymbol{R}^{-1} \cdot \boldsymbol{p}$
- MMSE value is at the bottom of this hyperparabola
- If you do not want to calculate matrix inversion
- The solution can alternative be sought iteratively by moving $\boldsymbol{w}$ in the direction of the **negative gradient**:

$$\boldsymbol{w}_{l+1} = \boldsymbol{w}_l + \mu \cdot \left(- \boldsymbol{\nabla} \xi_l \right)$$

  - gradient vector at l-th iteration is $\boldsymbol{\nabla}\xi_l = -2\boldsymbol{p} + 2\boldsymbol{R}\boldsymbol{w}_l$
  - μ is the step size
  - An initial value of $\boldsymbol{w}_0$ is needed
- No more inversion of $\boldsymbol{R}$, but statistics $\boldsymbol{R}$ and $\boldsymbol{p}$ are assumed to be given





Least Mean Square Algorithm

- Rather than using the mean square error $\mathcal{E}\{|e(k)|^2\}$, using an instantaneous squared error $|e[k]|^2$ leads to an instantaneous (stochastic) gradient:

$$\hat{\nabla}\xi_k = \frac{\partial}{\partial \boldsymbol{w}} e[k] \cdot e^*[k] = -2e^*[k] \cdot \boldsymbol{y}_k$$

- LMS algorithm — initialisation: given initial weight vector

$$\boldsymbol{w}_0 = \left[w_0[0] \; w_1[0] \cdots w_M[0] \right]^{\mathrm{T}}$$

- LMS algorithm — during the k-th symbol (sample) period, it does

1. filter output:

$$f[k] = \boldsymbol{w}_k^{\mathrm{H}} \cdot \boldsymbol{y}_k = \sum_{i=0}^{M} w_i^*[k] \cdot y[k-i]$$

2. estimation error:

$$e[k] = d[k] - f[k]$$

3. weight adaptation:

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \mu \cdot e^*[k] \cdot \boldsymbol{y}_k$$

Recursive Least Squares Algorithm

- Forgetting factor λ, initial weight vector $\boldsymbol{w}_0$ and initial covariance matrix $\boldsymbol{P}[0] = \rho\boldsymbol{I}$, ρ being a large positive number

- At sample k
  - Equalize filter error

$$e[k] = d[k] - \boldsymbol{w}_{k-1}^{\mathrm{H}} \cdot \boldsymbol{y}_k$$

  - Kalman gain

$$\boldsymbol{k}(k) = \frac{\lambda^{-1}\boldsymbol{P}[k-1]\boldsymbol{y}_k}{1 + \lambda^{-1}\boldsymbol{y}_k^{\mathrm{H}}\boldsymbol{P}[k-1]\boldsymbol{y}_k}$$

  - Weight update

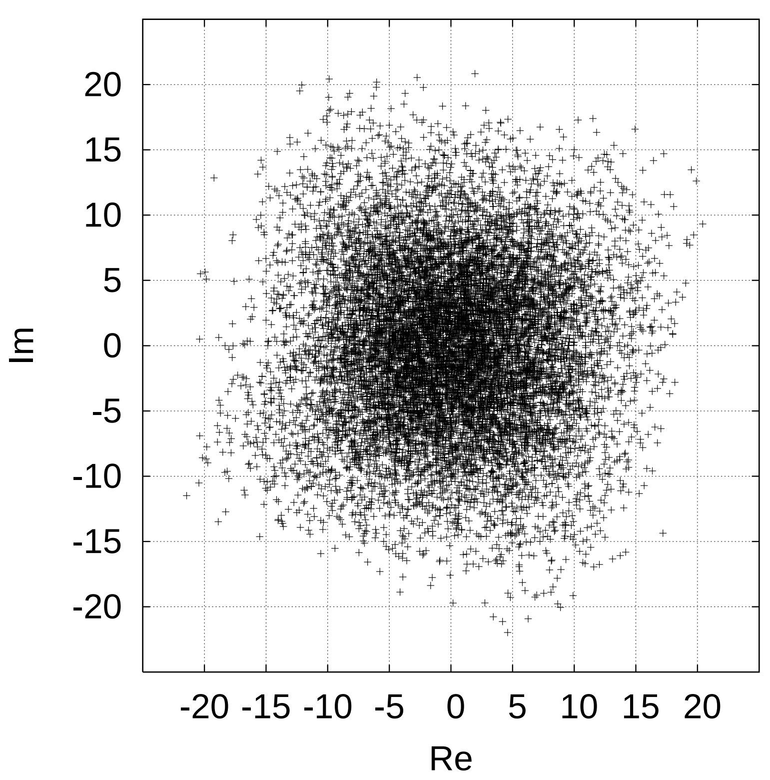
$$\boldsymbol{w}_k = \boldsymbol{w}_{k-1} + \boldsymbol{k}[k]e^*[k]$$

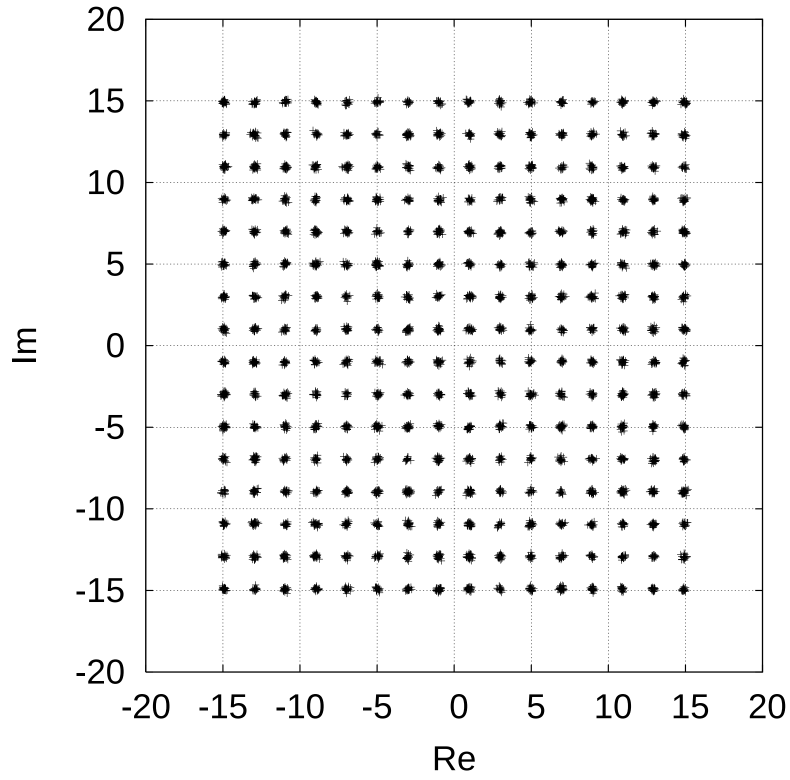
  - Covariance matrix

$$\boldsymbol{P}[k] = \lambda^{-1}P[k-1] - \lambda^{-1}\boldsymbol{k}[k]\boldsymbol{y}_k^{\mathrm{H}}P[k-1]$$

Blind Equalisation Example

- In blind equalisation, we do not have desired output $d[k] = x[k]$, equaliser must adjust its weights w_i based on channel observation $y[k]$ only and some other known information
- A blind equaliser called constant modulus algorithm aided soft decision directed scheme
  - 256-QAM
  - channel observations $y[k]$ and equaliser outputs $f[k]$ after convergence





Optimal Equalisation: Sequence Estimation

- Linear equalisation is based on **symbol-by-symbol** decision, i.e. at symbol instance k, it estimates symbol $x[k-k_d]$ transmitted at $k-k_d$

- Sequence estimation, i.e. estimate whole transmitted symbol sequence $\{x[k]\}$, is optimal (truly minimum symbol error rate)
  - In channel coding part, you'll learn convolutional coding, and optimal decoding can be done using Viterbi algorithm
  - Channel can be viewed as "convolutional codec", Viterbi algorithm used for "decoding", i.e. estimate transmitted symbol sequence $\{x[k]\}$

- For Example, GSM (QPSK modulation), training symbols used to estimate the channel $\{c_0, c_1, ..., c_6\}$ using for example LMS algorithm
  - Viterbi algorithm then used to "decode" Tx symbol sequence $\{x[k]\}$
  - Thus in GSM mobile phone hand set there are two Viterbi algorithms, one for channel coding, the other for equalisation

- Sequence estimation too complex for high-order modulation and long channel

Summary

- Discrete channel model in the presence of channel amplitude and phase distortion
- Equaliser tries to make the combined channel/equaliser a Nyquist system
- Design of equaliser is a trade off between eliminating ISI and not enhancing noise too much
- Adaptive equalisation structure: training mode and decision directed mode
- Linear equaliser (filter), the MMSE solution, and an iterative algorithm
- Least mean square algorithm
- Equalisation as sequence estimation