Chapters 9.3-9

Riccardo Micccini[1] Eren Can [1]

[1]Technical University of Denmark
Digital Communication

November 22, 2016

Modulation Schemes not Requiring Coherent References

- .

Differential Phase-Shift Keying (DPSK)

- .

Comparison of Digital Modulation Systems

- .

Multipath Interference (1)

- Additive Gaussian noise is not sufficient to accurately model the transmission channel
- Other sources of degradation:
  - bandwidth limiting by the channel
  - impulse noise (lightnings, switching)
  - RF interference from other transmitters
  - *multipath interference* from signal reflections and scattering

Multipath Interference (2)

- Two-way multipath model: $y(t) = s_d(t) + \beta s_d(t - \tau_m) + n(t)$

  $n(t)$ Gaussian noise component
  $s_d(t)$ Signal from the direct path
  β Gain of secondary path component
  τ_m Time delay of secondary path component

- For binary phase-shift keying signals: $s_d(t) - Ad(t)\cos(\omega_c t)$

  $d(t)$ Data stream (sequence of ± 1 rectangular pulses) of width T
  ω_c Carrier frequency

Multipath Interference (3)

- Input of the integrator at the receiving end: $x(t) = LP\{2y(t)\cos(\omega_c t)\} = Ad(t) + \beta Ad(t-\tau_m)\cos(\omega_c\tau_m) + n_c(t)$
- Two scenarios:

  $\tau_m/T \cong 0$ The original and reflected signals are almost congruent, so $\omega_c\tau_m$ is uniformly distributed in $[-\pi,\pi]$. When many reflection components are considered, the envelope of the signal assumes a Rayleigh or Ricean distribution

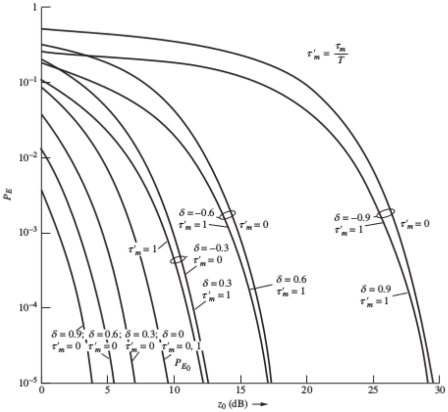
  $0 < \tau_m/T \leq 1$ Adjacent bits in the original and reflected signals overlap; inter-symbol interference appears

Multipath Interference - second scenario (1)

- Four equally likely cases; total probability of error is:
  $P_E = \frac{1}{4}[P(E|++) + P(E|-+) + P(E|+-) + P(E|--)]$
- Noise on the integrator integrator out is Gaussian-distributed with $\mu = 0$ and $\omega_n^2 = N_0 T$
- Due to the symmetric nature of the overlapping bits and Gaussian probability density function, only two cases need to be computed
  - $P(E|++) = P(E|--) = Q\left[\sqrt{\frac{2E_b}{N_0}}(1+\delta)\right]$
  - $P(E|+-) = P(E|-+) = Q\left[\sqrt{\frac{2E_b}{N_0}}\left((1+\delta) - \frac{2\delta\tau_m}{T}\right)\right]$
- After substituting the cases above into the matching ones:
  $P_E = \frac{1}{2}Q\left[\sqrt{2z_0}(1+\delta)\right] + \frac{1}{2}Q\left[\sqrt{2z_0}\left((1+\delta) - \frac{2\delta\tau_m}{T}\right)\right]$

Multipath Interference - second scenario (2)

- Overal probability of error changes with $z_0 = \frac{E_b}{N_0} = \frac{A^2 T}{2N_0}$



Equalization

- Equalization is used in telecommunication to reverse the signal degradation caused by multipath propagation and bandwidth limitations
- Simplest form of equalization consists in an inverse filter - *Tapped-delay-line filter*
- Two ways of determining the filter coefficients:
  - zero-forcing
  - mean-square error

Equalization by Zero Forcing

- Impulse response of equalized output:
  $p_{eq}(mT) = \sum_{n=-N}^{N} \alpha_n p_c((m-n)T) \Rightarrow [P_{eq}] = [P_c][A]$
- $[P_{eq}]$ is a column vector composed of: N zeros, 1, N zeros
- Equalizarion filter coefficient matrix: $[A]_{opt} = [P_c]^{-1}[P_{eq}]$
- Multiplying by $[P_{eq}]$ corresponds to picking the middle column of matrix $[P_c]^{-1}$

Equalization by Minimum Mean-Squared Error (1)

- Obtain filter coefficients that minimize the difference between the output of the equalizer and the actual output:
  $\varepsilon = E\left\{[z(t) - d(t)]^2\right\} = minimum$
  - $z(t)$ is the equalizer output response (incl. noise):
    $z(t) = \sum_{n=-N}^{N} \alpha_n p_c((m-n)T)$
  - $d(t)$ is the desired response

Equalization by Minimum Mean-Squared Error (2)

- ε is concave and can be minimized by derivation: $\frac{\delta\varepsilon}{\delta\alpha_m} = 0 = 2E\left\{[z(t) - d(t)]\frac{\delta z(t)}{\delta\alpha_m}\right\}$
- Substituting $z(t)$ gives the following conditions (in terms of cross-correlation): $R_{yz}(m\Delta) = R_{yd}(m\Delta) = 0$
  - $R_{yz}(\tau) = E[y(t)z(t+\tau)]$
  - $R_{yd}(\tau) = E[y(t)d(t+\tau)]$
- In terms of matrices: $[R_{yy}][A]_{opt} = [R_{yd}]$
- Solving for the filter taps: $[A]_{opt} = [R_{yy}]^{-1}[R_{yd}]$

Tap Weight Ajustment (LMS Algorithm) (1)

- How to obtain $d(t)$
  1. Periodically send known data sequence used for weight adjustment
  2. Use method 1 for first guess and then use detected data (*decision-directed* mode)
- Apply gradient descent to initial weight values ($[A]^{(0)}$):
  $[A]^{(k+1)} = [A]^{(k)} + \frac{1}{2}\mu[-\nabla\varepsilon^{(k)}]$

  k iteration of weights calculation

  $\nabla\varepsilon$ slope of error surface

  μ size of the step

Tap Weight Ajustment (LMS Algorithm) (2)

- Alternative approach (Least-Mean-Square):
  $\alpha_m^{(k+1)} = \alpha_m^{(k)} - \mu y[(k-m)\Delta]\epsilon(k\Delta)$
- $\epsilon(k\Delta)$ is the error given by $y_{eq}(k\Delta) - d(k\Delta)$

  $y_{eq}(k\Delta)$ equalization filter output
  $d(k\Delta)$ data sequence used for training