Digital Communications and Simulation
ECE 5654

Section 4 – Error Correction Coding

Module 1 – Block Codes

Set 4 – Classes of Block Codes

Major Classes of Block Codes

- Repetition Codes
- Hamming Codes
- Golay Code
- BCH Codes
- Reed-Solomon Codes
- Walsh Codes
- Others
- BCH and RS codes are the most frequently used.

Classes of Linear Block Codes: (n,1) Repetition Codes

$$r = \frac{1}{n}, \qquad d_{H,\min} = n, \qquad t = \left\lfloor \frac{n-1}{2} \right\rfloor$$

$0 \Rightarrow 0000000000000$

$1 \Rightarrow 1111111111111$

- These codes are relatively simple, very wasteful of bandwidth, and are not widely used.
- A Direct-Sequence Spread-Spectrum system may be viewed as an application of a repetition code.

Classes of Linear Block Codes: Hamming Codes

$$n = 2^j - 1,$$

$$k = 2^j - 1 - j,$$

$$r = \frac{2^j - 1 - j}{2^j - 1},$$

$$d_{H,\min} = 3,$$

$$t = 1$$

- Example was presented in previous class.
- Not in widespread practical use.

Plot of BER vs. SNR for several Hamming codes

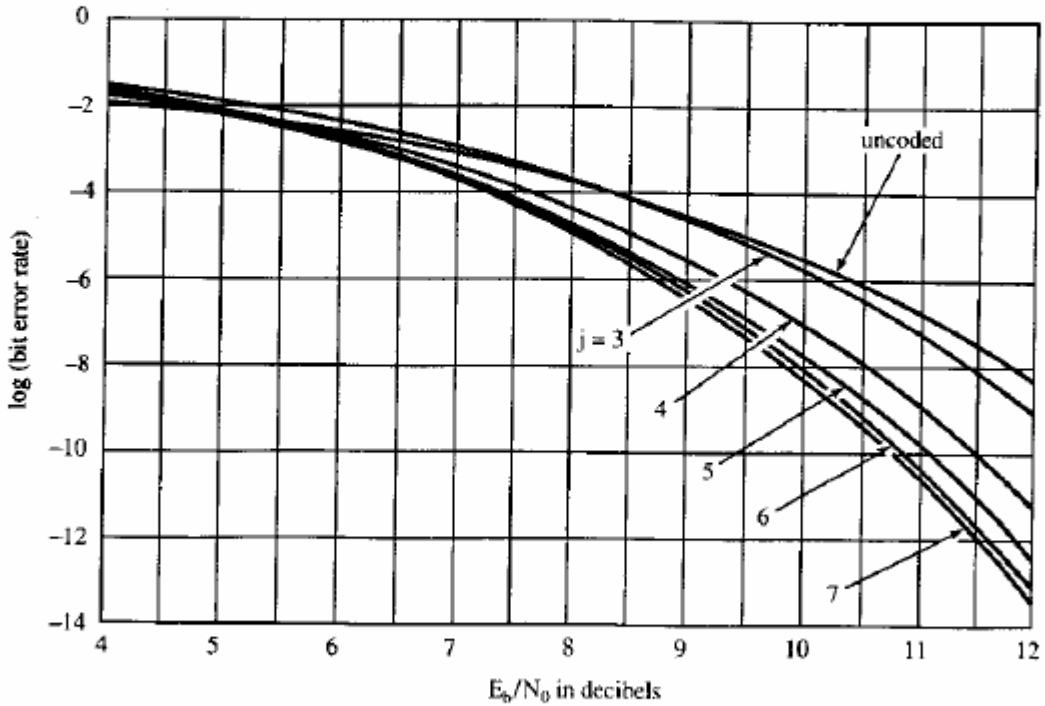

FIGURE 5-23. Bit error rate versus E_b/N_0 for Hamming codes with $j = 3$ through 7.

Notes on Hamming Code Performance

- Coding gain is achieved at high SNR
- BER is worse than uncoded system for low SNR
- Hamming code is not particularly powerful
  - single error correction only

Classes of Linear Block Codes: Golay Code

$$n = 23, \quad k = 12, \quad r = \frac{12}{23}, \quad d_{H,\min} = 7, \quad t = 3$$

- This is a special one-of-a-kind code with many interesting properties. The Golay code is the only non-trivial "perfect code":
  - 2^{12} = # of codewords
  - 2^{23} = # of possible binary vectors of length 23
  - Every possible received vector lies within distance 3 of exactly one codeword: $2^{12}\left[1+\binom{23}{1}+\binom{23}{2}+\binom{23}{3}\right]=2^{23}$
- n=23 is fairly short
  - this code is no longer used much in practice. One practical use: in Motorola pager system.

Plot of BER vs. SNR for Golay Code

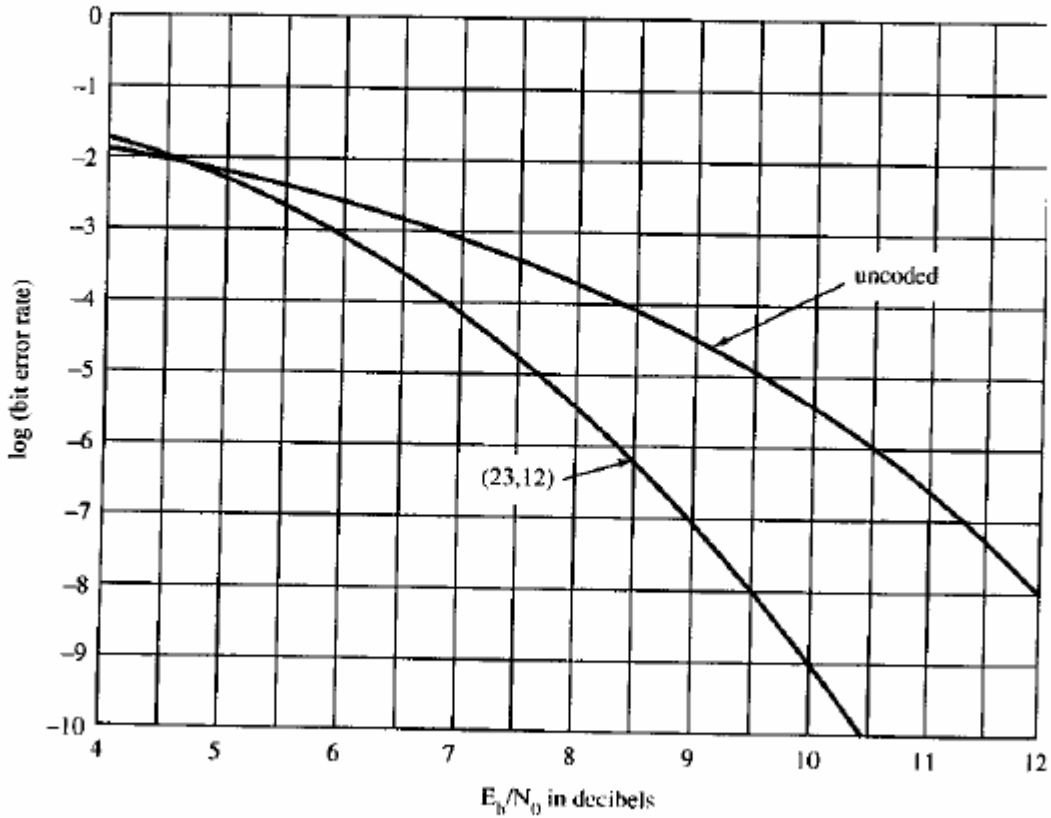

FIGURE 5-30. Bit error probability for Golay (23, 12) code.

Classes of Linear Block Codes: BCH Codes

- "Bose-Chaudhuri-Hocquenghem" - 1959
- Very important and useful class of codes.
- $$n = 2^j - 1, \quad k = \text{any value}, \quad t \geq \frac{2^j - 1 - k}{j} \text{ (guaranteed)}$$
- Widely used in satellite, wireless data links
- Decoded with the Berlekamp-Massey Algorithm

BER vs. SNR for r=3/4 BCH Codes

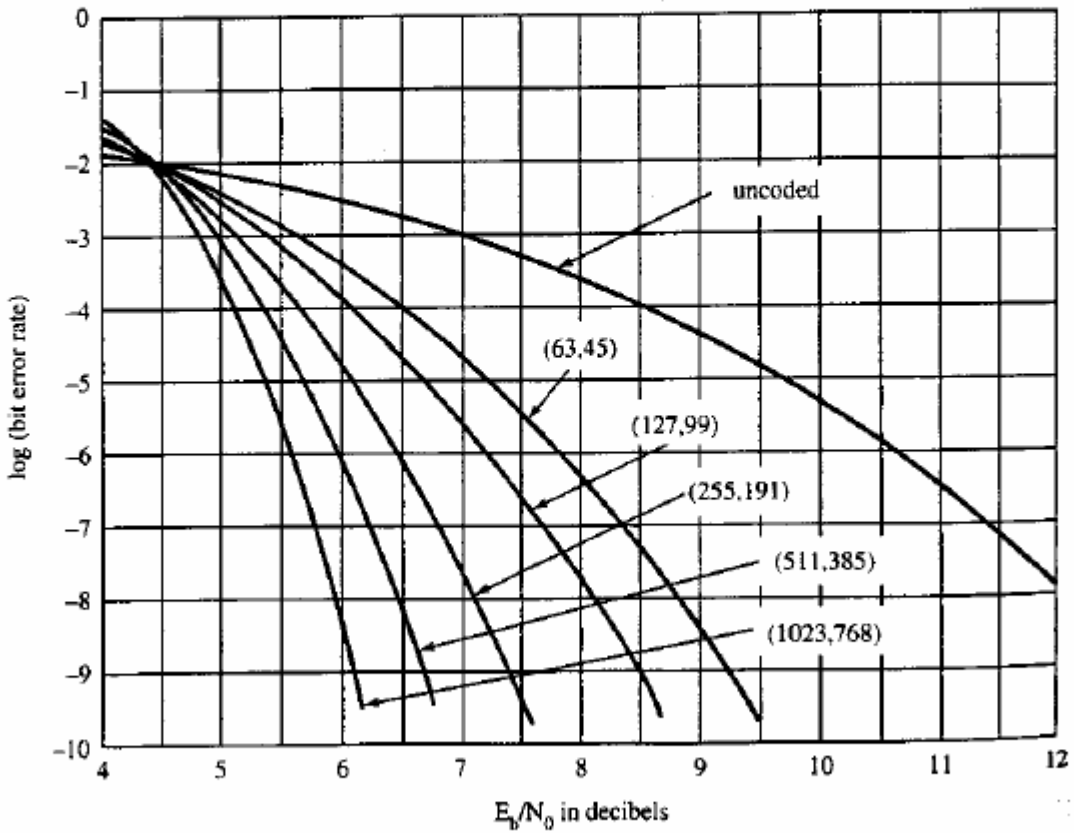

FIGURE 5-28. Bit error probability for BCH codes with $R \cong 3/4$.

BER vs. SNR for r=1/2 BCH Codes

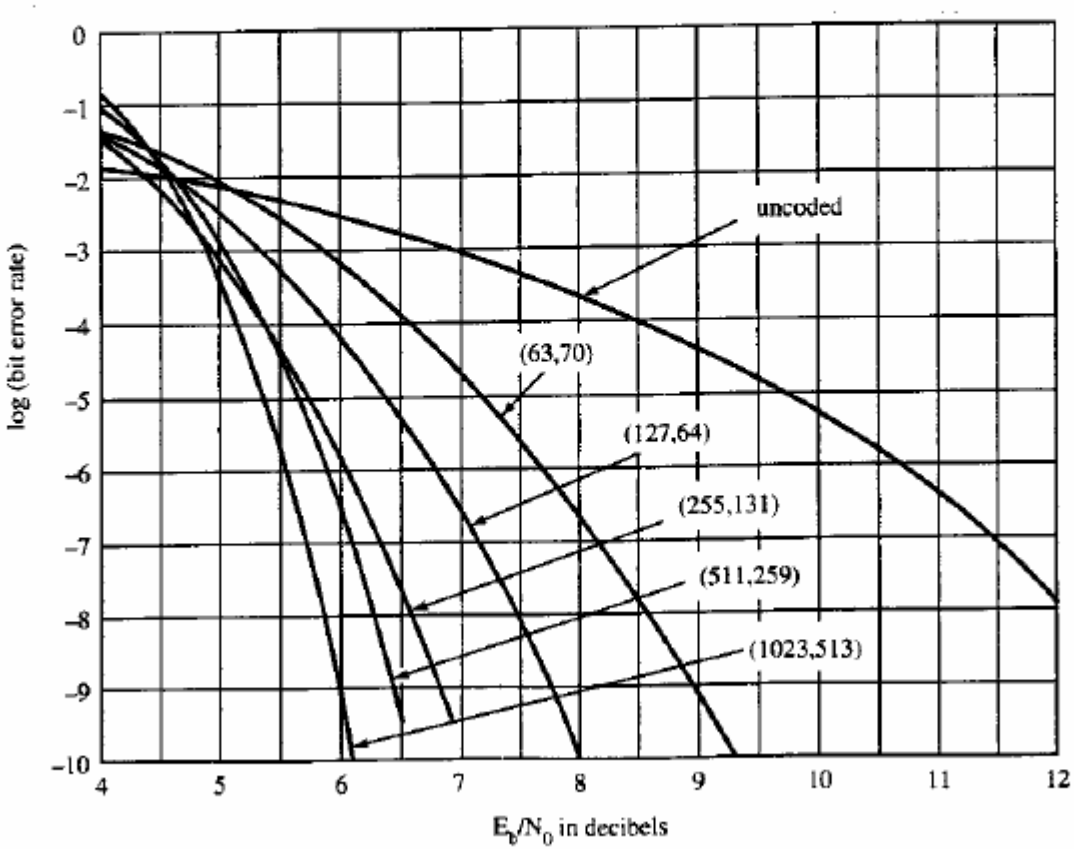

FIGURE 5-27. Bit error probability for BCH codes with $R \cong 1/2$.

BER vs. SNR for r=1/4 BCH Codes

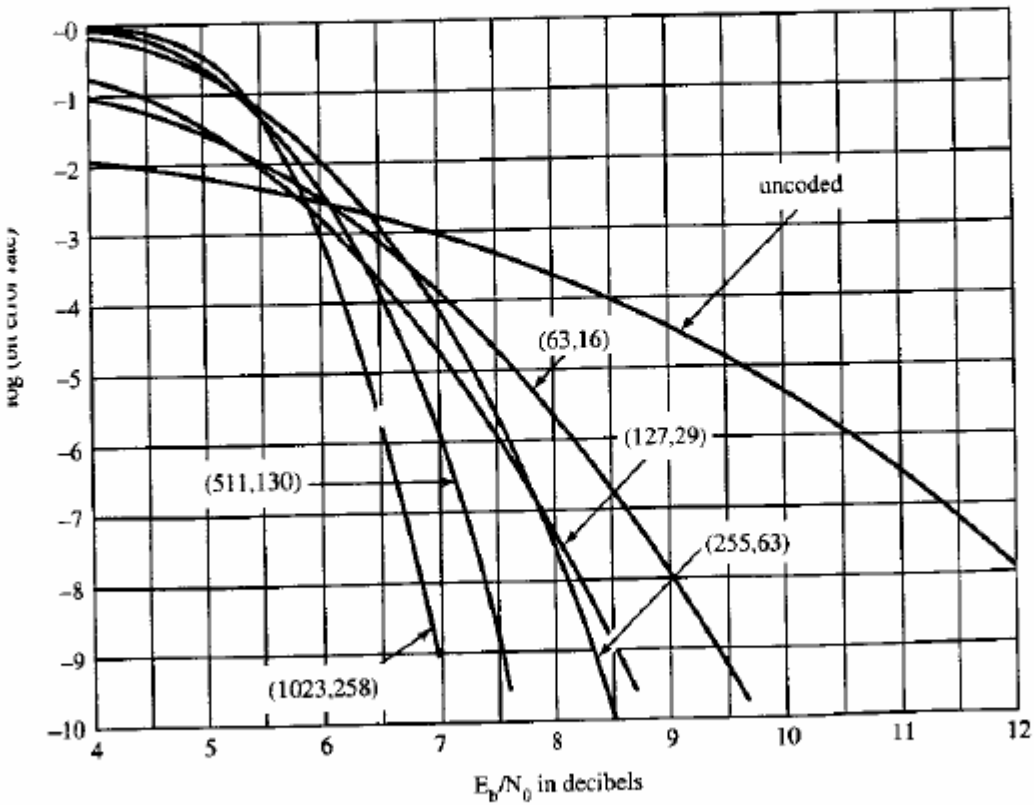

FIGURE 5-26. Bit error probability for BCH codes with $R \simeq 1/4$.

Notes on BCH Code Performance

- BCH Codes Exist for many values of n,k
- Large coding gains are possible for high SNR
- Coding gain increases with n
- Coding gain increases as rate r decreases (up to a point)

Classes of Linear Block Codes: Reed-Solomon (RS) Code

- 1962 - A generalization case of BCH codes
- $n = 2^j - 1, \;\; k = \text{any value}, \;\; d_{H,\min} = n - k + 1, \;\; t = \left\lfloor \frac{n-k}{2} \right\rfloor$
- RS codes are Maximum Distance Separable - have the largest possible distance for any code with the same value of n & k
- RS codes are constructed for nonbinary (M-ary) symbol sets - frequently used with M-ary FSK.

Applications of Reed-Solomon Codes

- RS codes are used for data communications in severely power-limited environments:
  - deep-space communications
  - military communications systems in conjunction with spread-spectrum
  - Compact Disks
  - Cellular Digital Packet Data Standard.

BER vs. SNR for RS Codes

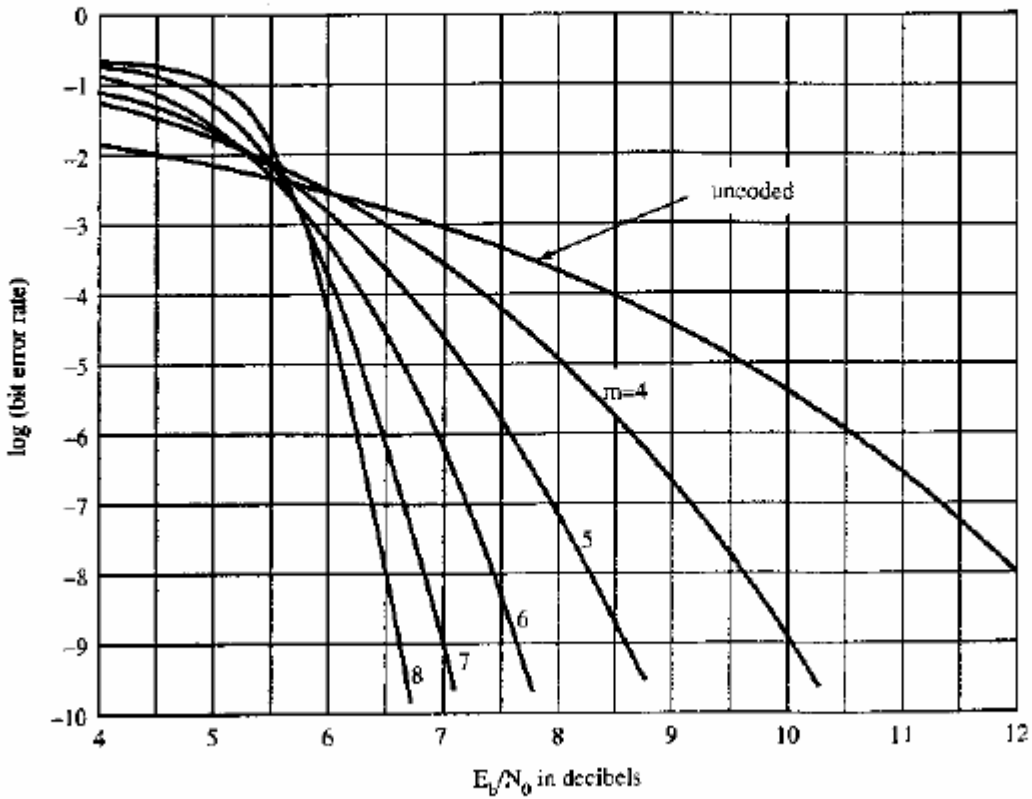

FIGURE 5-29. Bit error probability for Reed–Solomon codes with $R \cong 1/2$.

Orthogonal (Walsh) Codes

- Hadamard Matrices: $\mathbf{H}_1 = [1], \mathbf{H}_{2^{i+1}} = \begin{bmatrix} \mathbf{H}_{2^i} & \mathbf{H}_{2^i} \\ \mathbf{H}_{2^i} & \overline{\mathbf{H}}_{2^i} \end{bmatrix}$
- Examples: $H_2 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$
- The code parameters in general will be $\left(k, 2^k\right)$
- The minimum distance is given by $d_{\min} = n/2 = 2^{k-1}$
- Spectral efficiency becomes very poor but energy efficiency becomes good for large k

Other Well-Known Classes of Block Codes

- Reed-Muller Codes
  - discovered in mid-1950s
  - first large class of codes to correct more than a single error
  - used in *Mariner* deep space probes from 1969-1976
  - no longer attractive when compared to BCH and RS codes
- Fire Codes
  - useful in correcting long bursts of errors
  - sometimes used in magnetic data storage systems
  - largely replaced by RS codes in recent applications

Modifications to Known Codes

- Many known codes can be modified by an extra code symbol or deleting a symbol
  - can create codes that approximate almost any desired rate
  - can sometimes create codes with slightly improved performance
- The resulting code can usually be decoded with only slight modification to the decoding algorithm
- Sometimes modification process can be applied multiple times in succession

Modifications to Known Codes

- Puncturing: delete a parity symbol
  - (n,k) code -> (n-1,k) code
- Shortening: delete a message symbol
  - (n,k) code -> (n-1,k-1) code
- Expurgating: deleting some subset of codewords
  - (n,k) code -> (n,k-1) code

Modifications to Known Codes

- Extending: add an additional parity symbol
  - (n,k) code -> (n+1,k) code
- Lengthening: add an additional message symbol
  - (n,k) code -> (n+1,k+1) code
- Augmenting: add a subset of additional code words
  - (n,k) code -> (n,k+1) code

Digital Communications and Simulation
ECE 5654

Section 4 – Error Correction Coding

Module 1 – Block Codes

Set 5 – Performance of Block Codes

Probability of Codeword Error

- We wish to compute the probability $P_c(\varepsilon)$ that a bounded distance decoder will fail
- The decoder can correct up to, but not more than $t = \lfloor (d_{H,\min} - 1)/2 \rfloor$ errors
- We assume that the probability of an individual symbol error is p, and that symbol errors occur independently
- The symbol error probability p is determined from the modulation type

Probability of Codeword Error (continued)

- If we send n bits, the probability of receiving a <u>specific</u> pattern of i errors and n-i correct bits is:

$$p^i \cdot (1-p)^{n-i}$$

- There are $\binom{n}{i} = \frac{n!}{i! \cdot (n-i)!}$ distinct patterns of n bits with i errors and n-i correct bits, so the total probability of receiving a pattern with with i errors is:

$$\binom{n}{i} p^i \cdot (1-p)^{n-i}$$

Probability of Codeword Error (continued)

- Since we can correct any pattern of up to t errors, the overall probability of codeword error is:

$$P_c(\varepsilon) = 1 - \sum_{i=0}^{t} \binom{n}{i} p^i (1-p)^{n-i} = \sum_{i=t+1}^{n} \binom{n}{i} p^i (1-p)^{n-i}$$

Example:
Error Probability of (7,4) Hamming Code

- Assume we are using a (7,4) Hamming Code (t=1).
- Assume p=0.001
- There are fewer terms, so it is easiest to compute the summation:

$$P_c(\varepsilon) = 1 - \sum_{i=0}^{t} \binom{n}{i} p^i (1-p)^{n-i} = 1 - \sum_{i=0}^{1} \binom{7}{i} (0.001)^i (0.999)^{n-i}$$

$$= 1 - 1 \cdot (0.999)^7 - 7(0.001)^1 (0.999)^6$$

$$= 1 - 0.993 - 0.00696 = 2 \times 10^{-5}$$

Numerical Evaluation of $P_c(\varepsilon)$

- May need to use higher numerical precision if we are evaluating the form: $P_c(\varepsilon) = 1 - \sum_{i=0}^{t} \binom{n}{i} p^i (1-p)^{n-i}$
- In general, $\binom{n}{i}$ can be very large and $p^i(1-p)^{n-i}$ can be very small. You may need to evaluate them jointly in order to avoid overflow or underflow (Matlab is pretty good about avoiding this)
- Frequently the term $i=t+1$ and the first few terms thereafter are the most significant

Matlab Functions

- For evaluating $n!$: fact.m

```
function y=fact(n)
y=1;
for i=1:n y=y*i; end;
```

- For evaluating $\binom{n}{i}$: binom.m

```
function y = binom(n,i);
y=fact(n)/(fact(i)*fact(n-i));
```

Example:

- Find the error probability of a (63,45) BCH code with $t=3$ for $p=0.001$.
  - 1 term:
    - EDU» Pc=0; p=0.001;
    - EDU» for i=4:4 Pc=Pc+binom(63,i)*p^i *(1-p)^(63-i); end;
    - Pc = 5.6152e-007
  - 2 terms:
    - EDU» Pc=0; p=0.001;
    - EDU» for i=4:5 Pc=Pc+binom(63,i)*p^i *(1-p)^(63-i); end;
    - Pc = 5.6815e-007
  - 3 terms: Pc= 5.6822e-007

Copyright 2002 – Brian D. Woerner and William H. Tranter

An Important Point about E_b/N_o

- We frequently want to evaluate performance in terms of E_b/N_o
- When using coding, we send extra bits which contain no information at all. In order to make a fair comparison with uncoded systems, we must penalize ourselves by the extra energy used to send those bits.
- We will need to replace E_b/N_o by $r\,E_b/N_o$ in all our error formulas for different modulation types

Example

- Suppose BPSK modulation is employed and we have $E_b/N_o = 10dB$. Find the probability of error both for an uncoded system and for a system with a (63,45) BCH code:
- Uncoded System: $P_b(e) = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = 3.8794\text{e-}006$
- Coded System: $p = Q\left(\sqrt{\frac{2rE_b}{N_0}}\right) = 7.8625\text{e-}005$

$$P_c(\varepsilon) \approx \sum_{i=4}^{8} \binom{63}{i} \left(7.86\times10^{-5}\right)^i \left(1-7.86\times10^{-5}\right)^{63-i}$$

$$= 2.2679\text{e-}011$$

Relating Codeword Error Rate and Bit Error Rate

- If the codeword is correctly received, all bits will be correctly received.
- Note that the probability of receiving a block of 45 uncoded bits with no errors is:

```
EDU» 1-(1-3.8794e-006)^45
ans =  1.7456e-004
```

- If a codeword is incorrectly decoded, a good approximation is that 1/2 of the bits will be in error.
- More exact analytical evaluation of bit error rate is tedious for block codes.