EE 577 - Wireless and Personal Communications

Chapter 7: Mitigation Techniques

Fading Mitigation Techniques

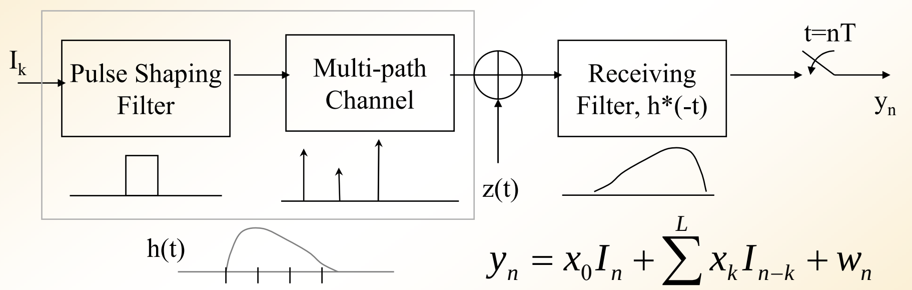
- **Equalization:** mitigates frequency-selective fading channels
- **Diversity:** mitigates flat fading
- **Channel Coding:** mitigates errors due to fading and noise, provide time diversity
- **Adaptive Modulation:** mitigates time-selective fading

Causes of ISI

- Channel Distortion
- **Multipath Fading:** Frequency selective channels act as an FIR filter causing channel-induced ISI
- Pulse shapes not designed for zero ISI (GMSK used in GSM)

Problems in FS Fading Channels

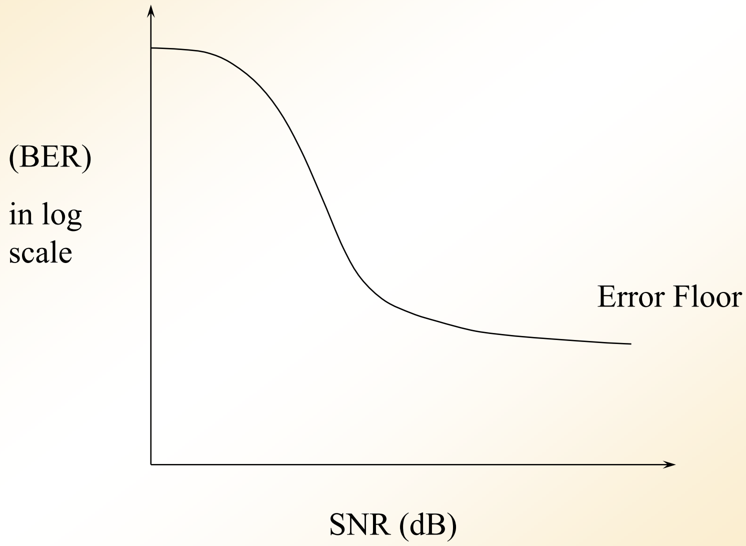
- Multi-path Channel model causes ISI at receiver



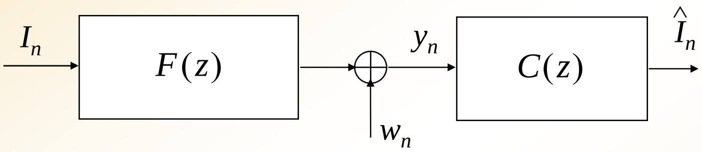
$$y_n = x_0 I_n + \sum_{k=1}^{L} x_k I_{n-k} + w_n$$

- As signal power S↑, ISI power ↑
- Hence, S/(I+N) decreases slowly with increasing S/N, independent of the signal power

Error Floor in FS Fading Channels



Equalization



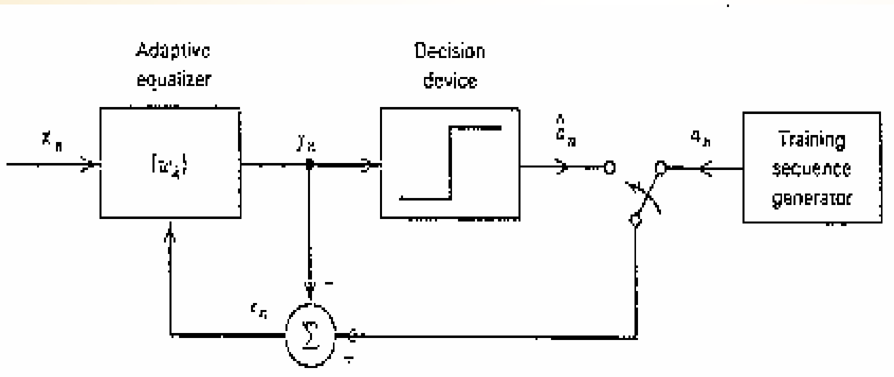
- An equalizer is a filter that equalizes the effect of the nonideal (frequency-selective) channel
- $C(f)\ H(f) = 1$ (Equalizer inverses the effect of the channel)
- For time-varying channels the equalizer has to be adaptive.

Equalizer Modes

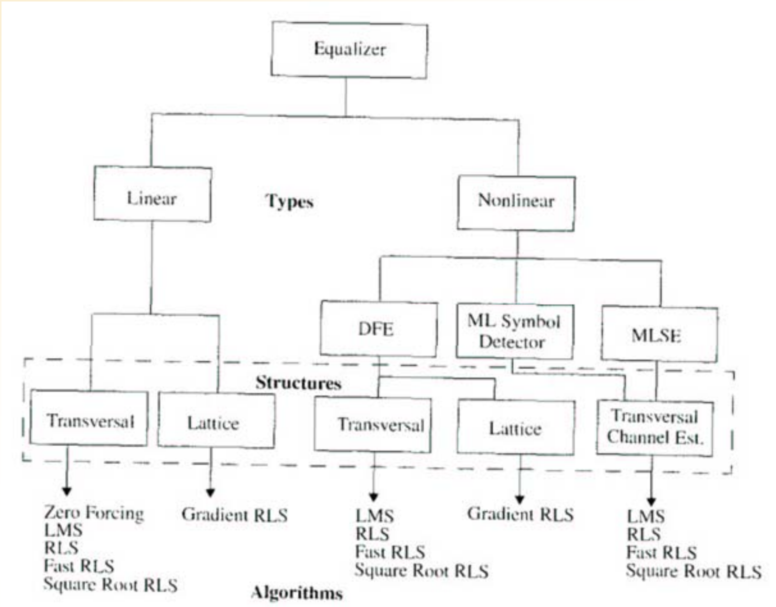
- **Training:**
  - The equalizer must periodically learn the channel by transmitting a training sequence
  - The training sequence is used at the receiver to choose the equalizer parameters
- **Tracking:**

  The equalizer parameters are adjusted based on the difference between the equalizer output and the output of the decision device

Training & Decision Modes



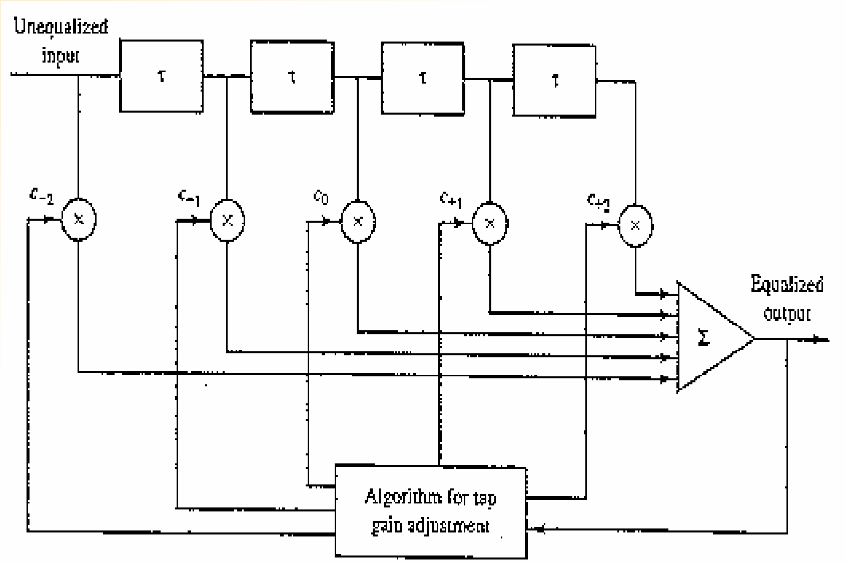
Classification of Equalizers



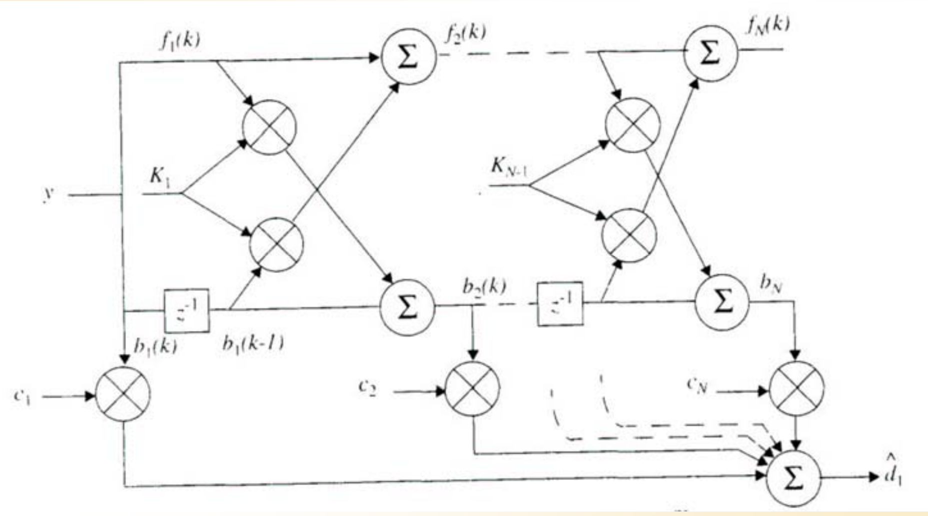
Classification of Equalizers

- **Linearity:**
  - Linear
  - Nonlinear (DFE, MLSE)
- **Structure:**
  - Transversal
  - Lattice
- **Algorithm:**
  - Zero Forcing
  - MMSE

Linear Transversal Equalizer



Lattice Implementation



Linear Equalizer

- Design $C(z)$ or c_n such that the output is as **close** to I_n as possible
- **Zero-Forcing:**
  - Forces the samples of combined channel/equalizer impulse response to be zero at all but one of the NT_s spaced samples
  - It is an impulse in time domain:

$$F(z)C(z) = 1$$

  - Simple, but may enhance the noise

Linear Equalizer

Minimum Mean Square Error (MMSE):

$$MSE = E\left[e_k^2\right] = E\left[(x_k - \hat{x}_k)^2\right]$$
$$= E\left[x_k^2\right] - 2p^T c + c^T Rc$$

$$y_k = \left[y_k \quad y_{k-1} \quad \cdots \quad y_{k-N}\right]^T$$
$$c = \left[c_0 \quad c_1 \quad \cdots \quad c_N\right]^T$$

$$\mathbf{p} = E\left[x_k \mathbf{y}_k^T\right] \qquad \mathbf{R} = E\left[\mathbf{y}_k \mathbf{y}_k^T\right]$$

- **To minimize MSE,**

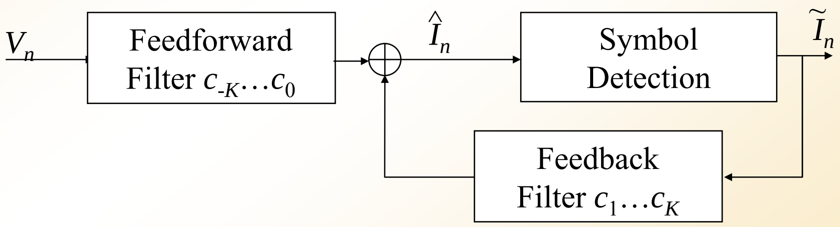
$$\frac{\partial E\left[e_k^2\right]}{\partial \mathbf{c}} = -2\mathbf{p} + 2\mathbf{R}\mathbf{c} = 0$$

$$\Rightarrow \quad \mathbf{c}_{opt} = \mathbf{R}^{-1}\mathbf{p}$$

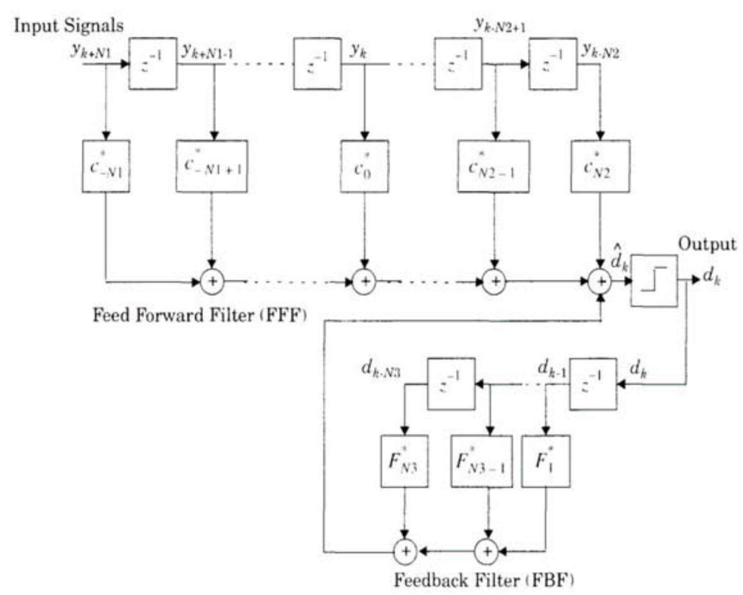
- To estimate **R** and **p**, the transmitter can transmit a training sequence that is known by the receiver.
- Equalizer requires periodic retraining in order to maintain effective ISI cancellation.

Decision Feedback Equalizer (DFE)

- DFE attempts to subtract from the current symbol the ISI created by previously detected symbols
- It performs better than linear equalizer always
- Suitable to channels with deep frequency nulls



DFE



DFE

- Consists of a feed-forward filter followed by a feedback filter with the bit decisions as input
- DFE does not suffer from noise enhancements
- Coefficients can be updated using LMS, RLS or MMSE algorithms

Performance Measures

- **Rate of convergence:**

  # iterations required for the equalizer to converge to the correct solution

- **Misadjustment:**

  Measure of the amount of deviation from the correct solution

- **Computational complexity:**

  # operations required to make one iteration in the equalizer

Adaptive Algorithms

- **Least Mean Square (LMS):**

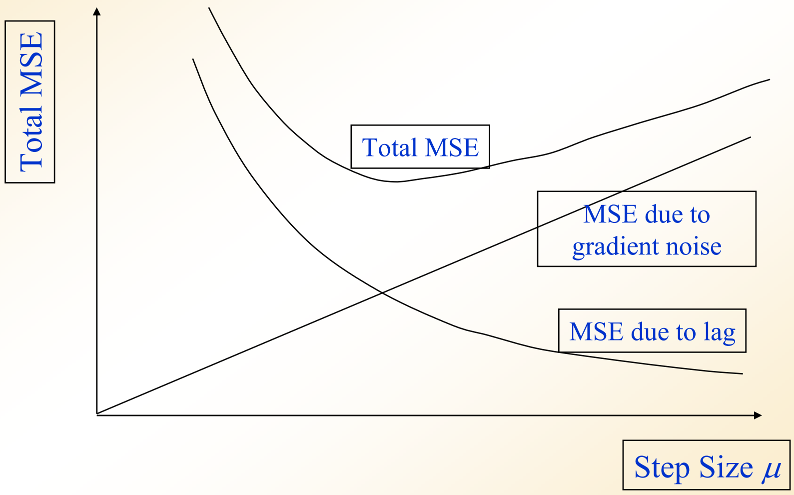
- To minimize: $$MSE = E\left[e_k^2\right] = E\left[\left(x_k - \hat{x}_k\right)^2\right]$$

- The algorithm is: $$\mathbf{C}^{(k+1)} = \mathbf{C}^{(k)} - \mu\nabla^{(k)}$$

  where $$\nabla^{(k)} = \frac{\partial E\left[e_k^2\right]}{\partial \mathbf{c}} = -2\mathbf{p} + 2\mathbf{Rc} = -2E[e_k \mathbf{y}_k]$$

- So, $$\mathbf{C}^{(k+1)} = \mathbf{C}^{(k)} + \mu e_k \mathbf{y}_k$$

MSE in LMS Algorithm



Training Algorithms Tradeoffs

- **LMS:** $2N+1$ multiply operations, low complexity, slow convergence, poor tracking
- **MMSE:** N^2 - N^3 multiply operations, very high complexity, very fast convergence, good tracking
- **RLS:** $2.5N^2 + 4.5N$ multiply operations, high complexity, fast convergence, good tracking

Maximum Likelihood Sequence Equalizer (MLSE)

- ISI introduces some form of memory (relation between adjacent samples over the span of ISI)
- Instead of detecting the received stream symbol-by-symbol like in previously discussed equalizers
- MLSE observes a sequence of received symbols and searches for the most likely transmitted sequence

MLSE

- Comparing the received sequence to all possible transmitted sequences is a very computational complex task
- An efficient algorithm of finding the most likely sequence without the need for comprehensive search is known as the **Viterbi Algorithm**
- MLSE is used in GSM

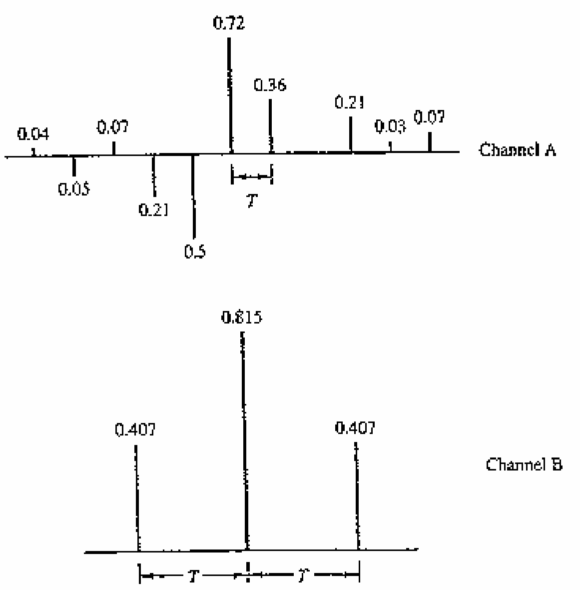
Fractionally-Spaced Equalizers (FSE)

- In previous equalizers, taps are separated by symbol duration T_s
- However, the pulse often extends to more than a symbol duration (such as in RC pulses)
- In this case FSE performs better
- In FSE, the taps are separated in time by the reciprocal of Nyquist rate ($<T_s$)
- FSE has better performance

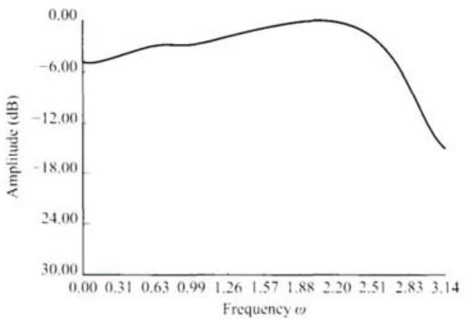
Equalizer Design

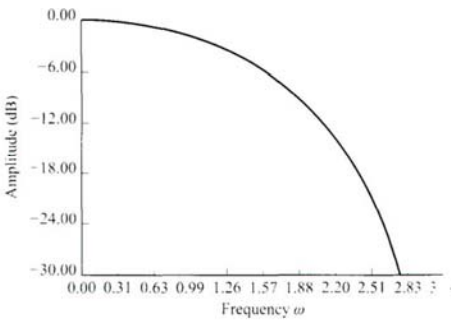
- Complexity has to be justified by SNR gain and battery savings
- Coherence time should be greater than equalizer convergence time
- Maximum number of resolvable multipath components in the channel dictates the number of taps in the equalizer
- An equalizer can equalize a channel with a maximum delay spread less than or equal the maximum delay in the equalizer

Case Study

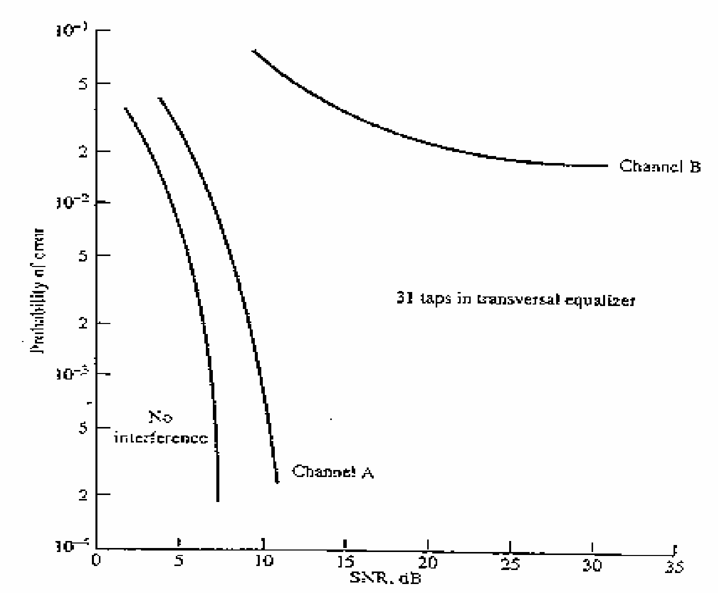


Study Case

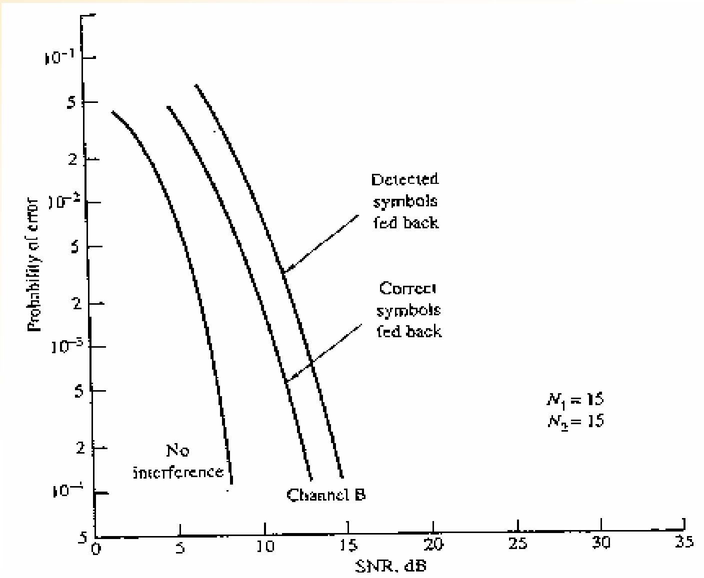




Performance of Linear Equalizers



Performance of DFE

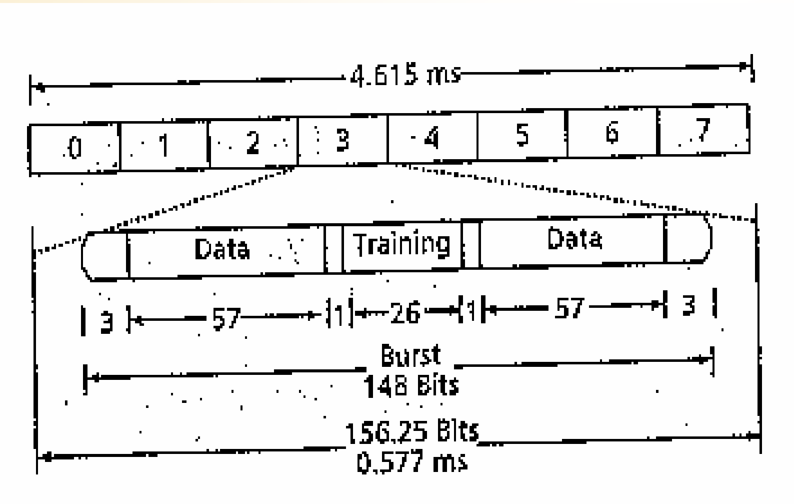


The GSM System

- Slot duration = 0.577 *ms*
- Bit duration: 3.69 μs
- Carrier freq. = 900 MHz
- W = 200 kHz
- T_m = 16 μs
- **Assumption:**

  Speed 100 km/hr (55.56 m/s)

Case Study: The GSM System



The GSM System

- Channel parameters:

  $T_c \approx 3\ ms$ $\Rightarrow$ slowly fading

  $B_c \approx 62.5$ kHz $\Rightarrow$ frequency selective

- Equalization is required
- Observation interval = 4-6 bits

EE 577 - Wireless and Personal Communications

Lecture 16: Diversity

Diversity

- **Basic idea:** send the same information over independently fading paths, then combine the paths
- If diversity branches are uncorrelated, the probability of deeply faded received signal is reduced

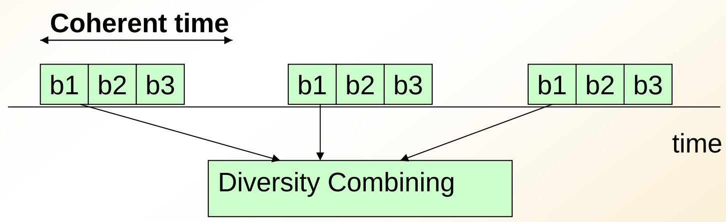
- **Macro Diversity:** provides a method to mitigate the effects of shadowing
- **Micro Diversity:** provides a method to mitigate the effects of multi-path fading

Diversity Approaches

- **Space Diversity:**
  - Using antennas spaced enough (at Tx or Rx)
- **Polarization Diversity:**
  - Using antennas with different polarizations
- **Frequency Diversity:**
  - Using frequency channels separated in frequency more than the channel coherence BW
- **Time Diversity:**
  - Using time slots separated in time more than the channel coherence time
- **Multi-path Diversity:**
  - Utilized efficiently in CDMA using RAKE receiver

Time Diversity

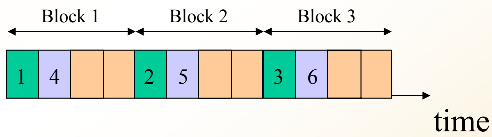
Interleave the repeated bits over a duration longer than the coherence time T_c



Block Interleaving

Example:

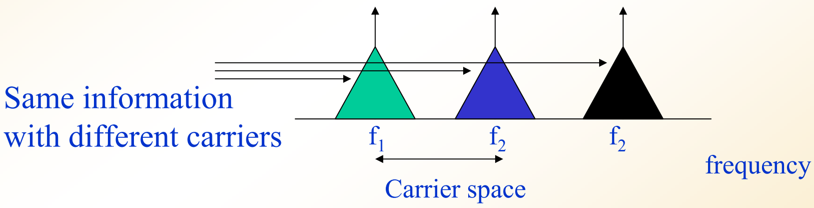
- If bit stream is 1 2 3 4 ….12
- After block interleaving,



- Less affected by burst errors

Frequency Diversity

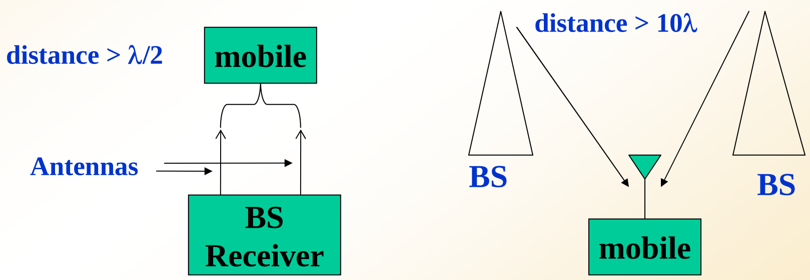
- Send the signal over multiple carriers separated in frequency by more than the **Coherence Bandwidth**



- FH-SS is a special case of frequency diversity

Space (Antenna) Diversity

- Use more than one antenna to receive the signal
- The distance between two antennas should:
  - exceed $\lambda/2$ at MS due to large amount of scatterers
  - around 10λ at BS due to less number of scatterers
- Very suitable for base station implementation



Polarization Diversity

- Use the same antenna to receive the signal
- Orthogonal polarizations (vertical and horizontal) are used to provide two diversity paths
- Reflection coefficients for vertical and horizontal polarized waves are different
- This causes orthogonally polarized waves to undergo uncorrelated fading
- Very suitable for fixed wireless links such as microwave links

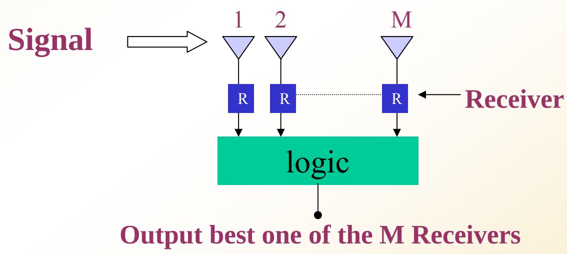
Combining Techniques

- Selection Combining (SC)
- Equal Gain Combining (EGC)
- Maximal Ratio Combining (MRC)
- Generalized SC (GSC)

MRC, EGC and GSD require a coherent phase reference to co-phase the different branch signals

Selection Combining

Key idea: Monitor **ALL** *M* branches at a time and select the branch with highest SNR to receive the signal



Selection Combining

- Let the average SNR in branch *i* be:

$$\bar{\gamma}_i = \bar{\gamma}$$

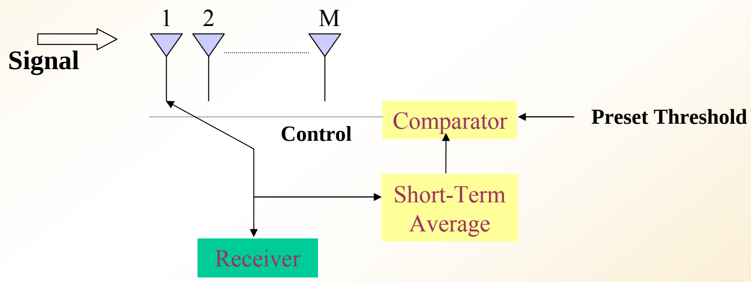
- The average SNR for Selection Combining is:

$$\bar{\gamma}_S = \bar{\gamma} \sum_{i=1}^{M} \frac{1}{i}$$

- The incremental gain becomes extremely small for large value of *M*

Scanning Diversity (Switched) Combining

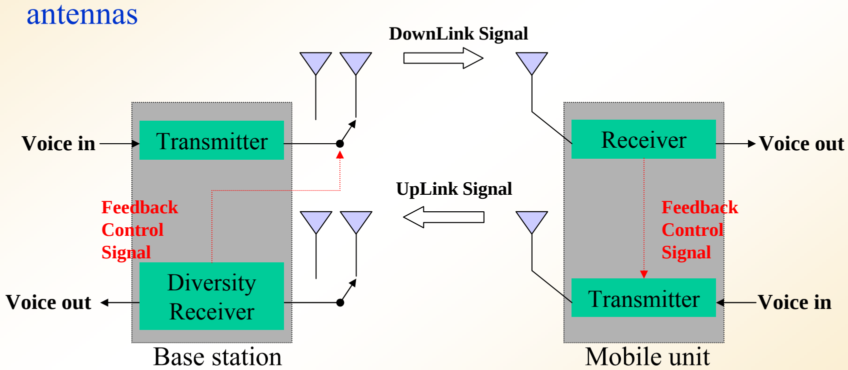
- Monitor **ONE** branch at a time.
- If signal quality of monitored branch falls below a threshold, the receiver scans other branches for better signal quality



- If threshold is large, the scanning process will be activated often
- If threshold is small, no improvement in diversity combining

Feedback Diversity Combining

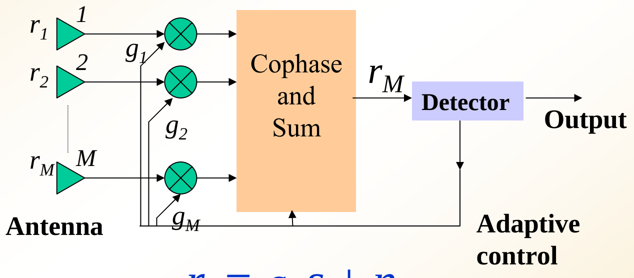
- Antenna switching is done at the BS
- Transmitter antennas are switched instead of the receiver antennas



- **Advantage:** Simplify the circuit complexity of mobile unit
- **Disadvantage:** It is not an optimal diversity technique

Maximal Ratio Combining (MRC)

- The signal from all branches are weighted and then summed together



$$r_i = a_i\, s + n_i$$

- where s is the transmitted signal
- n_i is the i-th noise process and g_i's are the weights

MRC

- The signals from each of the M branches are co-phased
- The resultant signal is represented as: $r_T = \sum_{i=1}^{M} g_i r_i$
- Let the average SNR in branch i be: $\overline{\gamma}_i = \frac{r_i^2}{2N_i}$
- The SNR can be written as: $\gamma = \frac{1}{2}\frac{\left(\sum_{i=1}^{M} g_i r_i\right)^2}{\sum_{i=1}^{M} g_i^2 N_i}$
- Maximization gives: $g_i = \frac{r_i}{N_i}$

MRC

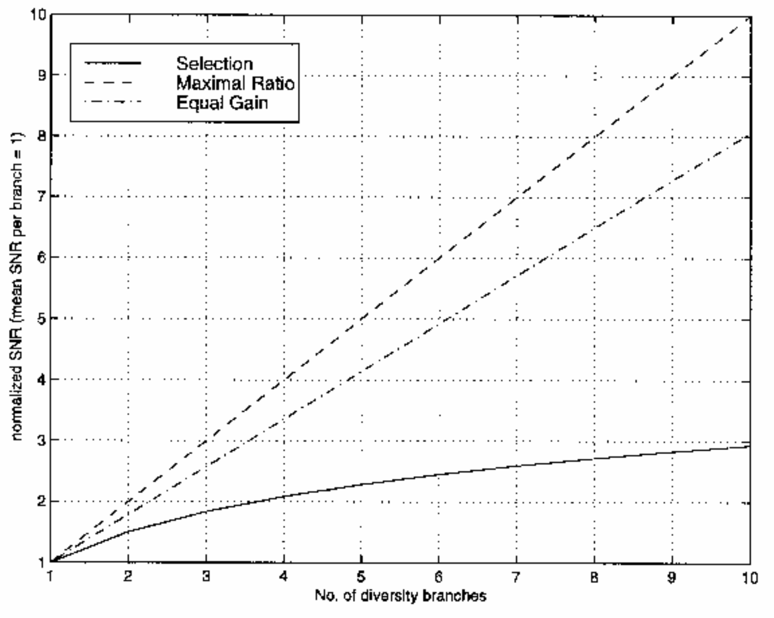
- Thus, the SNR becomes: $$\gamma_{MR} = \sum_{i=1}^{M} \gamma_i$$
- The mean SNR is given be: $$\bar{\gamma}_{MR} = M\bar{\gamma}$$
- **Advantage:** Produce an output with an acceptable SNR even when none of the individual branches are themselves acceptable
- **Disadvantage:** Channel estimation is required for each diversity gain

Equal Gain Combining (EGC)

- Similar to MRC with the weights a_i's are all equal to 1
- No need to estimate the channel gains for each diversity branch
- The average SNR is given by:

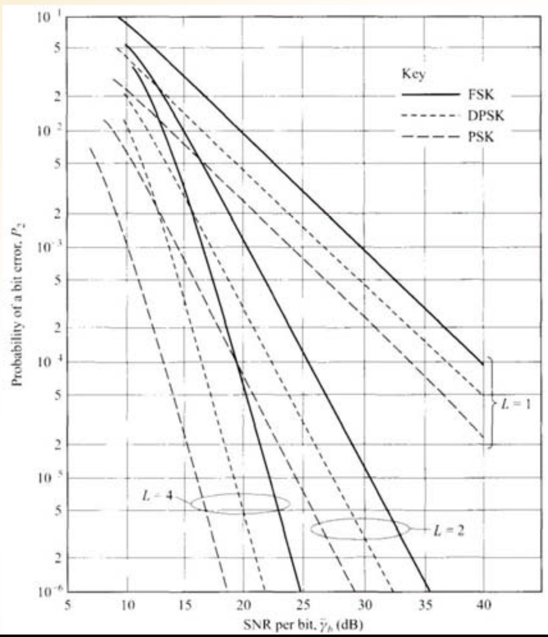
$$\bar{\gamma}_E = \bar{\gamma}\left(1 + \frac{(M-1)\pi}{4}\right)$$

Diversity SNR Gains



51

Diversity Performance



52

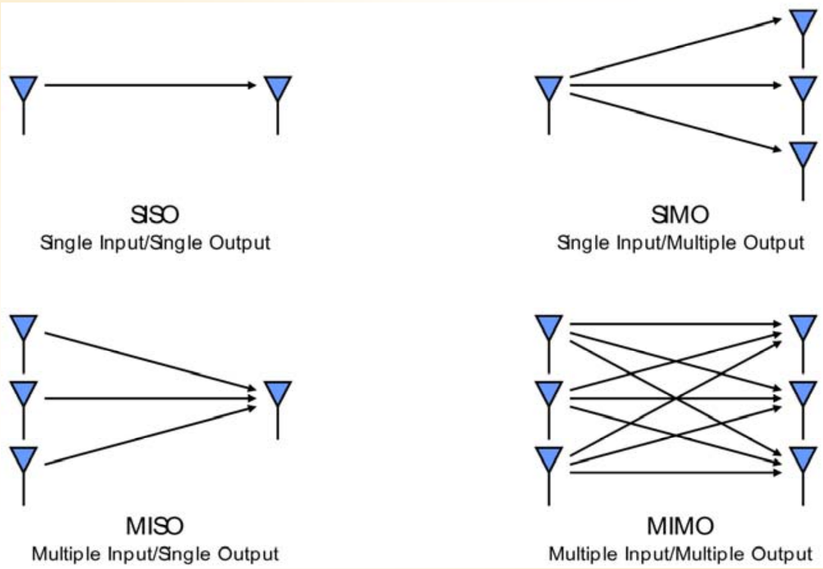
Generalized Selection Combining (GSC)

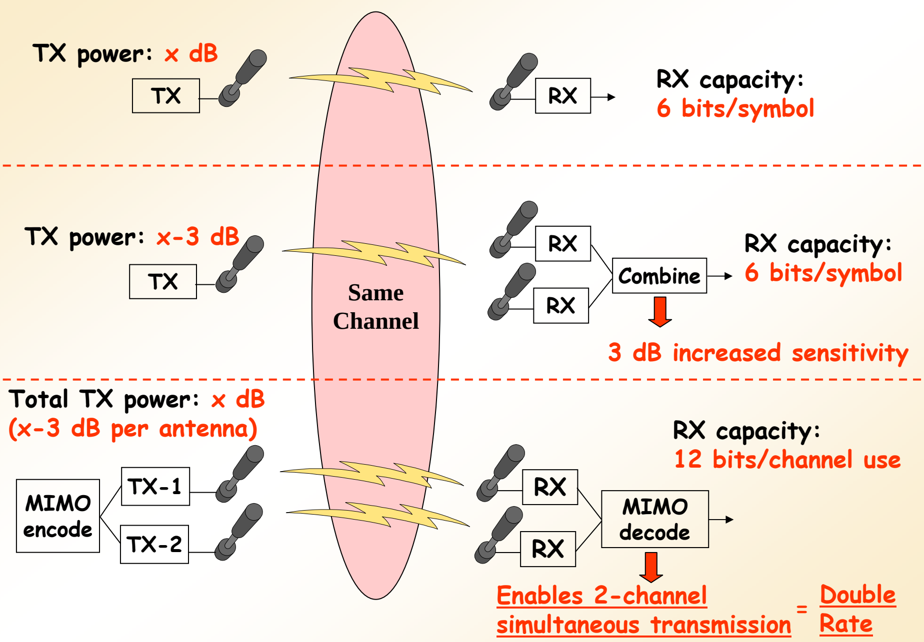
- Select the L diversity branches with the largest receive signal level (including noise and interference) among the M branches
- Combine the selected branches using MRC
- Provides a tradeoff between SC and MRC:
  - Performs better than SC
  - Less complexity than MRC
- Avoid noisy branches with small SNR values

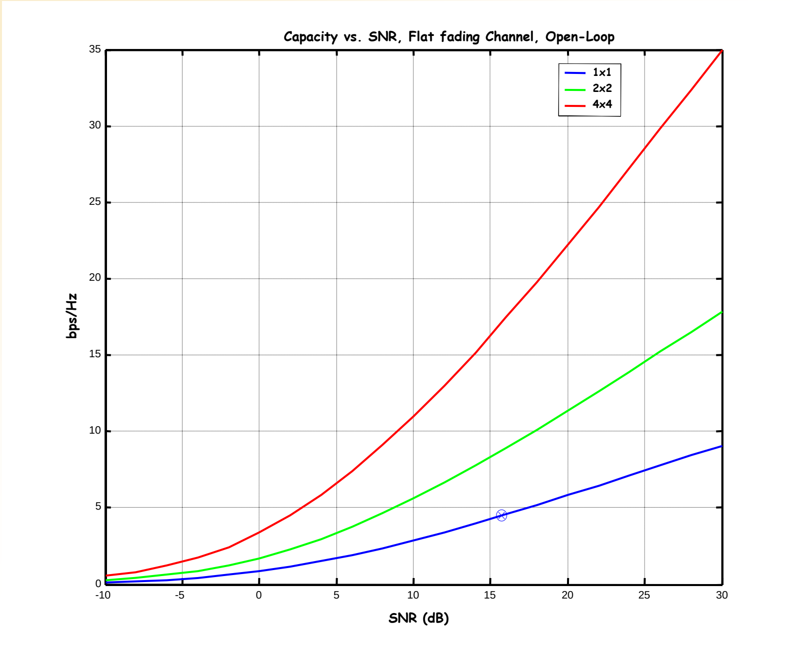
RAKE Receiver

- Used with DSSS systems
- Provides a means to combine resolvable multipath components as diversity branches
- The RAKE receiver works as follows:
  - Correlate the received signal with the PN sequence
  - Correlate with a delayed version of the PN sequence to capture the first delayed multipath finger
  - Repeat delay-and-correlate process until all multipath figures are captured
  - Combine the outputs of the correlators using MRC
- **Advantage:** Takes advantage of multipath
- **Disadvantage:** Needs several correlators

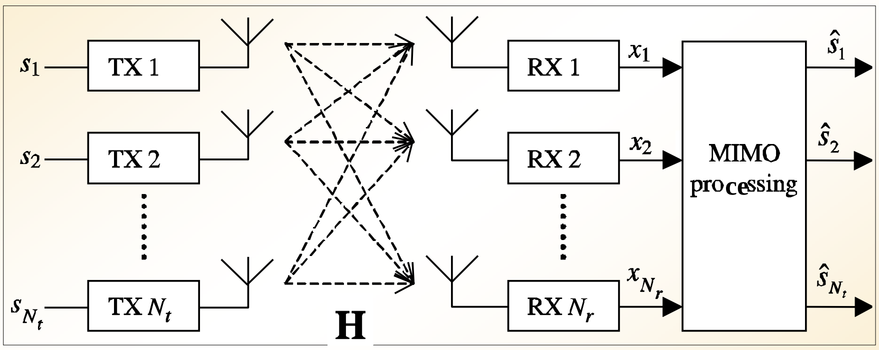
Multi-Input Multi-Output (MIMO)







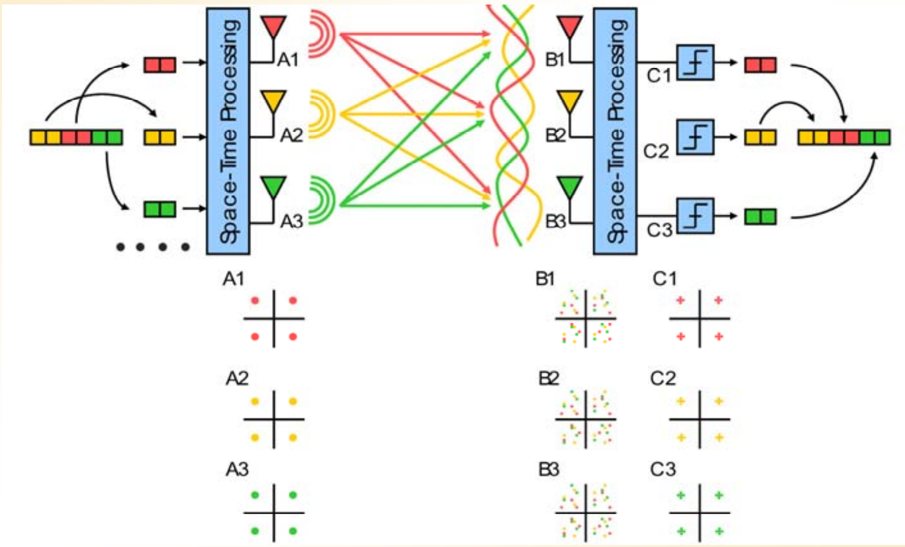
MIMO



H is N_t x N_r matrix

$$\mathbf{x} = \mathbf{s}\,\mathbf{H} + \mathbf{z}$$

Space-Time Coding



Advantages of MIMO

- Order of magnitude increased rate, range and robustness
- Good cost-performance trade-off and scalability
- Same bandwidth and higher rates

  => more efficient use of spectrum
- Increase downlink capacity
- Combats multipath fading
- Initial application in 3G WCDMA standard and 802.16 broadband wireless

MIMO Techniques

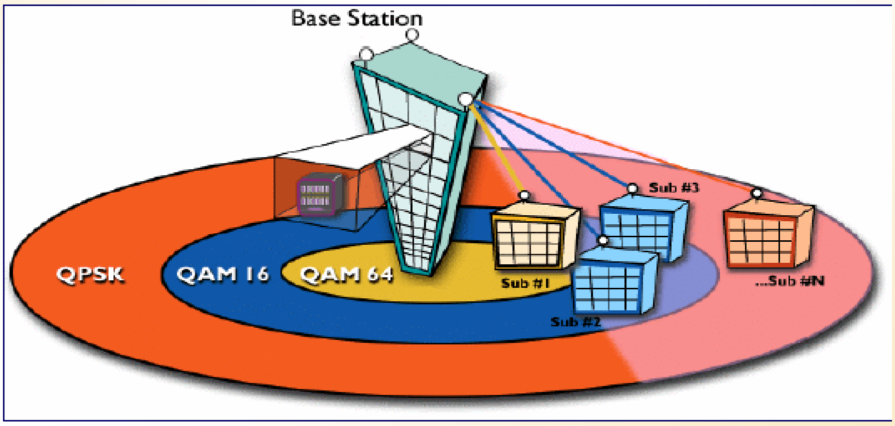
- **Open-Loop MIMO:**
  - Multiple coded data streams across multiple parallel transmitters
  - Achieve diversity gain by space-time coding and/or interference cancellation at the receiver
  - Linear increase in data rates with number of antennas
- **Closed-Loop MIMO:**
  - Waterfilling to achieve higher data rates
  - Transmitter requires channel knowledge
  - Data rates ***and*** range/throughput gains

Adaptive Modulation

- **Basic idea:**
  - Measure the channel at the receiver
  - Feed the measurement back to the transmitter
  - Adapt the transmission scheme relative to the channel estimate to **maximize the data rate**, **minimize transmit power** or **minimize BER**
- **What to adapt?**
  - Constellation size/power
  - Symbol time
  - Coding rate/scheme

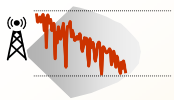
Bit Rates in IEEE 802.16a

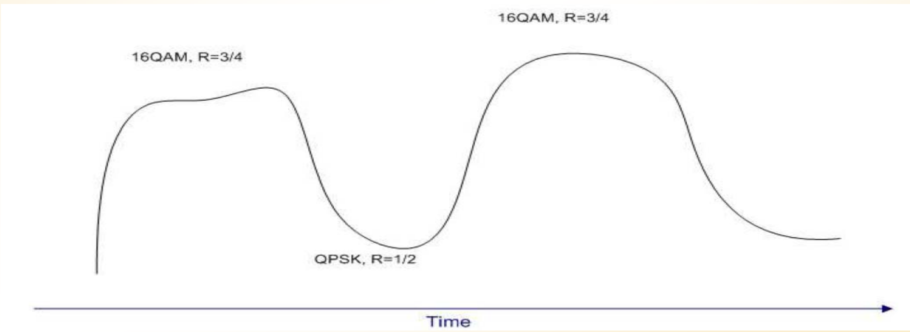
- ❑ Bit rate shifting is achieved using adaptive modulation.
- ❑ When you are near to the BS => offered high speed,
- ❑ When you are far, reliability decreases => offered lower speed.



HSDPA Features

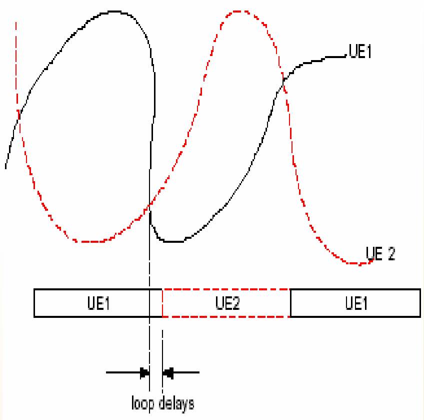
- ❑ Adaptive Modulation and Coding
  - ❑ Data rate adapted to radio conditions
  - ❑ 2 ms time basis





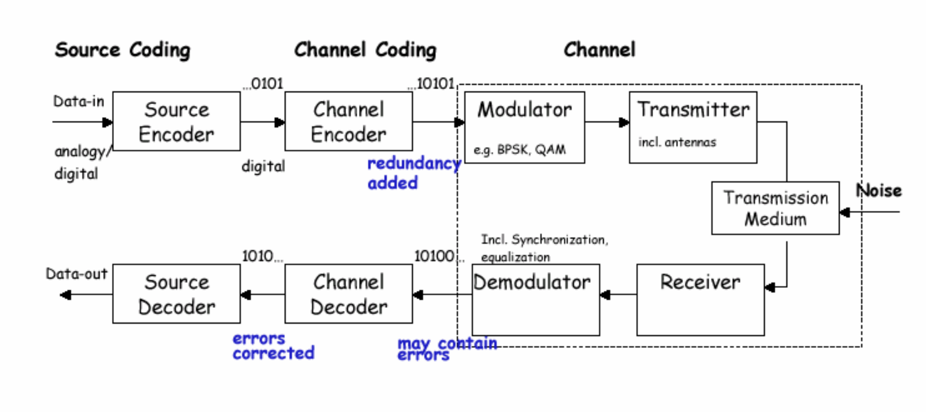
Multi-User Diversity

- Fast Scheduler
  - 2 ms time basis
  - Round Robin, Proportional Fair or Max-C/I
- Since users are independent with each other, let the users with good channel condition send at any given time ➔ Multiuser Diversity.
- Fairness is also an important attribute



Channel Coding

Communication System



Basic Channel Coding Concepts

Example: Binary Repetition Codes

- (3,1) code: 0 => 000, 1 => 111
- (3,1) repetition code can correct single errors
- Block error probability:

$$P_E = \binom{3}{2}(1-p)\,p^2 + \binom{3}{3}p^3$$

- **Gain:** For a BSC with $p = 10^{-2}$, $P_E = 3\text{x}10^{-4}$.
- **Cost:** Expansion in bandwidth by 3 times

Error Control Coding

- The function of the encoder is to introduce redundancy in the binary information sequence.
- Such redundancy is used in the receiver to overcome the effects of noise, interference and (fading) encountered through the channel.

- **Encoding** is the process of mapping k-bit information into a unique n-bit sequence called the "***codeword***"
- The ***code rate*** is defined as $R = k/n$

Shannon's Channel Capacity

- Shannon derived the capacity formula in 1948:

$$C = W \log_2(1 + \frac{S}{N})$$

  - W is the bandwidth in Hz
  - S is the signal power in watts
  - N is the total noise power
- The bandwidth efficiency can be found as:

$$\eta = \frac{\text{Transmission Rate}}{\text{Signal Bandwidth W}} \quad \text{[bits/s/Hz]}$$

$$\eta_{\max} = \log_2(1 + \frac{S}{N}) \quad \text{[bits/s/Hz]}$$

Shannon's Channel Capacity

- The average signal power: $S = \frac{kE_b}{T} = RE_b$

$$\eta_{\max} = \log_2\left(1 + \frac{RE_b}{N_0 W}\right)$$

  - E_b is energy per bit
  - k is the number of bits transmitted per symbol
  - T is the duration of a symbol
  - $R = k/T$ is the transmission rate in bits/s
  - $N = N_0W$ is the total noise power
  - N_0 is the one-sided noise power spectral density

Shannon's Channel Capacity

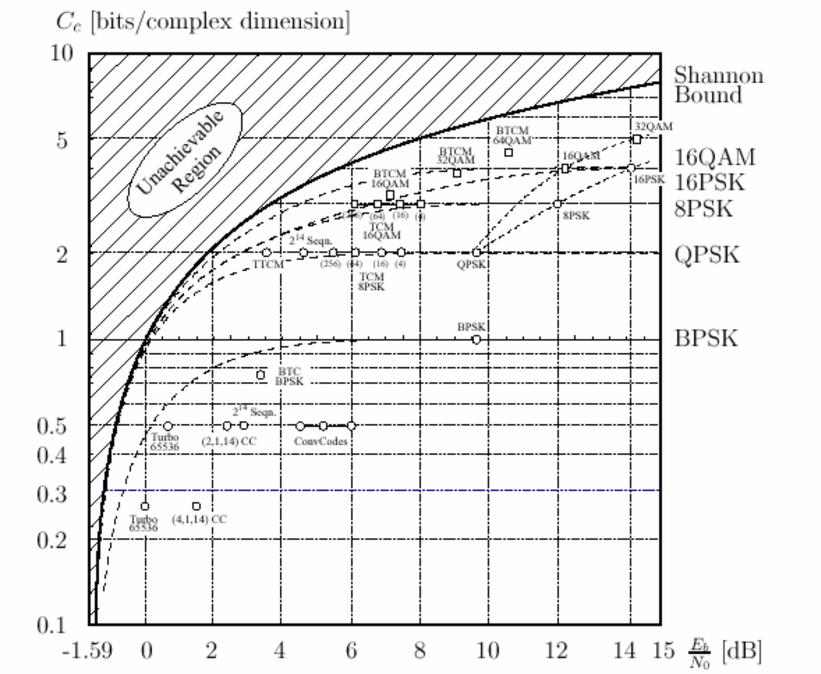
- The minimum bit energy required for reliable transmission (Shannon bound):

$$\frac{E_b}{N_0} \geq \frac{2^{\eta_{\max}} - 1}{\eta_{\max}}$$

- In the case of infinite bandwidth, i.e., $\eta_{max} \rightarrow 0$,

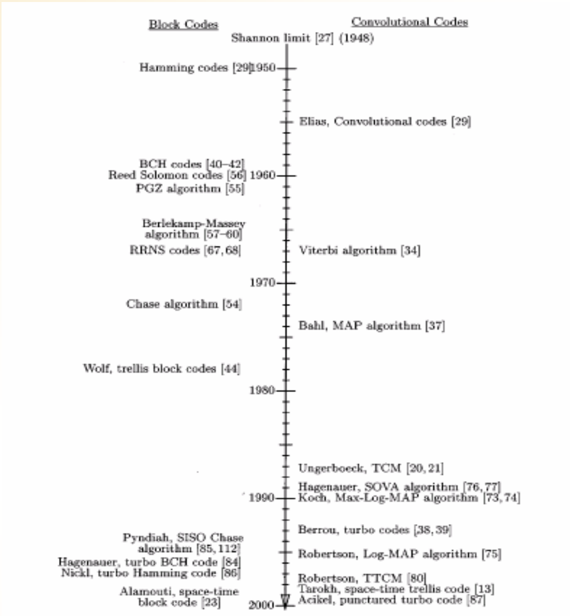
$$\frac{E_b}{N_0} \geq \lim_{\eta_{\max} \to 0} \frac{2^{\eta_{\max}} - 1}{\eta_{\max}} = \ln(2) = -1.59 \text{ dB}$$

- This is the **minimum signal-to-noise ratio** required to reliably transmit one bit of information



73

History of Channel Coding



74

Hamming Distance

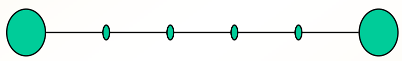
- The **Hamming distance** between two codewords c_i and c_j, denoted by $d_H(c_i,c_j)$, is the number of elements at which they differ

- **Examples:**

$$d_H(011,000) = 2$$
$$d_H(011,111) = 1$$

Error Correction and Detection

- Consider a code consisting of two codewords with Hamming distance d. How many errors can be detected? Corrected?



- # of errors that can be detected = $\lambda = d$-1

- # of errors that can be corrected $= t = \left\lfloor \frac{d-1}{2} \right\rfloor$

- In other words, for t-error correction: $d = 2t + 1$

Minimum Distance of a Code

- **Def.:** The **minimum distance** of a code C is the minimum Hamming distance between any two different codewords.

$$d_{\min} = \min_{i \neq j} d(c_i, c_j), \quad \forall\, c_i, c_j \in C$$

- A code with minimum distance d_{min} can correct all error patterns up to and including t-error patterns, where

$$d_{min} = 2t + 1$$

- It may be able to correct some higher weight error patterns, but not all.

Linear Block Codes

- A binary information vector (***X***) of k bits is mapped onto a binary vector ***C*** with $n>k$ bits
- The transformation is defined by a generator matrix ***G*** which is $k \times n$ matrix
- The message is segmented into blocks of k bits
- There are 2^k codewords (one for each 2^k possible information vectors)
- A binary block code is **linear** if and only if the modulo-2 sum of two codewords is also a codeword

Generator Matrix

- Message vector: $\boldsymbol{X} = [x_{m1} \;\; x_{m2} \;\; x_{m3} \;\; \ldots.. \;\; x_{mk}]$
- Codeword vector: $\boldsymbol{C} = [c_{m1} \;\; c_{m2} \;\; c_{m3} \;\; \ldots.. \;\; c_{mn}]$
- Generator matrix of the code:

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \vdots \\ \mathbf{g}_k \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{bmatrix}$$

- Encoding is performed by using:

$$\boldsymbol{C}_m = \boldsymbol{X}_m \, \boldsymbol{G}$$

(6,3) Linear Block Codes Example

Messages	Codewords
000	000000
100	110100
010	011010
110	101110
001	101001
101	011101
011	110011
111	000111

Systematic Property

- A codeword is divided into two parts:
  - Message (systematic) bits (k)
  - Parity-check bits (n-k)
- A linear block code with this structure is referred to as a linear ***systematic* block code**
- The generator matrix for such a code is given by:

$$G = \left[I_k \middle| P \right] = \begin{bmatrix} 1 & 0 & \dots & 0 & p_{11} & p_{12} & p_{13} & \cdots & p_{1n-k} \\ 0 & 1 & \dots & 0 & p_{21} & p_{22} & p_{23} & \cdots & p_{2n-k} \\ : & : & : & : & : & : & : & \dots & : \\ 0 & 0 & \dots & 1 & p_{k1} & p_{k2} & p_{k3} & \cdots & p_{kn-k} \end{bmatrix}$$

Encoding Example

- The (7,4) linear code has the following matrix as generator matrix

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \mathbf{g}_3 \\ \mathbf{g}_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

- If the message $\boldsymbol{x}$ = (1101), its corresponding codeword $\boldsymbol{C}$ is given by:

$$\boldsymbol{C} = \boldsymbol{X}\,.\boldsymbol{G}$$
$$= 1 \times g_1 + 1 \times g_2 + 0 \times g_3 + 1 \times g_4$$
$$= (1101000) + (0110100) + (1010001) = (0001101)$$

Encoding Example

- The following matrix $\boldsymbol{G}$ is in the systematic form:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

- The codeword $\boldsymbol{C}$ for a message $\boldsymbol{X}$ is given by:

$$\boldsymbol{C} = \boldsymbol{X} \cdot \boldsymbol{G}$$

$c_1 = x_1$

$c_2 = x_2$

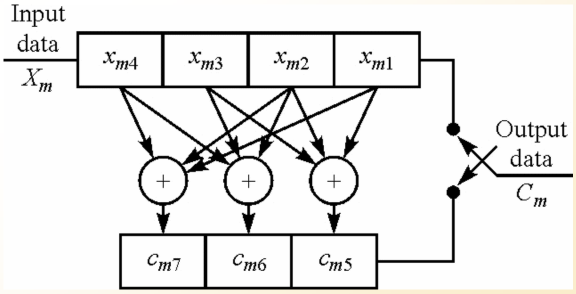
$c_3 = x_3$

$c_4 = x_4$

$c_5 = x_1 + x_2 + x_3$

$c_6 = x_2 + x_3 + x_4$

$c_7 = x_1 + x_2 + x_4$



Parity-Check Matrix

- An $(n\text{-}k) \times n$ **parity check matrix** H has its rows orthogonal to all codewords generated by G
- Thus, a vector $\boldsymbol{C}$ is a codeword in the code generated by $\boldsymbol{G}$ if and only if $\boldsymbol{C} \times \boldsymbol{H}^T = \mathbf{0}$
- The matrix H is called a **parity check matrix**
- For a generator matrix $G = [I_k\ P] \Rightarrow H = [P^T\ I_{n\text{-}k}]$
- For the last example:

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Hamming Weight & Distance

- The weight of the codeword is the number of non-zero element it contains
- **Hamming distance** d_{ij} is a measure of the difference between $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$ in any (n,k) block code
- The smallest value of the hamming weight is called the "**minimum distance**" $d_{\min}$
- The error detection capability of the code is:

$$d_{\min} - 1$$

- The error correction capability of the code is:

$$\left\lfloor \frac{1}{2}(d_{\min} - 1) \right\rfloor$$

Example: (7,4) Hamming Code

No.	Message	Codeword
0	0000	0000000
1	1000	1101000
2	0100	0110100
3	1100	1011100
4	0010	1110010
5	1010	0011010
6	0110	1000110
7	1110	0101110

No.	Message	Codeword
8	0001	1010001
9	1001	0111001
10	0101	1100101
11	1101	0001101
12	0011	0100011
13	1011	1001011
14	0111	0010111
15	1111	1111111

Linear Block Codes

- **Linear code:** The sum of any two codewords is a codeword.
- **Hamming codes** constitute a class of single-error correcting codes defined as:

$$n = 2^m - 1,\ k = n - m,\ m > 2$$

- The **minimum distance** of the code $d_{min} = 3$
- Hamming codes are perfect codes.

CYCLIC CODES

- An (n,k) linear code **C** is cyclic if every cyclic shift of a codeword in **C** is also a codeword in **C**
- If $c_0\ \ c_1\ \ c_2\ \ldots.\ c_{n-2}\ \ c_{n-1}$ is a codeword, then the following sequences:

$$\begin{array}{cccccc} c_{n-1} & c_0 & c_1 & \ldots. & c_{n-3} & c_{n-2} \\ c_{n-2} & c_{n-1} & c_0 & \ldots. & c_{n-4} & c_{n-3} \\ : & : & : & & : & : \\ c_1 & c_2 & c_3 & \ldots. & c_{n-1} & c_0 \end{array}$$

are all valid codewords.

Example

- The (7,4) Hamming code discussed before is cyclic:

1010001	1110010	0000000	1111111
1101000	0111001		
0110100	1011100		
0011010	0101110		
0001101	0010111		
1000110	1001011		
0100011	1100101		

Code Polynomial

- Let $\mathbf{c} = c_0\ c_1\ c_2\ \ldots.\ c_{n-1}$
- The code polynomial of **c** is:

$$c(X) = c_0 + c_1X + c_2X^2 + \ldots. + c_{n-1}X^{n-1}$$

where the power of X corresponds to the bit position, and the coefficients are **0**'s and **1**'s.

- **Example:**

1010001 $1+X^2+X^6$

0101110 $X+X^3+X^4+X^5$

Generator Polynomial

- **All code polynomials** are generated from one polynomial, *the* ***generator polynomial***, using

$$c(X) = a(X)g(X)$$

- The *generator polynomial* completely defines the code

- The (7,4) Hamming code can be generated from the generator polynomial $1+X+X^3$

BCH Codes

- **Definition of BCH codes:**

  For any positive integers m (m>2) and t_0 ($t_0 < n/2$), there is a BCH binary code of length $n = 2^m - 1$ which corrects all combinations of t_0 or fewer errors and has no more than mt_0 parity-check bits.

Codeword length	$2^m - 1$
Number of parity-check bits	$n - k \le mt_0$
Minimum distance	$d_{min} \ge 2t_0 + 1$

Table of Some BCH Codes

n	*k*	*d* (designed)	*d* (actual)	g(X)*
7	4	3	3	13
15	11	3	3	23
15	7	5	5	721
15	5	7	7	2463
31	26	3	3	45
31	16	5	7	107657
31	11	7	11	5423325

* Octal representation with highest order at the left.
721 is 111 010 001 representing $1+X^4+X^6+X^7+X^8$

Reed-Solomon (RS) Codes

- A class of non-binary BCH codes.
- The codeword consists of *n* *m*-bit symbols.
- The parameters of the code are related by: $n-k = 2t$
- **Example:** $m = 4$, $n = 15$, $k = 11$,
  - Codeword length is 15 symbols, or 15*4 = 60 bits.
  - It is a double-error correcting code ($t = 2$).
  - It can correct any burst of 8 or less bit errors.

Convolutional Codes

- Convolutional codes differ from block codes in:
  - The encoder has memory
  - The n-bit output codeword depends on the k-bit input massage and the previous input bits
- Convolutional coding is suitable for long messages such as streaming data (e.g., voice)
- The encoder consists of linear finite-state shift registers of K stages,
- Input bits are shifted k at a time to give n coded bits
- K is called the constraint length of the code.
- The rate of the code is $R = k/n$.

Convolutional Code Example 1

- This code has $K = 3$, $k = 1$, $n = 3$, => rate = 1/3
- The generator functions for the code are:

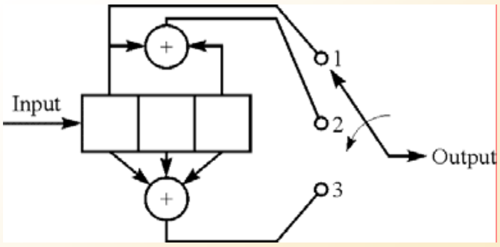
$$\boldsymbol{g}_1 = [100],\ \boldsymbol{g}_2 = [101],\ \boldsymbol{g}_3 = [111].$$

- The generator functions can be represented either in **octal form** or by **generator polynomials**:
  - Octal form: $\mathbf{(4, 5, 7)_8}$
  - Generator polynomials:

    $\boldsymbol{g}_1(x) = 1$

    $\boldsymbol{g}_2(x) = 1 + x^2$

    $\boldsymbol{g}_3(x) = 1 + x + x^2$



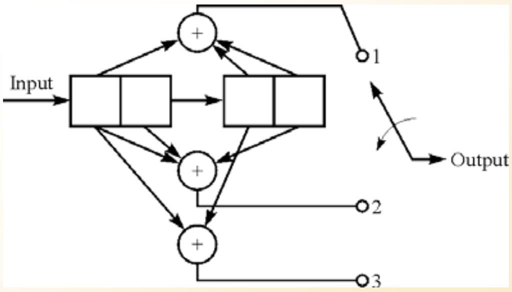
Convolutional Code Example 2

- This code has $K = 2$, $k = 2$, $n = 3$, and a rate of 2/3
- The generator functions for this code are $\boldsymbol{g}_1 = [1011]$, $\boldsymbol{g}_2 = [1101]$, $\boldsymbol{g}_3 = [1010]$ and in octal form, these generators are: $(13, 15, 12)_8$.
- Generator polynomials can be written as:

$$\boldsymbol{g}_1(x) = 1 + x^2 + x^3$$

$$\boldsymbol{g}_2(x) = 1 + x + x^3$$

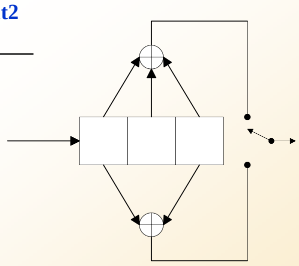
$$\boldsymbol{g}_3(x) = 1 + x^2$$



Encoding Example

- Find the state changes and the resulting output codeword sequence for the message m = 1 1 0 1 1 0 0. Assume that the initial contents of the encoder are all zero

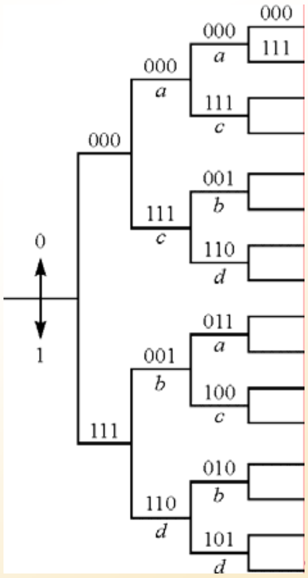
Input bit	Register Content	State at time t_i	State at time t_{i+1}	output1	Output2
-	0 0 0	0 0	0 0	-	-
1	1 0 0	0 0	1 0	1	1
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
1	1 0 1	0 1	1 0	0	0
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
0	0 0 1	0 1	0 0	1	1



- Output sequence: C = 1 1 0 1 0 1 0 0 0 1 0 1 1 1

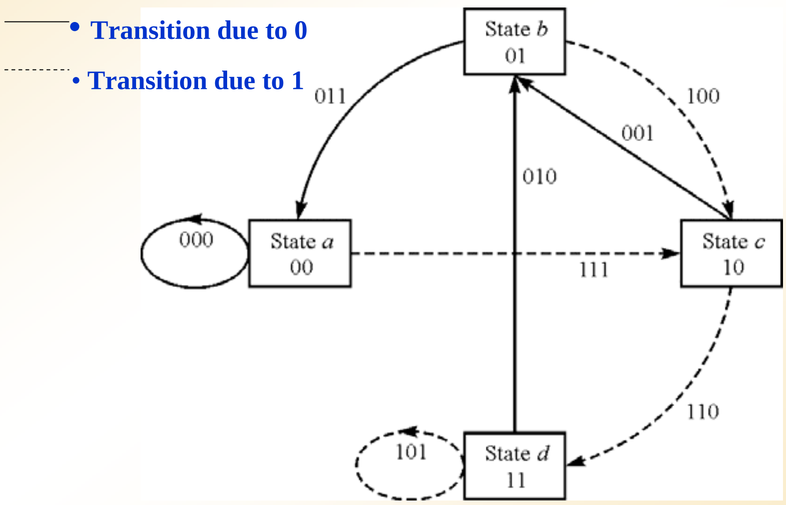
Convolutional Codes Representation

- Three methods to represent a convolutional code:
  - Tree diagram
  - Trellis diagram
  - State diagram
- A tree diagram for the rate-1/3, K =3 code is shown



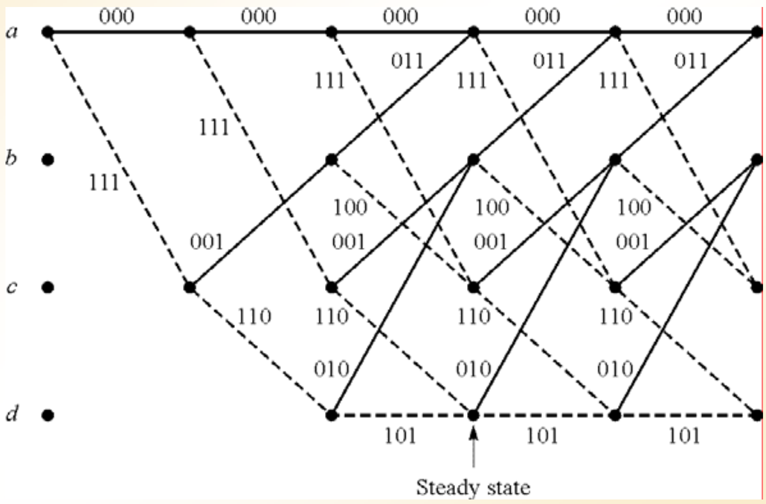
State Diagram

A state diagram for the rate-1/3, K =3 code is shown:



Trellis Diagram

A trellis diagram for the rate-1/3, *K* =3 code is shown:



Decoding of Convolutional Codes

- **Hard Decoding:**

  The received symbols at the output of the demodulator are quantized into two levels; zero and one, and fed to the decoder

- **Soft Decoding**:
  - The received symbols at the output of the demodulator are unquantized value (analog value) is used and fed to the decoder
  - Yields a gain of 2 - 2.2 dB compared to hard decoding

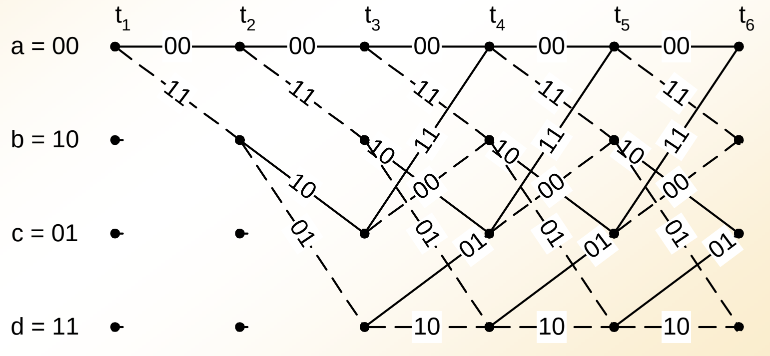
Decoding of Convolutional Codes

- **Maximum Likelihood Sequence Decoding:**
  - A convolutional code is converted into a block code of length *L* by feeding zeros at the end of the input message to force the encoder back to the zero state
  - Find the codeword with closest Hamming distance (hard decoding) or Euclidean distance (soft decoding) from the possible 2^L codewords.

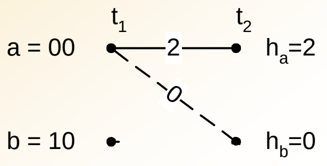
- **Viterbi Algorithm:**
  - Computes the distance between the received sequence and all the potential trellis paths
  - At each stage, keeps one "most likely" (surviving) path for each state

Viterbi Decoding Example

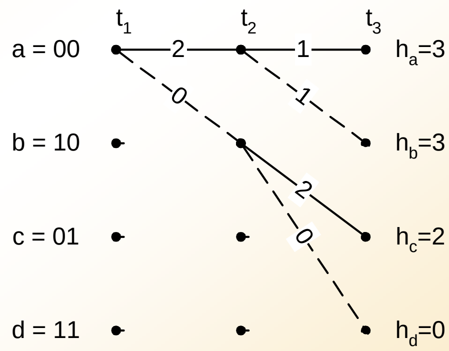
- For the convolutional encoder, assume that the received sequence *z* = 11 01 01 10 01.
- Find the input message sequence *m*.



Viterbi Decoding Example

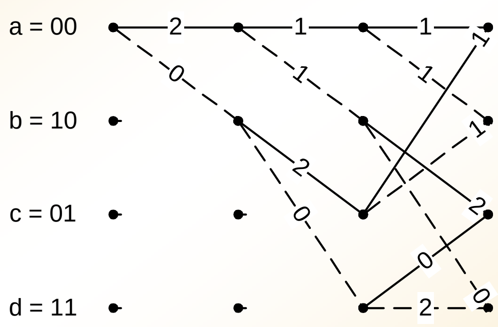


(a)



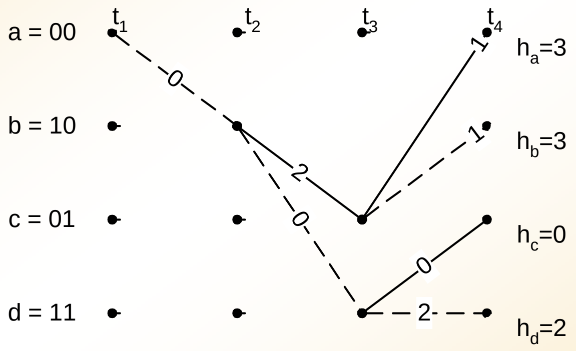
(b)

Viterbi Decoding Example



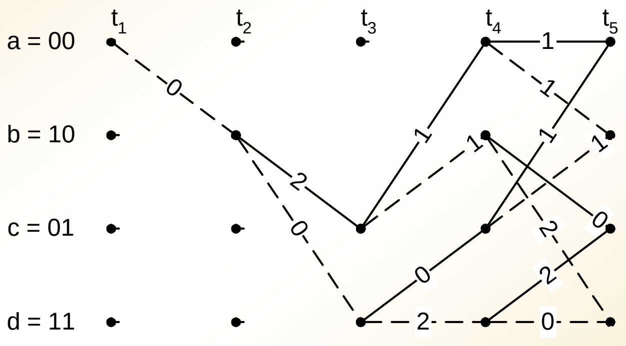
(c)

Viterbi Decoding Example



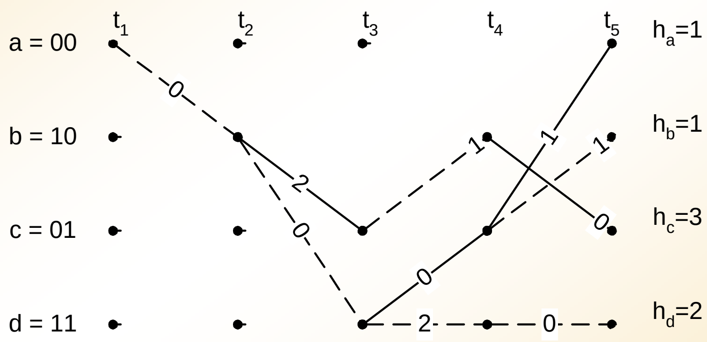
(d)

Viterbi Decoding Example



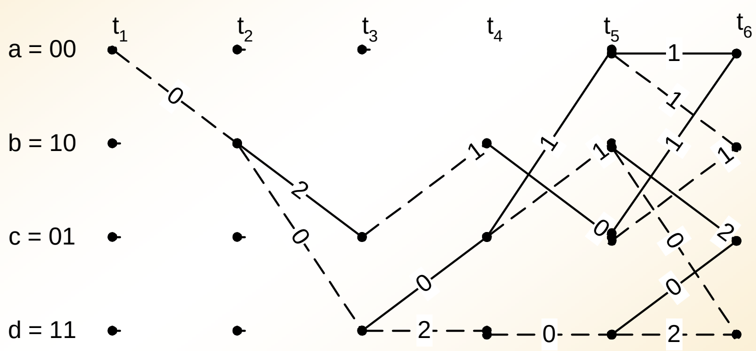
(e)

Viterbi Decoding Example



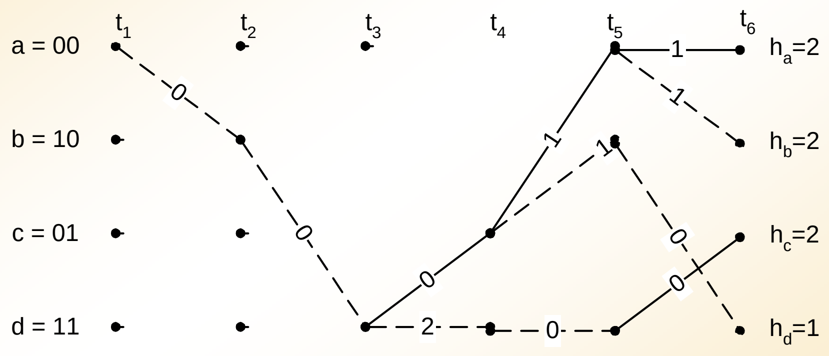
(f)

Viterbi Decoding Example



(g)

Viterbi Decoding Example

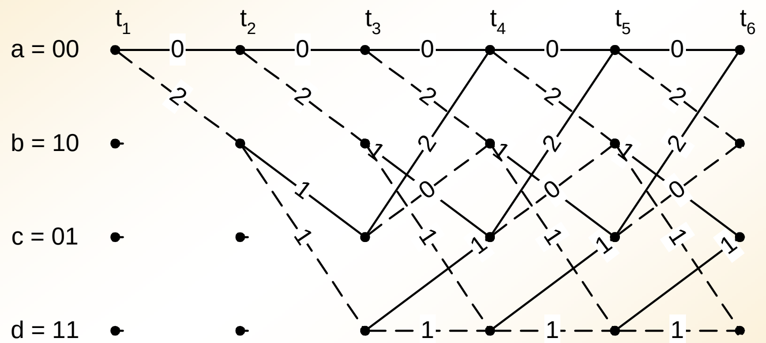


(h)

Distance Properties of Convolutional Codes

- Convolutional codes are linear codes
- The free distance of convolutional codes is associated with the path that starts and ends in the all zero state and does not return in between.
- So given the *all-zero* transmission an error occurs whenever the *all-zero* path does not survive.
- The minimum distance is found by exhaustively searching every path from the *all-zero* state to the *all-zero* state.

Distance Properties of Convolutional Codes



Trellis diagram labelled with distance from the *all-zero* path for the encoder

Distance Properties of Convolutional Codes

- Examining the pervious trellis diagram it is clear that the *free* distance of the code is 5

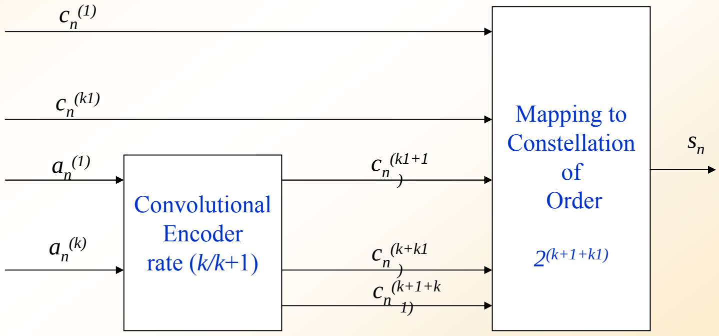
- This means that this code can correct up to:

$$\left\lfloor \frac{1}{2}(d_{\min} - 1) \right\rfloor = 2 \qquad \text{errors}$$

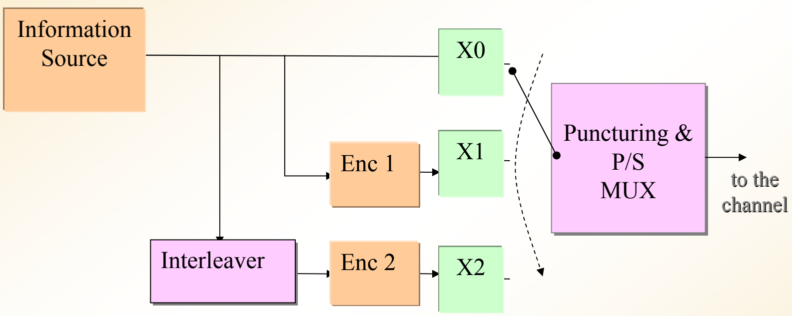
- A more closed form expression can be obtained by finding the transfer function of the code

Trellis Coded Modulation (TCM)

- ❑ Convolutional encoder with a signal output (instead of binary bits)
- ❑ Encoding is done to maximize some distance criterion in the signal constellation



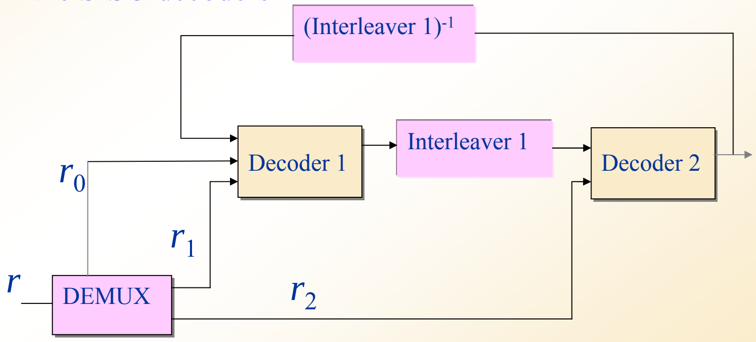
Turbo Codes



- ❑ Achieves performance close to channel capacity over AWGN and flat fading channels
- ❑ Information is encoded by two encoders after being interleaved

Turbo Decoder

- **Iterative decoding** is composed of:
  - Two soft-input soft-output decoders for codes 1 and 2
  - Interleaver – DeInterleaver pair
  - Soft information about massage bits are exchanged between the SISO decoders



Coding and Interleaving

- Codes designed for the AWGN channel do not work well in fading channels due to burst errors

- This can be compensated for by using standard AWGN channel combined with an interleaver to spread burst errors at the decoder

Coding and Interleaving

- Channel coding is a form of time diversity
- Independent fading is needed on each bit in a codeword to get the diversity gain
- Interleaving breaks the memory of the channel and provides independent fading for each bit
- The cost of interleaving is increased complexity and delay