# 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

- In this section, now we consider two modulation schemes that you do not need to require the acquisition of a local reference signal in phase coherence with the received carrier.

# Differential Phase-Shift Keying (DPSK)

- The implementation of a such a scheme presupposes two things;
  1. The unknown phase perturbation on the signal varies slowly that the phase is constant from one signalling interval to next.
  2. The phase during a given signalling interval bears a known relationship to the phase during the preceding signalling interval bears a known relationship to the phase during the preceding signalling interval.

**Table 9.3 Differential Encoding Example**

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Message sequence: | | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| Encoded sequence: | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| Reference digit: | ↑ | | | | | | | | | |
| Transmitted phase: | 0 | 0 | $\pi$ | 0 | 0 | 0 | 0 | $\pi$ | 0 | $\pi$ |

# Differential Encoding Message Sequence

- An arbitrary reference binary digit is being selected as an initial digit of the sequence
- For each digit , the present digit used as a reference
- 0 in the message sequence is encoded as a transition from state of reference digit to the opposite state in the encoded message sequence
- 1 encoded as no change of state

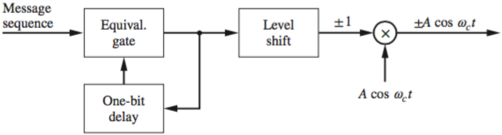
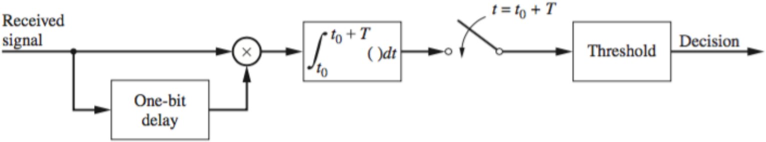



**Figure 9.16**
Block diagram of a DPSK modulator.

# Figure for Differential Encoding Message Sequence

**Table 9.4 Truth Table for the Equivalence Operation**

| Input 1 (Message) | Input 2 (Reference) | Output |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |



# Differential Encoding Message Sequence

- After the reference bit and plus the first encoded bit, signal input become $S_1 = A\cos(\omega_c)t$ and $R_1 = A * cos(w_c) * t$
- Than the output correlator is; $v_1 = \int_0^T A^2 \cos^2(\omega_c t)dt$ which eventually become $\frac{1}{2}A^2T$
- The optimum detector for binary will become $l = x_k x_k - 1 + y_k y_k - 1$
- Without a loss of of generality, we can choose $\theta = 0$; we found outputs at $t = 0$ to be:
  - $x_0 = \frac{AT}{2} + n_1$ and $y_0 = n_3$ and where $n_1 = \int_{-T}^{0} n(t)\cos^2(\omega_c t)dt$
  - $n_3 = \int_{-T}^{0} n(t)\sin^2(\omega_c t)dt$. Similarly, at the time $t = T$, the outputs are ; $x_1 = \frac{AT}{2} + n_2$ and $y_1 = n_4$
  - $n_2 = \int_0^T n(t)\cos^2(\omega_c t)dt$
  - $n_4 = \int_0^T n(t)\sin^2(\omega_c t)dt$

# Important Figure for Differential Encoding

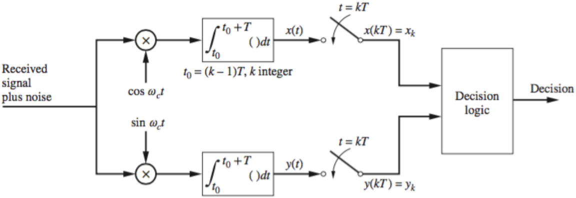



**Figure 9.18**
Optimum receiver for binary differential phase-shift keying.

If $\ell > 0$, the receiver chooses the signal sequence

$$s_1(t) = \begin{cases} A\cos(\omega_c t + \theta), & -T \le t < 0 \\ A\cos(\omega_c t + \theta), & 0 \le t < T \end{cases} \tag{9.95}$$

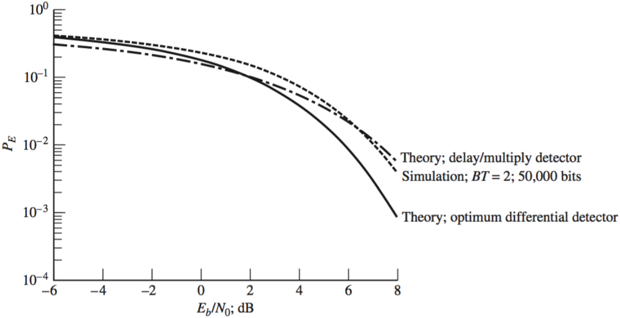
as having been sent. If $\ell < 0$, the receiver chooses the signal sequence

$$s_2(t) = \begin{cases} A\cos(\omega_c t + \theta), & -T \le t < 0 \\ -A\cos(\omega_c t + \theta), & 0 \le t < T \end{cases} \tag{9.96}$$

- It follows as $n_1, n_2, n_3 and n_4$ are uncorrelated and zero-mean Gaussian random variables with variances $\frac{N_0 T}{4}$ and they are independent.
- Expression for Probability error
  $P_E = Pr[(\frac{AT}{2} + n_1)(\frac{AT}{2} + n_2) + n_3 n_4 < 0]$
- We can define new gaussian random variables such as:
  $\omega_1 = \frac{n_1}{2} + \frac{n_2}{2}$
  $\omega_2 = \frac{n_1}{2} - \frac{n_2}{2}$
  $\omega_3 = \frac{n_3}{2} + \frac{n_4}{2}$
  $\omega_4 = \frac{n_3}{2} - \frac{n_4}{2}$

- Probability can be written in terms of Gaussian variables: $P_E = Pr[(\frac{AT}{2} + \omega_1)^2 + (\omega_3)^2 < (\omega_2^2 + \omega_4^2)]$
- Gaussian variables will also let us define the Ricean random variables. Ricean random variable will become: $R_1 = \sqrt{(\frac{AT}{2} + \omega_1)^2 + \omega_3^2}$
- Also Rayleigh random variable will become $R_2 = \sqrt{\frac{\omega_2^2}{\omega_4^2}}$
- If we also define the bit energy $E_b$ as $A^2 \frac{A^2 T}{2}$ will give; $P_E = \frac{1}{2} e^{(\frac{-E_b}{N_0}}$ for the optimum DPSK receiver.
- At the large values $\frac{-E_b}{N_0}$ values of ; $P_E = Q[\sqrt{\frac{-E_b}{N_0}}] = Q[\sqrt{z}]$

- Following result obtained by using the asymptotic approximation; $P_E = \frac{e^{(-E_b/N_0)}}{2\sqrt{\pi\frac{E_b}{N_0}}}$



# Comparison of Digital Modulation Systems

- Bit error probabilities are compared in Figure 9.22 for the modulation schemes that considered in this chapter. Note that the curve for antipodal binary PAM is identical to BPSK
- Also bit error probability of antipodal PAM becomes worse the larger M. Curves move more to the right as M gets larger

# Important figure for Chapter 9

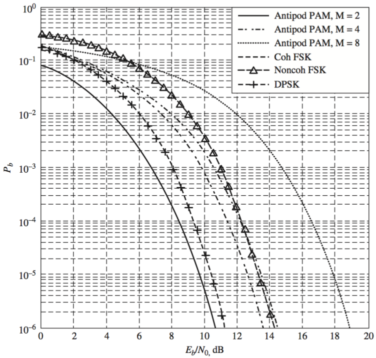



**Figure 9.22**
Error probabilities for several binary digital signaling schemes.

- Non-coherent binary FSK and PAM with M=4 have almost identical performance at large signal-to-noise ratios.
- In addition to cost and complexity implementation, there are many other considerations in choosing one type of digital data system over another.
- Some channels, where the channel gain, phase or when both are in effect,we use a noncoherent system may be dictated because of impossibility of establishing a coherent reference at the receiver under such conditions. They will be referred as "fading".

# 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 |
  | $\beta$ | Gain of secondary path component |
  | $\tau_m$ | 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 $\pm 1$ rectangular pulses) of width T |
  | $\omega_c$ | 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

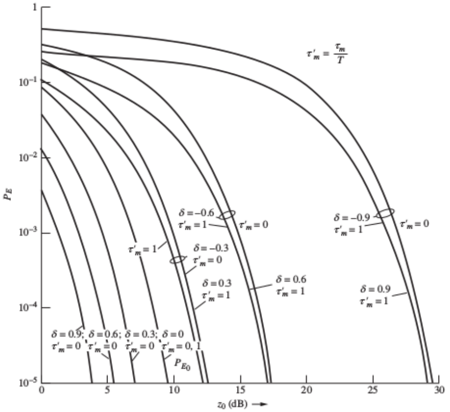
  $0 < \tau_m/T \leq 1$ Adjacent bits in the original and reflected signals overlap; inter-symbol interference appears

# 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)

- $\varepsilon$ 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

  $\mu$ 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