ELEC6021 — COMMS Assignment II

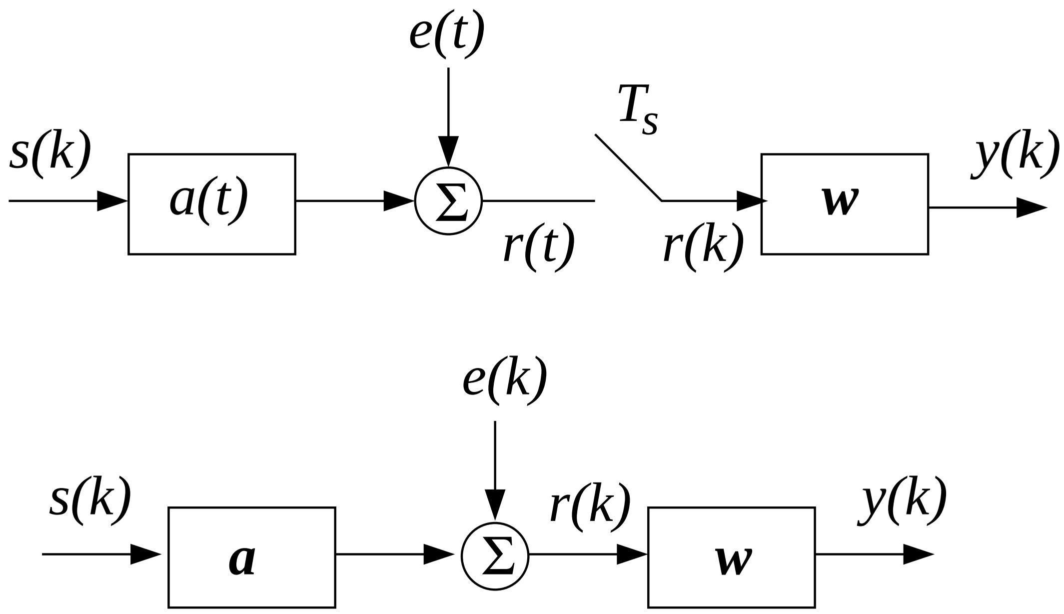
- Complete the set of simulations as indicated in Assignment II
- Hand in your assignment to the ECS Student Services (Course Office) on ——-
- Mark contribution to the unit: 25% (Total 25 marks)
- A few "hints" are given in this lecture
- Assignment and this lecture notes can be downloaded from

  http://www.ecs.soton.ac.uk/~sqc/EZ619/

- There will be no lecture for 2nd lecture time slot – use it for you to do this assignment

Baseband System with Symbol-Spaced Equaliser

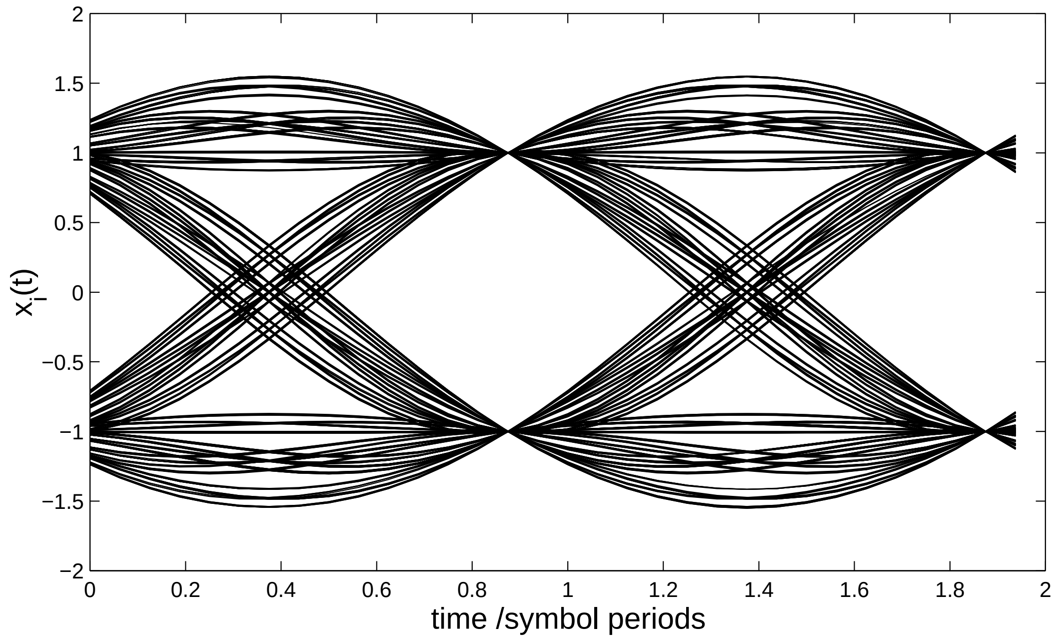
- Baseband model with symbol-spaced equaliser and equivalent symbol-spaced model:
  - T_s: symbol period
  - k: indicates symbol-spaced sampling quantity
  - Baseband "channel": $a(t) = g_R(t) \star c(t) \star g_T(t)$, convolution of Tx filter, channel (medium) and Rx filter
  - Symbol-spaced "channel": $\mathbf{a} = [a_0 \ a_1 \cdots a_{n_a}]^T$ is obtained by sampling $a(t)$ at symbol rate



  - Symbol spaced equaliser: $\mathbf{w} = [w_0 \ w_1 \cdots w_M]^T$

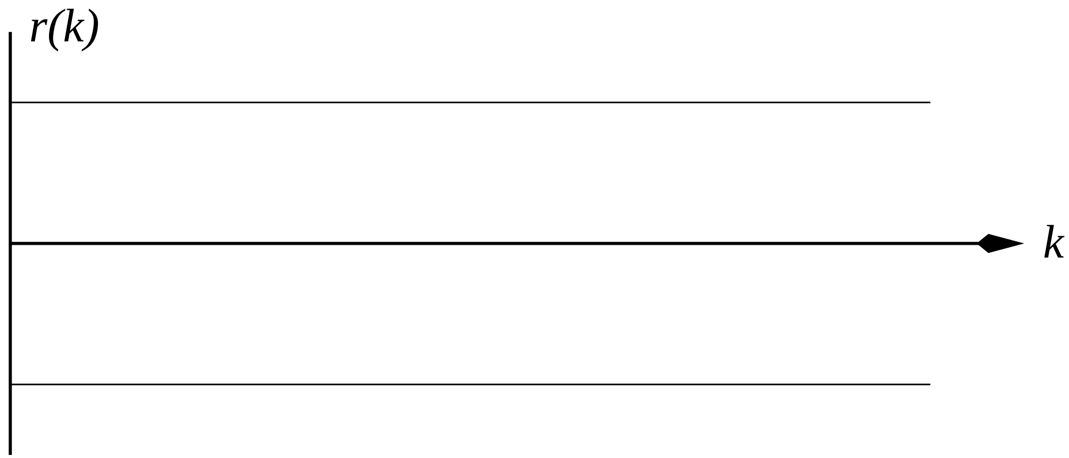
Eye Diagram and Correct Sampling

- BPSK modulation with ideal channel $c(t)$ and no noise, eye diagram of $r(t)$ looks like:



- Sampling $t = kT_s + \tau$: you have to choose correct sampling offset τ

- If sampling correctly and with a long sequence, $r(k)$ should look like:



- If sampling incorrectly, what $r(k)$ looks like?

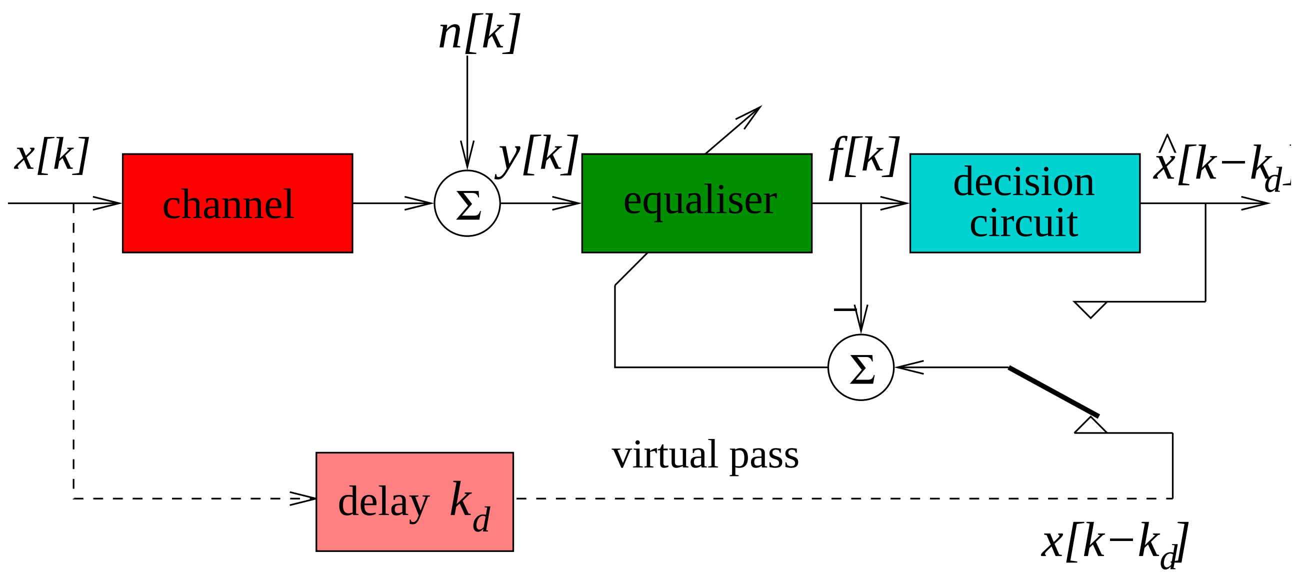
Symbol Spaced Channel Model

- Symbol spaced channel model $\mathbf{a}$ is obtained by sampling $a(t)$ at symbol rate
- If $c(t) = \delta(t) + 0.5\delta(t - T_s)$ and pulse shaping filter pair $g_T(t)$ and $g_R(t)$ are designed properly as square root of raised cosine pulse, then $\mathbf{a} = [1.0 \quad 0.5]^T$ by ignoring the delay length of pulse shaping

  This is only valid for the channel $c(t)$ having symbol-spacing paths
- Again, you'll have to choose sampling instances correctly, or the equaliser output $y(k)$ will not look as expected
- Can you see that the equaliser in the Assignment II 1 (c) is a zero-forcing one?
- For the MMSE solution and adaptive LMS algorithm, read for example your Digital Transmission Lecture Notes

Adaptive Equalisation

- The generic framework of adaptive equalisation:



- *Training mode*: Tx transmits a prefixed sequence known to Rx. The equaliser uses the locally generated symbols $x[k]$ as the desired response to adapt the equaliser
- *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
- *Blind equalisation*: adapt equaliser based only on Rx signal (Note that decision-direct adaptation is a blind equalisation)

Adaptive Algorithm

- **Adaptive equalisation with training**
  - The LMS algorithm

$$w_i(k) = w_i(k-1) + \mu(s(k-d) - y(k))r(k-i), \quad 0 \leq i \leq M$$

  - μ is a small positive adaptive gain, d is equaliser decision delay (In assignment II we choose $d = 0$), and initialisation $\mathbf{w} = [0\ 0 \cdots 0]^T$

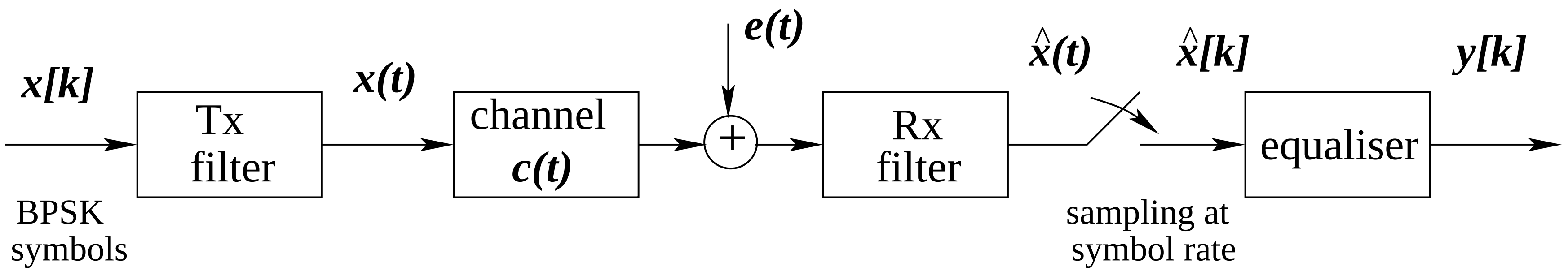
- **Blind adaptive equalisation**
  - The decision directed adaptation: Let $\hat{s}(k-d) = \mathcal{Q}(y(k))$,

$$w_i(k) = w_i(k-1) + \mu(\hat{s}(k-d) - y(k))r(k-i), \quad 0 \leq i \leq M$$

  - μ is a small positive adaptive gain, and initialisation $\mathbf{w} = [0 \cdots 0\ 1\ 0 \cdots 0]^T$

- Adaptive step size μ should be small enough to ensure convergence, but if it is too small convergence will be too slow

A Few Hints



- Make sure that your system is so implemented such that the overall system gain from $x[k]$ to $\hat{x}[k]$ is unity for ideal channel $c(t)$
- Make sure the gain from the noise $e(t)$ to Rx signal is also unity, as we want a signal to noise ratio (SNR) of $10 \cdot \log_{10} \frac{1}{0.04} \approx 14$ dB, with a noise variance of 0.04
- Choose sampling instances carefully (consider half length of the combined Rx/Tx filters impulse response), and use eye diagram to tune sampling instances if needed
- Think how are you going to plot Bit Error Rate versus Signal to Noise Ratio

Symbol-Spaced Signal Model

- Received signal model

$$r(k) = \sum_{i=0}^{n_a} a_i s(k-i) + e(k)$$

- Equaliser model

$$y(k) = \sum_{i=0}^{M} w_i r(k-i)$$

- In vector/matrix form

$$y(k) = \mathbf{w}^T \mathbf{r}(k), \qquad r(k) = \mathbf{A}\,\mathbf{s}(k) + \mathbf{e}(k)$$

where $\mathbf{w} = [w_0 \; w_1 \cdots w_M]^T$, $\mathbf{r}(k) = [r(k) \; r(k-1) \cdots r(k-M)]^T$, $\mathbf{e}(k) = [e(k) \; e(k-1) \cdots e(k-M)]^T$, $\mathbf{s}(k) = [s(k) \; s(k-1) \cdots s(k-M-n_a)]^T$, and

$$\mathbf{A} = \begin{bmatrix} a_0 & a_1 & \cdots & a_{na} & 0 & \cdots & 0 \\ 0 & a_0 & a_1 & \cdots & a_{na} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \cdots & \ddots & 0 \\ 0 & \cdots & 0 & a_0 & a_1 & \cdots & a_{na} \end{bmatrix}$$

Minimum Mean Square Solution

- Mean square error

$$J(\mathbf{w}) = E[|s(k-d) - y(k)|^2]$$

- Minimum mean square error solution

$$\mathbf{w}_{\text{MMSE}} = \arg\min_{\mathbf{w}} J(\mathbf{w}) = \left(\mathbf{A}\mathbf{A}^T + \frac{\sigma_e^2}{\sigma_s^2}\mathbf{I}\right)^{-1} \mathbf{A}|_d$$

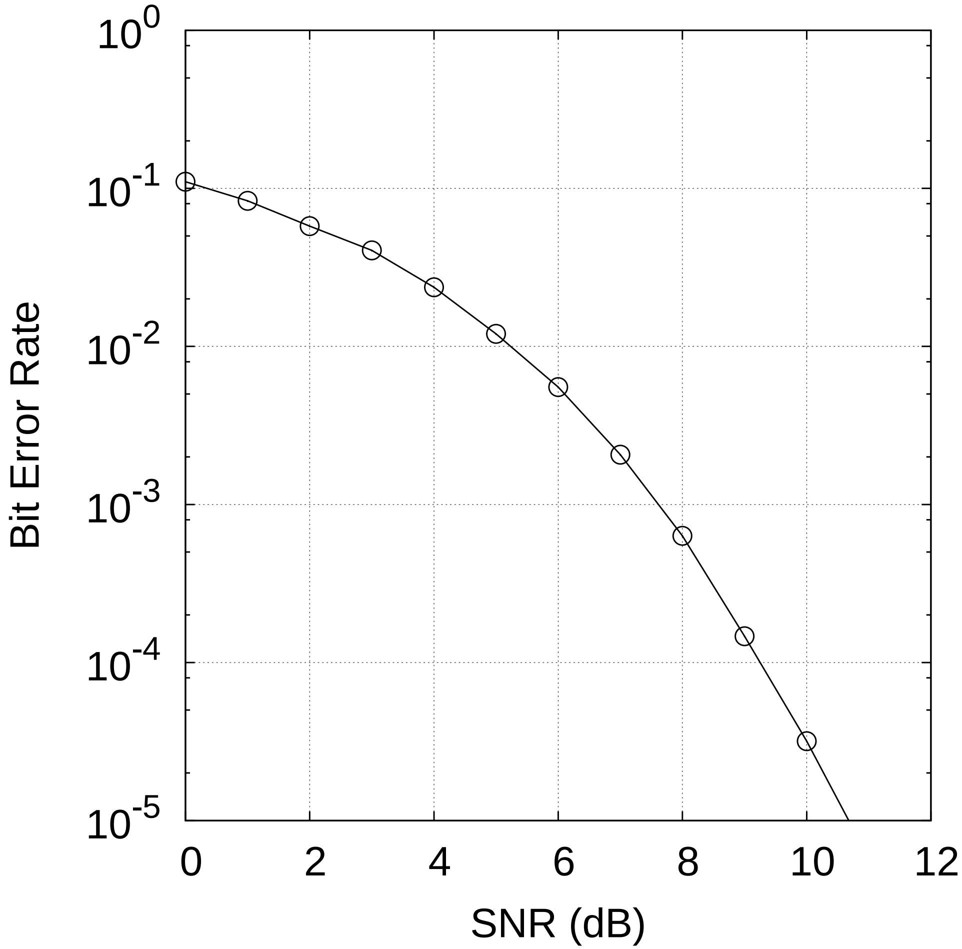
  where $\mathbf{I}$ is the identity matrix, $\sigma_s^2 = 1$ is symbol power of $s(k)$, $\mathbf{A}|_d$ denotes d-th column of $\mathbf{A}$, and d is equaliser's decision delay

- Signal to noise ratio:

$$\text{SNR} = \frac{\sigma_s^2}{\sigma_e^2} \sum_{i=0}^{n_a} a_i^2$$

Bit Error Rate Plot

- BER plot



- You can calculate the theoretical BER, or you can use simulation to compute BER – think how to get adequate simulation result

Investigation of Least Mean Square Algorithm

$E[\mathbf{a}(k)]$ and $J(\mathbf{a}(k))$ are approximated using sample averages over 500 runs