Modulation and Coding
A General Digital Communication System

Source → A/D Conversion → Source Encoder → Channel Encoder → Modulator

Channel

Synchronization

User → D/A Conversion → Source Decoder → Channel Decoder → Demodulator

Costas N. Georghiades
Modulation

Binary (baseband) Modulator:

1001... → Modulator → 1 0 0 1

1 ↔ 0 T

0 ↔ 0 T

Costas N. Georghiades
M-ary Modulation

- M-ary modulators have an alphabet of $M$ signals
- Example: $M=4$

\[
\begin{array}{c}
00 \leftrightarrow s_1(t) \\
01 \leftrightarrow s_2(t) \\
10 \leftrightarrow s_3(t) \\
11 \leftrightarrow s_4(t)
\end{array}
\]

In general, $M$ signals can convey $\log_2(M)$ bits of information each.
An optimum receiver minimizes the average probability of error:

$$\min \Pr(m \neq \hat{m})$$
Optimum Detection

If signals are equally likely, the optimum receiver (detector) is a **Maximum Likelihood (ML)** receiver. It chooses the most likely (probable) signal to have been transmitted given the received data $r(t)$.

For an AWGN channel, the ML receiver becomes:

$$
\max_{i=1,2,\cdots,M} \ell_i = \int_0^T r(t) \cdot S_i(t) \, dt - \frac{1}{2} \int_0^T S_i^2(t) \, dt
$$

Correlation

Signal Energy

$S_k(t) \Leftrightarrow m_k$
Block Diagram of Optimum Receiver
The **Matched-filter Receiver**

If the modulation signals have equal energy:

\[
\max_{i=1,2,\cdots,M} \ell_i = \int_0^T r(t) \cdot S_i(t) dt
\]

Another implementation of the optimum receiver:

\[
\int_0^T r(t) \cdot S_i(t) dt = \int_0^T r(a) \cdot h(T-a) da \iff h(t) = S_i(T-t)
\]
Example: Matched Filter Receiver

Binary, \textit{antipodal} signaling

\[ S_1(t) = \sin\left(\pi \frac{t}{T}\right) \]
\[ 0 \leq t \leq T \]

\[ S_2(t) = -S_1(t) \]

Optimum Receiver:

\[ r(t) \rightarrow h(t) \rightarrow t = kT \rightarrow \ell_i \]

Matched-filter output: noiseless case

\[ h(t) = S_1(T - t) = S_1(t) \]

Matched-filter output: with noise
Performance of the Matched-filter Receiver

Binary Modulation:

\[
P(e) = \frac{1}{2} \text{erfc}\left(\frac{d}{2\sqrt{N_0}}\right)
\]

where

\[
\text{erfc}(x) = 1 - \frac{2}{\sqrt{\pi}} \int_{0}^{x} e^{-y^2} dy
\]

\[
d^2 = \int_{0}^{T} [S_1(t) - S_2(t)]^2 dt
\]

Euclidean Distance

\[
P(e)
\]
The Chernoff Bound

\[ \Pr(X \leq a) \leq e^{\mu(s) - a \cdot s}, \quad s \leq 0 \]

\[ \mu(s) = \ln E[e^{sX}] \]

To obtain the tightest bound:

\[ \left. \frac{d\mu(s)}{ds} \right|_{s=s_{\text{min}}} = a \]

For the binary error probability in an AWGN channel:

\[ P(e) = \frac{1}{2} \text{erfc} \left( \frac{d}{2\sqrt{N_0}} \right) \leq e^{-\frac{d^2}{4N_0}} \]
The Chernoff Bound: Antipodal and Orthogonal Signals

\[ P(e) = \frac{1}{2} \text{erfc} \left( \sqrt{\frac{E}{N_0}} \right) \leq e^{-\frac{E}{N_0}}, \quad \text{Antipodal Signals} \]

\[ P(e) = \frac{1}{2} \text{erfc} \left( \sqrt{\frac{E}{2N_0}} \right) \leq e^{-\frac{E}{2N_0}}, \quad \text{Orthogonal Signals} \]
**Exact BER and the Chernoff bound**

\[
\frac{E}{N_0} \quad \frac{1}{2} \text{erfc} \left( \sqrt{\frac{E}{N_0}} \right)
\]

Error-Probability

Antipodal Signals

SNR, dB

Costas N. Georghiades
Performance of $M$-ary Signaling

To determine the performance for $M$-ary signaling, we often use the **Union-Chernoff bound**

$$P(A \cup B) \leq P(A) + P(B)$$

$$P(e) \leq \frac{2}{M} \sum_{k=1}^{M} \sum_{i=1}^{k-1} e^{\frac{d_{ik}^2}{4N_0}} \leq (M - 1) \cdot e^{\frac{d_{\text{min}}^2}{4N_0}}$$

where

$$d_{\text{min}}^2 = \min_{i,k} d_{ik}^2$$

At large SNR’s, the performance of a signal set in an AWGN channel is determined by the minimum Euclidean distance between signals.
**Signal Space**

**Definition:** A set of signals \( \{\varphi_1(t), \varphi_2(t), \ldots, \varphi_N(t)\} \), for \( 0 \leq t \leq T \), are said to be **orthonormal** if:

\[
\int_0^T \varphi_i(t)\varphi_j(t)dt = \begin{cases} 
1, & i = j \\
0, & i \neq j
\end{cases} \quad i, j = 1, 2, \ldots, N
\]

If \( s(t) \) is any signal in the \( N \)-dimensional space spanned by the above signals, then it can be expressed as

\[
s(t) = \sum_{i=1}^{N} s_i \varphi_i(t)
\]

for some set or real numbers \( s_1, s_2, \ldots, s_N \). The \( N \) coefficients uniquely describing \( s(t) \) can be obtained using

\[
s_k = \int_0^T s(t)\varphi_k(t)dt, \quad k = 1, 2, \ldots, N
\]
Energy, Distance, Correlation in Signal Space

A signal can be represented as a vector in signal space

\[ S(t) \leftrightarrow \mathbf{S} \]

\[ \int_0^T s^2(t)dt \leftrightarrow \| \mathbf{S} \|^2 = \sum_{i=1}^{N} S_i^2 \]

\[ \int_0^T r(t) \cdot S(t)dt \leftrightarrow \mathbf{r}^T \cdot \mathbf{S} = \sum_{i=1}^{N} r_i \cdot S_i \]

\[ \int_0^T \left[ S_1(t) - S_2(t) \right]^2dt \leftrightarrow \| \mathbf{S}_1 - \mathbf{S}_2 \|^2 = \sum_{k=1}^{N} (S_{1i} - S_{2i})^2 \]
Example

\[ S_1(t) = \varphi_1(t) + \varphi_2(t) \]
\[ S_2(t) = \varphi_1(t) - \varphi_2(t) \]
\[ S_3(t) = -2\varphi_1(t) \]

\[ E_1 = \left(1^2 + 1^2 \right) = 2 \]
\[ E_2 = \left(1^2 + (-1)^2 \right) = 2 \]
\[ E_3 = \left(2^2 + 0^2 \right) = 4 \]
**Binary Orthogonal and Antipodal Signaling**

**Orthogonal**

\[ d = \sqrt{2E} \]

\[ P(e) = \frac{1}{2} \text{erfc} \left( \sqrt{\frac{E}{2N_0}} \right) \leq e^{-\frac{E}{2N_0}} \]

**Antipodal**

\[ d = 2\sqrt{E} \]

\[ P(e) = \frac{1}{2} \text{erfc} \left( \sqrt{\frac{E}{N_0}} \right) \leq e^{-\frac{E}{N_0}} \]

\[ \text{SNR} = \frac{E}{N_0} \]

Antipodal signaling is 3dB better than orthogonal
Orthogonal Vs. Antipodal Signaling

![Graph showing error probability vs. SNR for orthogonal and antipodal signaling]

Orthogonal

Antipodal

3dB
Pulse-Amplitude Modulation (PAM)

\[ S_1(t) = -3A\phi(t) \quad 00 \iff S_1(t) \]
\[ S_2(t) = -A\phi(t) \quad 01 \iff S_2(t) \]
\[ S_3(t) = -S_2(t) \quad 10 \iff S_3(t) \]
\[ S_4(t) = -S_1(t) \quad 11 \iff S_4(t) \]

\[ \phi(t) = \sqrt{\frac{2}{T}} \cos(2\pi f_c t) \quad 0 \leq t \leq T \]

\[ \bar{E} = \frac{A^2}{M} \sum_{i=1}^{M} (2i - 1 - M)^2 = A^2 \frac{(M^2 - 1)}{3} \]
\[ E_p = A^2 (M - 1)^2 \]

\[ \frac{E_p}{E} = 3 - \frac{6}{M + 1} \rightarrow 3 \]

Disadvantage: Large peak-to-average energy
PAM Performance in AWGN (coherent)

\[ \ell_i = a_i \cdot \int_0^T r(t) \cdot \phi(t) \, dt - \frac{a_i^2}{2} \]

\[ a_i = A \cdot (2i - 1 - M) \]

Optimum Detector

\[ P(e) = \frac{(M - 1)}{M} \cdot \text{erfc}\left( \sqrt{\frac{3 \log_2(M)}{M^2 - 1} \cdot \frac{\bar{E}_b}{N_0}} \right) \]

\[ \bar{E}_b \equiv \text{Average energy per bit} \]
PAM Performance in AWGN

![Graph showing PAM performance in AWGN](image-url)

- Error-Probability
- SNR, dB
- M=2
- M=4
- M=8
- M=16
Phase-Shift Keying (PSK)

Signal space representation and decision regions

\[ S_i(t) = \sqrt{\frac{2E}{T}} \cdot \cos\left(2\pi f_c t - (i - 1)\frac{2\pi}{M}\right), \]

\[ i = 1, 2, \ldots, M, \quad 0 \leq t \leq T \]

\[ \bar{E} = E \quad E_p = E \]

\[ \frac{E_p}{E} = 1 \]

Requires good carrier (frequency and phase) reference. Used for some telephone and satellite modems.
Optimum (coherent) Receiver for PSK

\[ \ell_i = a_i \cdot \int_0^T r(t) \cdot \cos(2\pi f_c t) \, dt + b_i \cdot \int_0^T r(t) \cdot \sin(2\pi f_c t) \, dt \]

\[ = a_i \cdot r_c + b_i \cdot r_s \]

\[ a_i = \cos \left[ \frac{2\pi}{M} (i - 1) \right] \quad b_i = \sin \left[ \frac{2\pi}{M} (i - 1) \right] \]

No gain control is needed, but it requires an absolute carrier phase reference.
Coherent M-PSK Receiver Block Diagram

\[ r(t) \times \cos(2\pi f_c t) \rightarrow \int () dt \]

\[ r(t) \times \sin(2\pi f_c t) \rightarrow \int () dt \]

Metric Computation
\[ a_i r_c + b_i r_s \]

\[ t = T \]

Costas N. Georghiades
PSK Performance in AWGN

\[ P(e) = \frac{1}{2} \text{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right) \quad \text{Binary PSK (BPSK)} \]

\[ P(e) = \text{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right) \cdot \left[ 1 - \frac{1}{4} \text{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right) \right] \quad \text{Quaternary PSK (QPSK)} \]

\[ P(e) = 1 - \int_{\frac{\pi}{M}}^{\frac{\pi}{M}} f(\theta) \cdot d\theta \quad M \text{ - ary PSK} \]

\[ f(\theta) = \frac{1}{2\pi} e^{-\text{SNR}} \left[ 1 + \sqrt{\pi \cdot \text{SNR} \cdot \cos(\theta)} \cdot e^{\text{SNR}\cos^2(\theta)} \left( 1 + \text{erf} \left( \sqrt{\text{SNR} \cdot \cos(\theta)} \right) \right) \right] \]

\[ P(e) \approx \text{erfc} \left( \sqrt{\log_2(M) \frac{E_b}{N_0}} \cdot \sin \left( \frac{\pi}{M} \right) \right), \quad M \text{ - ary PSK} \]
$f(\theta)$
PSK Performance

![Graph showing PSK performance with error probability versus SNR for different modulation levels (M=2, M=4, M=8, M=16).]
Differential PSK (DPSK)

Phase is differentially encoded and detected:

**Binary DPSK:**
- 0 $\iff$ Shift by $0^\circ$ relative to previous phase
- 1 $\iff$ Shift by $180^\circ$ relative to previous phase

**Quaternary DPSK:**
- 00 $\iff$ Shift by $0^\circ$ relative to previous phase
- 01 $\iff$ Shift by $90^\circ$ relative to previous phase
- 10 $\iff$ Shift by $180^\circ$ relative to previous phase
- 11 $\iff$ Shift by $270^\circ$ relative to previous phase
Detection of DPSK Signals

\[ r_k = \int_{-\infty}^{\infty} r(t) \cdot e^{-j2\pi f_c t + \phi} p(t - kT) dt \]

\[ r_{k-1} = \sqrt{E} \cdot e^{j(\theta_{k-1} - \phi)} + n_{k-1} \]

\[ r_k = \sqrt{E} \cdot e^{j(\theta_k - \phi)} + n_k \]

\[ r_{k-1}^* \cdot r_k = E \cdot e^{j(\theta_k - \theta_{k-1})} + \sqrt{E} \cdot e^{j(\theta_k - \phi)} \cdot n_{k-1} + \sqrt{E} \cdot e^{-j(\theta_{k-1} - \phi)} \cdot n_k + n_k \cdot n_{k-1}^* \]

Differential detection doubles the amount of noise

\[ P(e) = \frac{1}{2} e^{\frac{E}{N_0}} \]
Receiver Block Diagram for DPSK

- **LO**
  - $r(t)$
  - $\cos(2\pi f_c t)$
  - $\sin(2\pi f_c t)$

- **Matched Filter**
  - $t = T$
  - $t = T$

- **Delay by $T$**

- **Phase Comparator**
Coherent BPSK Vs. DPSK

![Graph showing error probability vs. SNR for Binary DPSK and BPSK]

- **Binary DPSK**
- **BPSK**
Quadrature Amplitude Modulation (QAM)

Used in some high bandwidth-efficiency applications such as telephone modems, and microwave radio.

Requires both good carrier and amplitude control.

“Square” and “Cross” constellations
Performance of QAM in AWGN

\[ P(e) = 1 - (1 - p)^2 \]

\[ p = \left(1 - \frac{1}{\sqrt{M}}\right) \cdot \text{erfc}\left(\sqrt{\frac{3}{2(M - 1)} \cdot \frac{E}{N_0}}\right) \]

\( \bar{E} = \text{Average Energy} \)

For square constellations

Gains of \( M \)-QAM Vs. \( M \)-PSK

<table>
<thead>
<tr>
<th>( M )</th>
<th>dB Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1.65</td>
</tr>
<tr>
<td>16</td>
<td>4.20</td>
</tr>
<tr>
<td>32</td>
<td>7.02</td>
</tr>
<tr>
<td>64</td>
<td>9.95</td>
</tr>
</tbody>
</table>
BER of QAM Constellations

M=4  M=16  M=64
Binary Frequency-Shift Keying (FSK)

\[ S_1(t) = A \cos \left( 2\pi \left( f_c - \frac{\Delta f}{2} \right) \cdot t \right) \]

\[ S_2(t) = A \cos \left( 2\pi \left( f_c + \frac{\Delta f}{2} \right) \cdot t \right) \]

For a rate of \( \frac{1}{T} \) bits per s, \( \Delta f = \frac{1}{T} \) results in orthogonal signaling

Allows coherent and incoherent detection
Incoherent Detection of FSK

\[
\ell_i = \left[ \int_0^T r(t) \cdot \cos(2\pi f_it) dt \right]^2 + \left[ \int_0^T r(t) \cdot \sin(2\pi f_it) dt \right]^2
\]

\[f_i = f_c + \frac{2i - 3}{2T}, \quad i = 1, 2\]
Performance of Binary FSK

Incoherent detection

\[ P(e) = \frac{1}{2} e^{-\frac{E}{2N_0}} \]

Coherent detection

\[ P(e) = \frac{1}{2} \text{erfc} \left( \frac{E}{\sqrt{2N_0}} \right) \]
Coherent FSK Vs. BPSK

![Graph showing the comparison between Coherent FSK and Binary PSK in terms of Probability of Error vs. Signal-to-Noise Ratio (SNR) in dB. The graph illustrates that at higher SNR values, Binary PSK has a lower Probability of Error compared to Coherent FSK.]
Bandwidth Considerations

- Besides good error-rate performance, signals must also use as small a bandwidth as possible, for the data-rate they produce.
- FCC imposes strict limitations on the bandwidth occupancy of transmitted signals.
- For a given channel, there is a maximum data rate known as channel capacity.
Spectrum of Modulated Signals

\[ S(t) = \sum_{k=-\infty}^{\infty} a_k \cdot p(t - kT), \quad \text{PAM Signal} \]

\[ R_S(t + \tau, t) = \sum_{k=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} E[a_k a_n] \cdot p(t - kT)p(t + \tau - nT) \quad \text{Periodic in } t \]

\[ \bar{R}_S(\tau) = \sum_{k=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} E[a_k a_n] \cdot \frac{1}{T} \int_{-T/2}^{T/2} p(t - kT)p(t + \tau - nT) dt \]

\[ = \frac{1}{T} \sum_{k=-\infty}^{\infty} R_a(k) \cdot R_p(\tau - kT) \]

\[ R_a(k) = E[a_n \cdot a_{k+n}] \quad \text{(for WSS data sequences)} \]

\[ R_p(\tau) = \int_{-\infty}^{\infty} p(t) \cdot p(t + \tau) dt \]
The Power Spectrum

\[ \bar{R}_S = \frac{1}{T} \sum_{k=-\infty}^{\infty} R_a(k) \cdot R_p(\tau - kT) \]

\[ R_a(k) = E[a_n \cdot a_{k+n}] \]

\[ R_p(\tau) = \int_{-\infty}^{\infty} p(t) \cdot p(t + \tau) dt \]

\[ S_S(f) = \frac{1}{T} S_a(f) \cdot |P(f)|^2 \]

\[ S_a(f) = \sum_{k=-\infty}^{\infty} R_a(k) \cdot e^{-j2\pi fkT} \]

\[ \leftrightarrow \]

\[ R_a(k) = T \int_{-1/2T}^{1/2T} S_a(f) \cdot e^{j2\pi fkT} df \]

The spectrum of a modulated signal depends on the transmitted pulse-shape and the statistical properties of the transmitted symbols.
Example Power-Spectra

If $E[a_k] = 0$, $E[a_k^2] = \sigma^2$ and we have independent symbols

$$R_a(k) = \begin{cases} \sigma^2, & \text{if } k = 0 \\ 0, & \text{otherwise} \end{cases}$$

$$S_s(f) = \frac{1}{T} \sigma^2 \cdot |P(f)|^2$$

$$|P(f)|^2 = ET \left[ \frac{\sin(\pi f T)}{\pi f T} \right]^2$$
Signaling Through Bandlimited Channels

- *Intersymbol interference* (ISI) introduces memory into the received bit-stream
- There is a need for *sequence estimation*, instead of symbol by-symbol detection
- Bandlimited signals cannot be time-limited

\[
S(t; a) \xrightarrow{\text{Ideal Bandlimited Channel}} r(t) + n(t)
\]

\[a \equiv \text{Transmitte d symbol sequence}\]
The Likelihood-Function

\[ S(t; a) = \sqrt{E} \sum_{k} a_k p(t - kT) \]

\[ \int_{-\infty}^{\infty} p^2(t) dt = 1 \quad a_k \in \{1, -1\} \]

\[ \ell(a) = \int_{-\infty}^{\infty} r(t) \cdot S(t; a) dt - \frac{1}{2} \int_{-\infty}^{\infty} S^2(t; a) dt \]

\[ = \sqrt{E} \sum_{k} a_k \int_{-\infty}^{\infty} r(t) p(t - kT) dt - \frac{E}{2} \sum_{k} \sum_{j} a_k a_j \int_{-\infty}^{\infty} p(t - kT) p(t - jT) dt \]

\[ = \sqrt{E} \sum_{k} a_k \cdot r_k - \frac{E}{2} \sum_{k} \sum_{j} a_k a_j x[(k - j)T] \]

\[ x(t) = \int_{-\infty}^{\infty} p(\alpha) p(\alpha + t) d\alpha \]

\[ r_k = \int_{-\infty}^{\infty} r(t) p(t - kT) dt \]
Condition for no ISI

\[ \ell(a) = \sqrt{E} \sum_k a_k \cdot r_k - \frac{E}{2} \sum_k \sum_j a_k a_j x[(k - j)T] \]

If \( x(kT) = \begin{cases} 1, & k = 0 \\ 0, & \text{otherwise} \end{cases} \), then \( \rightarrow \) Nyquist pulses

Not a function of \( a \)

\[ \ell(a) = \sqrt{E} \sum_k a_k \cdot r_k - \frac{E}{2} \sum_k a_k^2 \] \( \rightarrow \) Bit-by-bit detection

Nyquist pulses result in no ISI at the sampling instants
In the frequency-domain, Nyquist’s criterion becomes:

$$\sum_k X\left(f + \frac{k}{T}\right) = T$$

**Example of a Nyquist pulse:**

[Diagram showing a Nyquist pulse with frequency axis and amplitude axis, with a series of pulses spaced at intervals of T.]
Signal Design for no ISI

Problem: We would like to design Nyquist pulses bandlimited to $W$ Hz.

$$x(t) = \sum_{k} x\left(\frac{k}{2W}\right) \cdot \frac{\sin[\pi(2Wt - k)]}{\pi(2Wt - k)}$$

Let $1/T = 2W$. Then

$$x(t) = \sum_{k} x(kT) \cdot \frac{\sin\left[\pi\left(\frac{t}{T} - k\right)\right]}{\pi\left(\frac{t}{T} - k\right)} = \frac{\sin\left[\pi\left(\frac{t}{T}\right)\right]}{\pi\left(\frac{t}{T}\right)}$$

Impractical to implement due to sync errors and filter-design limitations
Raised-Cosine Pulses: Frequency-domain

\[ X(f) = \begin{cases} 
T, & 0 \leq |f| \leq (1 - \alpha) / 2T \\
\frac{T}{2} \left[ 1 + \cos \left( \frac{\pi \frac{t}{\alpha} \left( |f| - \frac{1-\alpha}{2T} \right)}{} \right) \right], & \frac{1-\alpha}{2T} \leq |f| \leq \frac{1+\alpha}{2T} \\
0, & \text{elsewhere}
\end{cases} \]

\( \alpha \) is the "rolloff - factor"
Raised-Cosine Pulses: Time-domain

\[ x(t) = \frac{\sin(\pi t / T)}{\pi t / T} \cdot \frac{\cos(\alpha \pi t / T)}{1 - 4\alpha^2 \frac{t^2}{T^2}} \]

\[ \alpha = 0 \]

\[ \alpha = 0.5 \]

\[ \alpha = 1 \]
Square-Root Raised-Cosine Signaling

\[
X(f) = |P(f)|^2 \Rightarrow |P(f)| = \sqrt{X(f)}
\]

When the channel is non-ideal, **equalization** is needed to avoid ISI (we will discuss equalization later)
If the channel is not ideal, but known, and the noise is not white, then we can design the optimum transmit and receive filters to maximize the SNR for bit-by-bit decisions.

Transmit Filter, $P(f)$

Channel $C(f)$

Receive Filter, $H(f)$

Data

$P(f) \cdot C(f) \cdot H(f) = G_{rc}(f)e^{-j2\pi f_0 t}, \quad |f| \leq W$

Condition for no ISI

$r_k = \sqrt{E} \cdot a_k + n_k$, where $a_k \in \{1,-1\}$ and $n_k \sim N(0, \sigma^2)$

$P(e) = \frac{1}{2} \text{erfc}\left(\sqrt{\frac{E}{2\sigma^2}}\right)$
Optimization of Transmit/Receive Filters (cont’d)

\[ E_t = E \int_{-\infty}^{\infty} p^2(t) dt = E \int_{-\infty}^{\infty} |P(f)|^2 df \Rightarrow \]

\[ E = \frac{E_t}{\int_{-\infty}^{\infty} |P(f)|^2 df} = \frac{E_t}{\int_{-\infty}^{\infty} \frac{|G_{rc}(f)|^2}{|C(f)|^2 |H(f)|^2} df} \]

\[ \sigma^2 = \int_{-\infty}^{\infty} S_n(f) \cdot |H(f)|^2 df \]

\[ \frac{\sigma^2}{E} = \frac{1}{E_t} \int_{-\infty}^{\infty} S_n(f) \cdot |H(f)|^2 df \cdot \int_{-\infty}^{\infty} \frac{|G_{rc}(f)|^2}{|C(f)|^2 |H(f)|^2} df \]

\[ \geq \frac{1}{E_t} \left[ \int_{-\infty}^{\infty} \sqrt{S_n(f)} \cdot |H(f)| \cdot \frac{|G_{rc}(f)|}{|C(f)||H(f)|} df \right]^2 \]

(Cauchy - Schwartz inequality)
Optimization of Transmit/Receive Filters (cont’d)

Condition for equality
\[ \sqrt{S_n(f)} \cdot |H(f)| = K \frac{|G_{rc}(f)|}{|C(f)||H(f)|} \Rightarrow \]

\[ \therefore |H(f)| = K \frac{\sqrt{|G_{rc}(f)|}}{4\sqrt{|S_n(f)|}\sqrt{|C(f)|}}, \quad |f| \leq W \]

\[ \therefore |P(f)| = \frac{1}{K} \frac{\sqrt{|G_{rc}(f)|} \cdot 4\sqrt{|S_n(f)|}}{\sqrt{|C(f)|}}, \quad |f| \leq W \]

\[ \left( \frac{E}{\sigma^2} \right)_{\text{max}} = \frac{E_t}{\left[ \int_{-W}^{W} \frac{\sqrt{S_n(f)} \cdot |G_{rc}(f)|}{|C(f)|} df \right]^2} \]
Partial-response Signaling

- Introduce *controlled* ISI (duobinary signaling)

\[
x(nT) = \begin{cases} 
1, & n = 0, 1 \\
0, & \text{otherwise} 
\end{cases}
\]

\[
x(t) = \sum_{k} x\left(\frac{k}{2W}\right) \frac{\sin[\pi(2Wt - k)]}{\pi(2Wt - k)}
\]

\[
= \frac{\sin(2\pi Wt)}{2\pi Wt} + \frac{\sin[\pi(2Wt - 1)]}{\pi(2Wt - 1)}
\]
Frequency Response

\[ X(f) = \begin{cases} 
\frac{1}{W} e^{-j2\pi f \frac{f}{W}}, & |f| \leq W \\
0, & \text{otherwise}
\end{cases} \]
The Likelihood-Function

\[ \ell(a) = \sqrt{E} \sum_k a_k \cdot r_k - \frac{E}{2} \sum_k \sum_j a_k a_j x[(k - j)T] \]

\[ = \sqrt{E} \sum_k a_k \cdot r_k - \frac{E}{2} \sum_k \left(a_k^2 + a_{k-1} \cdot a_k\right) \]

Requires sequence estimation for optimum performance
(The Viterbi algorithm is an efficient way of implementing)
Symbol-by-Symbol Detection

\[ r_k = \int_{-\infty}^{\infty} r(t) p(t - kT) dt = \sum_i a_i \int_{-\infty}^{\infty} p(t - iT) p(t - kT) dt + \int_{-\infty}^{\infty} n(t) p(t - kT) dt \]

\[ = \sum_i a_i x[(i - k)T] + n_k \Rightarrow r_k = (a_{k-1} + a_k) + n_k \]

Precoding:

\[ w_k = b_k \oplus w_{k-1} \Leftrightarrow b_k = w_k \oplus w_{k-1} \]

\[ w_k = 0 \Leftrightarrow a_k = -1 \]

\[ w_k = 1 \Leftrightarrow a_k = 1 \]

\[
\begin{array}{c|c|c|c|c}
 w_{k-1} & w_k & a_{k-1} & a_k & b_k \\
\hline
 0 & 0 & \text{-1} & -1 & 0 \\
 0 & 1 & -1 & 1 & 1 \\
 1 & 0 & 1 & -1 & 1 \\
 1 & 1 & 1 & 1 & 0 \\
\end{array}
\]

\[ |r_k| \begin{cases} 
\leq 1 & \text{if } b_k = 0 \\
> 1 & \text{if } b_k = 1 \end{cases} \]

\[ p(r_k|a_{k-1}, a_k) \Rightarrow r_k (a_k + a_{k-1}) - a_k a_{k-1} \]

\[
\begin{array}{c|c|c}
 b_k = 0 & b_k = 1 & b_k = 0 \\
\hline
 -1, -1 & -2r_k - 1 \\
 1, 1 & 2r_k - 1 \\
 1, -1 & 1 \\
 -1, 1 & 1 \\
\end{array}
\]
Channel Coding

Source → A/D Conversion → Source Encoder → Channel Encoder → Modulator

Channel

Synchronization

User → D/A Conversion → Source Decoder → Channel Decoder → Demod
Introduction

- Channel coding is used to improve error-probability performance.
- It adds redundant bits into the data stream in such a way that they may be used to correct some (hopefully all) of the errors made during transmission.
- Price to pay for improved performance is reduction in information rate.
Categories of Coding

Block codes map $k$ information bits into $N$ channel bits, where $N > k$.

Convolutional codes are finite state machines that take as inputs binary sequences and yield output sequences through convolution with the code generator sequences.
Block Coding

For an \((N,k)\) code, the rate, \(R\), is:

\[
R = \frac{k}{N}
\]

We would like to have \(R\) as close to one as possible, and the code to correct as many errors as possible.
Block Coding (cont’d)

- The performance of the code is a function of the *minimum Hamming distance* between codewords (i.e., the number of bits they differ at).

- If the minimum Hamming distance for a code is $d_{\text{min}}$, the code can correct

$$t = \left\lfloor \frac{d_{\text{min}} - 1}{2} \right\rfloor$$

errors and detect $(d_{\text{min}} - 1)$ errors.
Hard Vs Soft-Decision Decoding

- Soft-decision decoding: Complex, better performance
- Hard-decision decoding: Simple, performance penalty

The optimum hard-decisions decoder is a minimum Hamming distance decoder
Errors can be made only if the channel flips \((t+1)\) or more bits.

\[
t = \left\lfloor \frac{d_{\text{min}} - 1}{2} \right\rfloor
\]

\[
P(e) \leq \sum_{k=t+1}^{N} \binom{N}{k} \cdot p^k (1 - p)^{N-k}
\]

\(p\) is a function of the modulation, detection method and the SNR *per information bit*.
Example 1: Block coding

A simple repeat code:

\[ 0 \leftrightarrow 000 \]
\[ 1 \leftrightarrow 111 \]

The code can correct 1 error and detect 2 errors. The rate of the code is 1/3 (very bad!!)

Decoding:

\[ \{000,001,010,100\} \leftrightarrow 0 \]
\[ \{111,110,101,011\} \leftrightarrow 1 \]
**Encoding of Linear Block Codes**

**Definition:** A block code is said to be **linear** when the sum of any two codewords is another codeword.

Linear block codes are described by a *generator matrix*, which consists of $K$ linearly independent $N$-dimensional binary vectors:

$$
G = \begin{bmatrix}
g_{00} & g_{01} & g_{02} & \cdots & g_{0(N-1)} \\
g_{10} & g_{11} & g_{12} & \cdots & g_{1(N-1)} \\
g_{20} & g_{21} & g_{22} & \cdots & g_{2(N-1)} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
g_{(K-1)0} & g_{(K-1)1} & g_{(K-1)2} & \cdots & g_{(K-1)(N-1)}
\end{bmatrix} \quad K \times N
$$

$$
v = m \cdot G
$$
Example: Encoding

\[ m = (m_0, m_1, m_2) \quad G = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \]

\[ v = m \cdot G = [m_0, (m_0 + m_2), m_1, (m_0 + m_1), (m_0 + m_1 + m_2), m_2] \]
Linear Systematic Codes

Systematic code codeword structure

\[ (N-k) \text{ parity bits} \quad \text{(N-k)} \quad k \text{ information bits} \]

\[ G = \begin{bmatrix} P & I \end{bmatrix} \]

\[ K \times (N - K) \quad K \times K \quad \text{identity} \]

The Parity Check Matrix:

\[ H = \begin{bmatrix} I & P^T \end{bmatrix} \]

\[ (N - K) \times (N - K) \quad (N - K) \times K \]
Syndrome Decoding of Linear Codes

Property of the Parity-Check Matrix:
*If \( \mathbf{v} \) is any codeword, then:*

\[
\mathbf{v} \cdot \mathbf{H}^T = 0
\]

Received vector = \( \mathbf{v} + \mathbf{e} \)

Error vector

Codeword

Where \( \mathbf{e} \) contains “1”’s the corresponding codeword bits are flipped

\[
\mathbf{S} = \mathbf{r} \cdot \mathbf{H}^T = (\mathbf{v} + \mathbf{e}) \cdot \mathbf{H}^T = \mathbf{v} \cdot \mathbf{H}^T + \mathbf{e} \cdot \mathbf{H}^T
\]

\[
\mathbf{S} = \mathbf{e} \cdot \mathbf{H}^T
\]
Syndrome Decoding (cont’d)

\[ S = e \cdot H^T \]

\((N-K)\) equations for \(N\) unknowns
There are \(2^K\) solutions

- Most likely error pattern contains the smallest number of errors: \textit{minimum weight} solution for \(e\)
- For small codes, minimum weight solutions can be obtained off-line and stored

\[ \hat{y} = r + \hat{e} \]

Decoded Codeword

Minimum-weight estimate
Example

Consider the \((7,4)\) Hamming code (which can correct 1 error). Let \(v = (1\ 0\ 0\ 1\ 0\ 1\ 1)\) be transmitted and \(r = (1\ 0\ 0\ 1\ 0\ 0\ 1)\) received (which contains one error)

\[
H = \begin{bmatrix}
1 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 & 1 \\
\end{bmatrix}
\]

\[
S = e \cdot H^T = (1\ 1\ 1)
\]

\[
1 = e_0 + e_3 + e_5 + e_6
\]

\[
1 = e_1 + e_3 + e_4 + e_5 \quad \Rightarrow \quad \hat{e} = (0\ 0\ 0\ 0\ 0\ 0\ 1\ 0)
\]

\[
1 = e_2 + e_4 + e_5 + e_6
\]

\[
\hat{v} = r + \hat{e} = (1\ 0\ 0\ 1\ 0\ 1\ 1)
\]
Example of Popular Block Codes

- Reed-Solomon Codes
  - Powerful
  - Used in compact-disc players and mobile radio/satellite applications
- Golay codes
- Reed-Muller codes
- BCH codes
- Hamming codes
Coded Vs Uncoded Performance

(64,127) BCH Code

Uncoded

(\(d_{\text{min}} = 21\))
**Convolutional Codes**

An \((n,k,m) = (2,1,2)\) Convolutional Code

\[ x \rightarrow g_1 = \begin{pmatrix} 1 & 0 & 1 \end{pmatrix} \rightarrow y_1 \]

\[ g_2 = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \rightarrow y_2 \]

\[ y \cdots 1101001001 \]

\[ \text{Generator sequences} \]

\[ g_1 = \begin{pmatrix} g_{10} & g_{11} & g_{12} \end{pmatrix} \]

\[ g_2 = \begin{pmatrix} g_{20} & g_{21} & g_{22} \end{pmatrix} \]

\[ R = \frac{k}{n} \quad \text{The rate of the code} \]

\[ (m + 1) \cdot n \quad \text{The constraint length of the code} \]
The Encoder

\[ y_j = x \star g_j, \quad j = 1, 2 \ldots, n \]

\[ y_{j\ell} = \sum_{i=0}^{m} x_{\ell-i} \cdot g_{ji} \]

Example:

\[ g_1 = (1 \quad 0 \quad 1) \quad y_{1\ell} = x_\ell + x_{\ell-2} \]

\[ g_2 = (1 \quad 1 \quad 1) \quad y_{2\ell} = x_\ell + x_{\ell-1} + x_{\ell-2} \]
Convolutional Codes can be described by a **state diagram**
The Trellis Diagram
Optimum Decoding of Convolutional Codes

- Maximum-likelihood (ML) decoding of convolutional codes can be efficiently done using the \textit{Viterbi algorithm}.

- The Viterbi Algorithm (for Hard-Decision Decoding)
  - Set the \textit{Hamming distance} for the initial state (00) to zero.
  - As data arrive, move to the next set of states.
  - For each state, compute the incurred \textit{Hamming distance} for each possible way to get to that state. Record the smallest such distance and save a pointer to the previous state for the “surviving” path.
    - \textbf{Metric computation:} For a given path, add the Hamming distance for the previous state to the Hamming distance incurred by the transition from the previous to the current state.
  - Repeat as more data arrive.
  - At some depth in the trellis (empirically 6 constraint lengths), choose the state with the smallest Hamming distance. The surviving path corresponding to that state is the ML estimate.
Practical Viterbi Decoding Considerations

- In practice, we use a sliding-window approach.
- The size of the window is about 6 constraint lengths.
- Once enough data is received to fill the window initially, then a first bit is released by backtracking.
- Thereafter, for every bit transmitted, the algorithm moves to the next set of states, finds the state with the smallest Hamming distance, uses it to backtrack and releases the bit that has been in the window the longest.
Soft-decision decoding is similar, but minimizes Euclidean distance instead.
The Transfer Function of a Convolutional Code

\[ T(D) = \frac{X_B}{X_A} = \frac{D^5}{1 - 2D} = D^5 + 2D^6 + 4D^7 + 8D^8 + 16D^9 + \cdots \]

\[ = \sum_{k=5}^{\infty} 2^{k-5} \cdot D^k \]

The free distance of the code

\[ d_{\text{free}} = 5 \]
Performance

\[ P(e) \leq T(D) \bigg|_{D = e^{-SNR}} \quad \text{Soft-decision Decoding} \]

\[ P(e) \leq T(D) \bigg|_{D = \sqrt{4p(1-p)}} \quad \text{Hard-decision Decoding} \]
Combined Modulation and Coding

- Combines modulation and coding for improved performance
- It is possible to obtain a coding gain without bandwidth expansion
- Trellis-coded modulation (TCM), introduced by G. Ungerboeck, has revolutionized coding for bandlimited channels