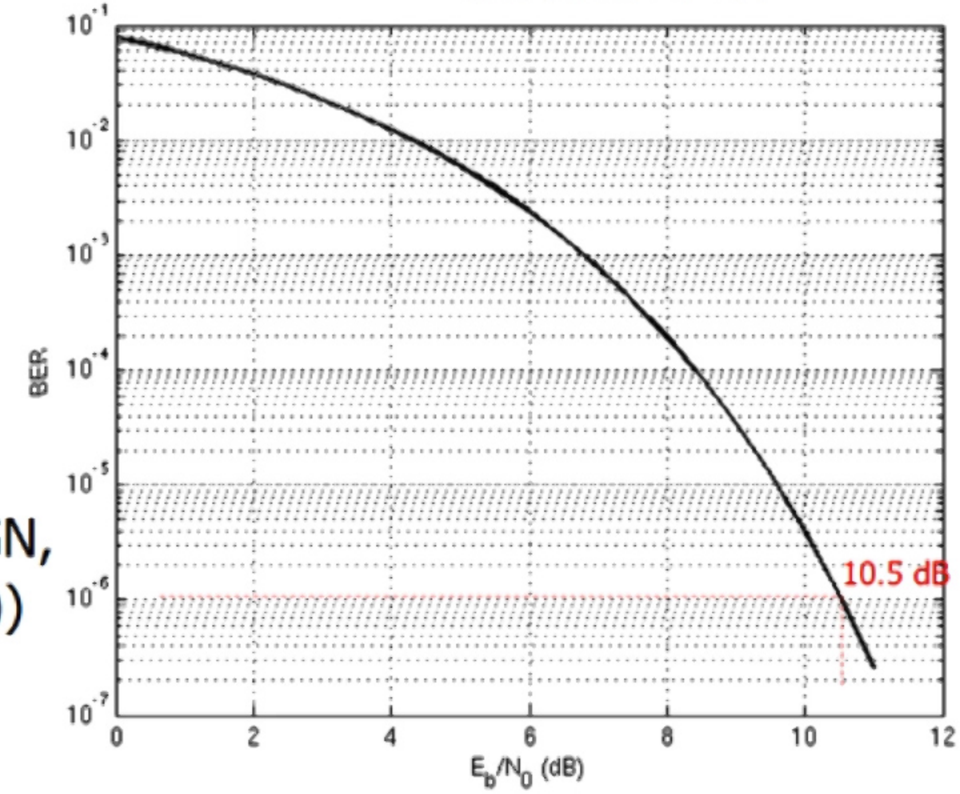
E_b/N_0 vs. BER

Signal energy per information bit:
$E_b = E_s/R$
Noise power:
$N_0/2 = \sigma^2$

$SNR = E_s/\sigma^2 = RE_b/(N_0/2)$
Or
$SNR = 2R\,(E_b/N_0)$

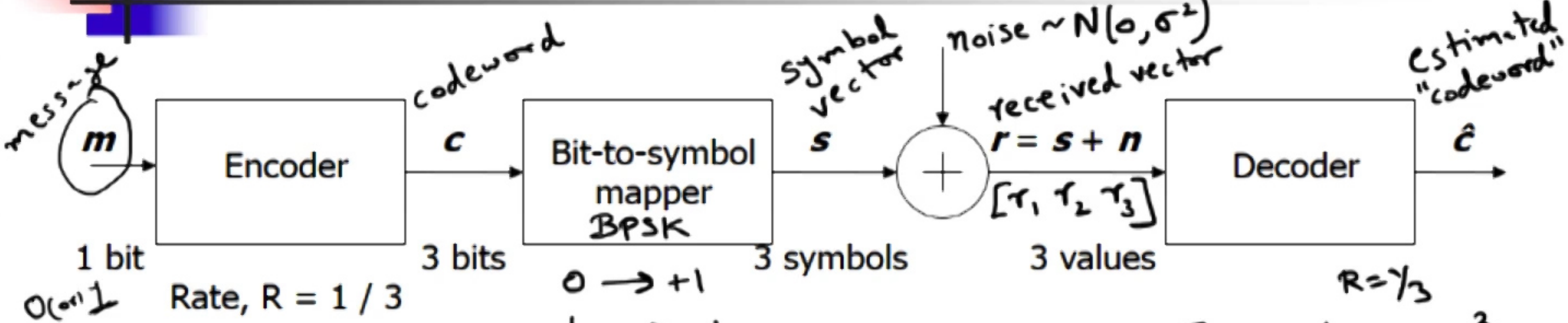
For uncoded BPSK over AWGN,
$BER = Q(1/\sigma) = Q(\mathrm{sqrt}(SNR))$
$= Q(\ \mathrm{sqrt}(2(E_b/N_0))\ )$

Plot of BER vs. E_b/N_0 (dB) for uncoded BPSK



Coding enables same BER at lower E_b/N_0's!

n=3 Repetition Code



Encoder:

If $\boldsymbol{m} = 0$, $\boldsymbol{c} = 000$, $\boldsymbol{s} = [+1\ +1\ +1]$.

If $\boldsymbol{m} = 1$, $\boldsymbol{c} = 111$, $\boldsymbol{s} = [-1\ -1\ -1]$.

$$\frac{E_b}{N_0} = \frac{1}{2R\sigma^2_{coded}} = \frac{3}{2\sigma^2_{coded}}$$

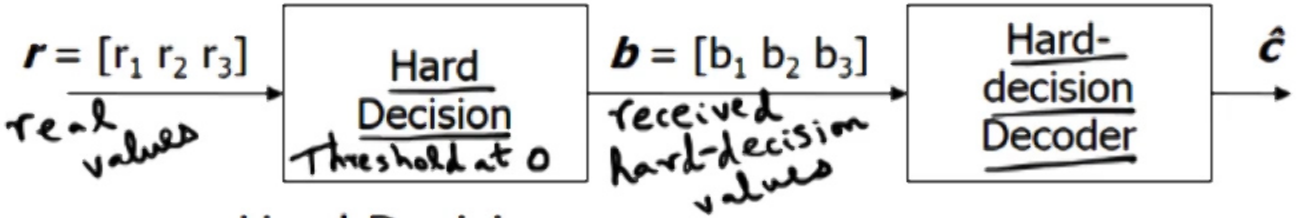
(n=3, repetition code)

$$\left(\frac{E_b}{N_0}\right)_{uncoded} = \frac{1}{2\sigma^2_{uncoded}}$$

Same $\frac{E_b}{N_0}$:

$$\sigma_{coded} > \sigma_{uncoded}$$

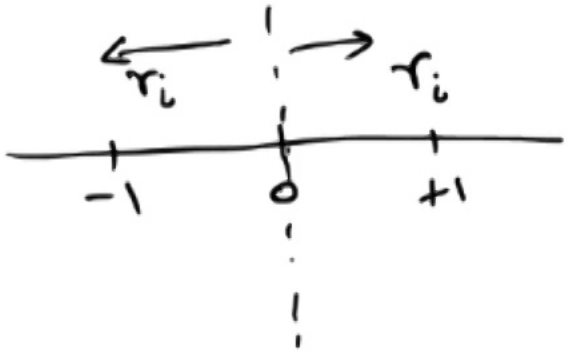
Decoder 1 for Repetition Code



Hard Decision:

If $r_i > 0$, $b_i = 0$.

If $r_i < 0$, $b_i = 1$.



$\boldsymbol{b}$	$\hat{c}$
000	000
001	000
010	000
100	000
011	111
101	111
110	111
111	111

closer to 000

closer to 111

BER vs. E_b/N_0 Analysis

$E_b/N_0 = (E_s/\sigma^2)/(2R)$

$= 1/(2R\sigma^2) = 3/(2\sigma^2)$

hard-decision value

Probability of error in b_i:

$p = Q(1/\sigma) = Q(\mathrm{sqrt}(2E_b/3N_0))$

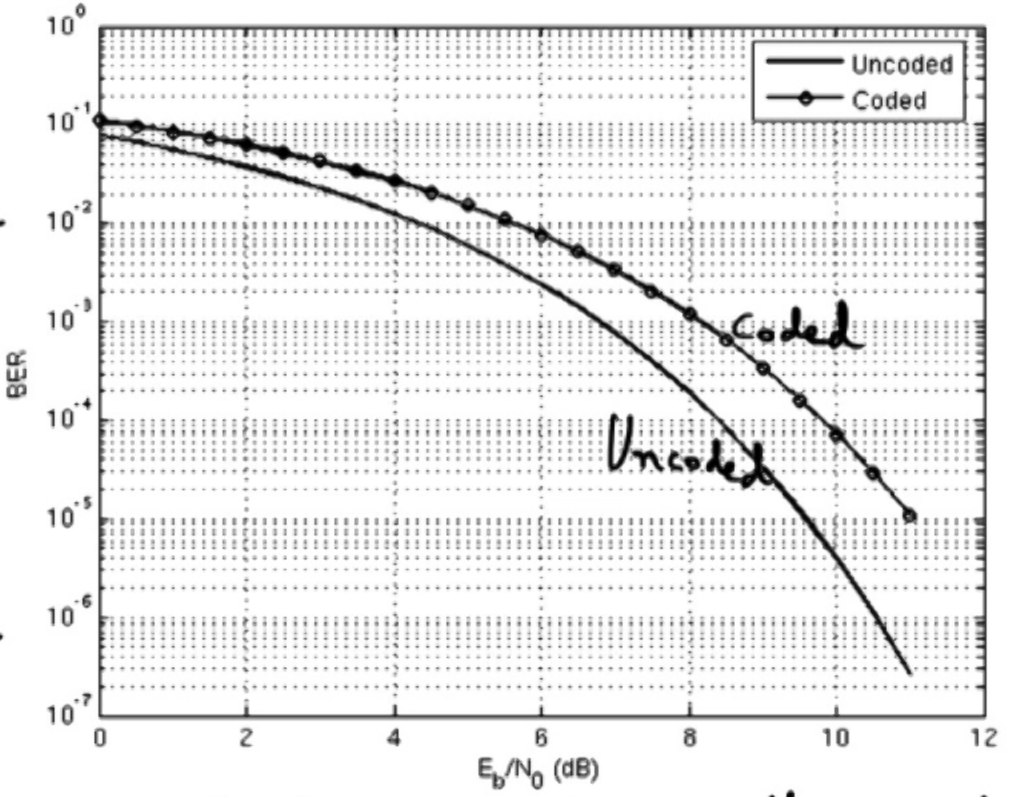
Probability of error in $\hat{c}$:

$P = 3p^2(1-p) + p^3$

2 b_i in error

3 b_i in error

two (or) more b_i should be in error



Surprise: coded worse than uncoded

Decoder 2 for Repetition Code

$r_1+r_2+r_3 > 0 \quad \hat{c} = 000$

$ < 0 \quad \hat{c} = 111$

$\boldsymbol{r} = [r_1\ r_2\ r_3] \longrightarrow$ Soft-decision Decoder $\longrightarrow \hat{c}$

If $|\underline{\boldsymbol{r}} - \overbrace{[+1\ +1\ +1]}^{\text{symbol vector for } [0\ 0\ 0]}|^2 < |\underline{\boldsymbol{r}} - \overbrace{[-1\ -1\ -1]}^{\text{symbol vector for } [1\ 1\ 1]}|^2$

$(r_1-1)^2+(r_2-1)^2+(r_3-1)^2 < (r_1+1)^2+(r_2+1)^2+(r_3+1)^2$

$r_1+r_2+r_3 > -r_1-r_2-r_3$

or $\boldsymbol{r}\,.\,[+1\ +1\ +1] > \boldsymbol{r}\,.\,[-1\ -1\ -1]$,

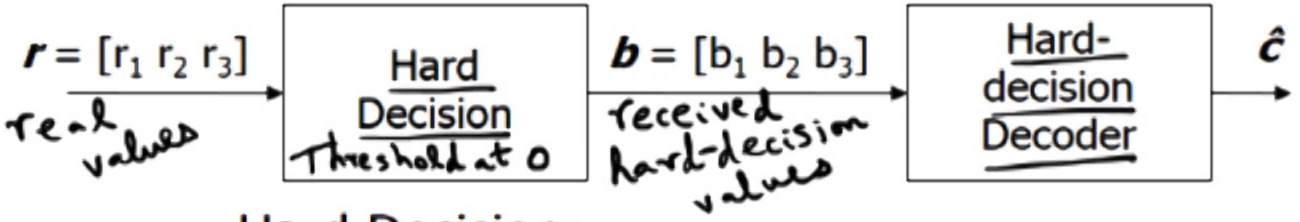
or $r_1+r_2+r_3 > 0$,

then $\underline{\hat{\boldsymbol{c}} = 000}$

else $\underline{\hat{\boldsymbol{c}} = 111}$

– Optimal decoder

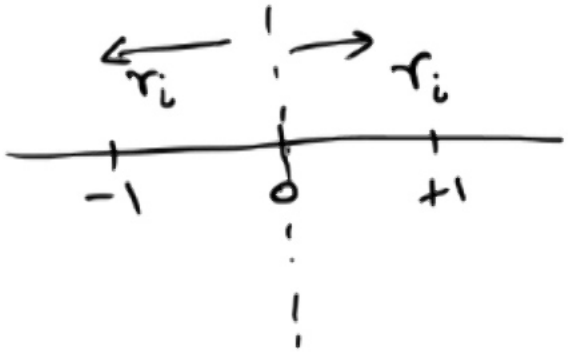
Decoder 1 for Repetition Code



Hard Decision:

If $r_i > 0$, $b_i = 0$.

If $r_i < 0$, $b_i = 1$.

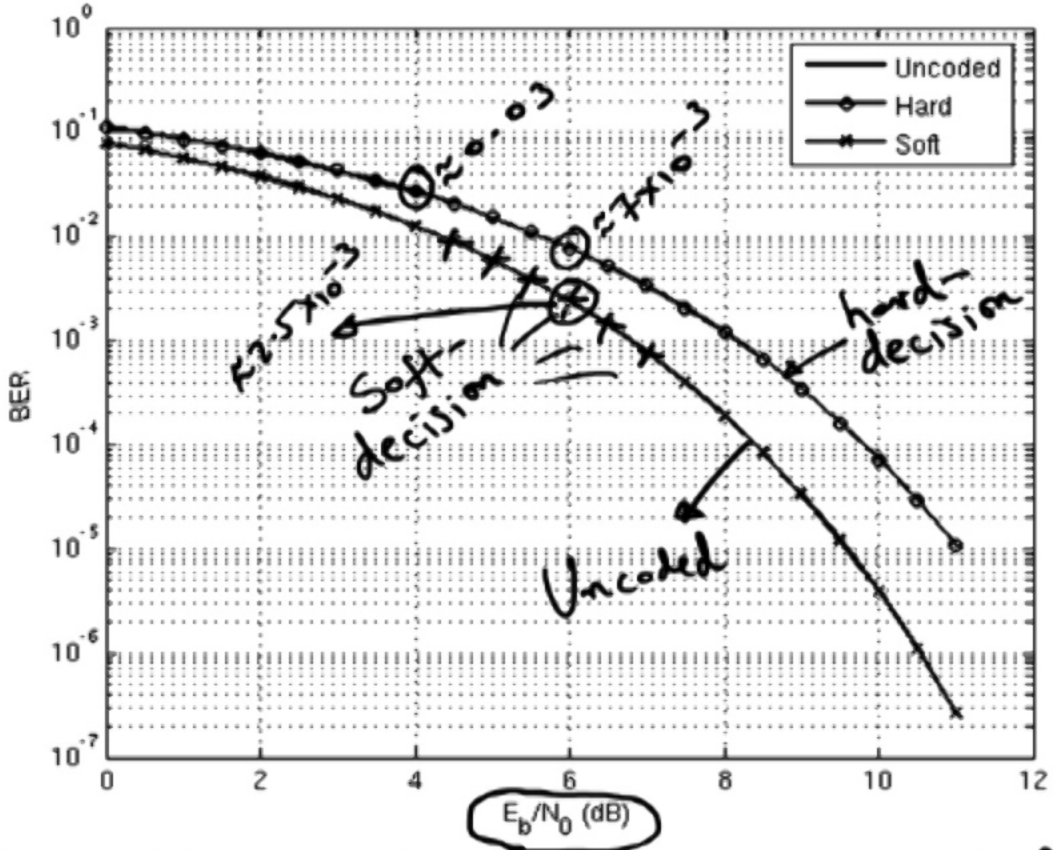


b	$\hat{c}$
000	000
001	000
010	000
100	000
011	111
101	111
110	111
111	111

closer to 000

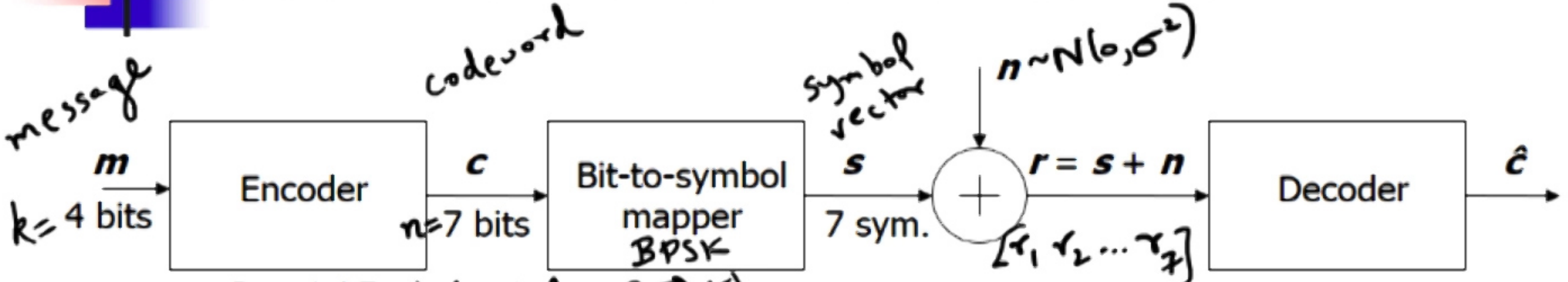
closer to 111

BER vs. E_b/N_0 Analysis



- Repetition code is not good enough for coding gain in E_b/N_0
- There are many others that are good!

Example: (7,4) Hamming Code



$R = 4/7$ bits/symbol

Message	Codeword
0000	0000000
0001	0001011
0010	0010110
0101	0101100
1011	1011000
0110	0110001
1100	1100010
1000	1000101

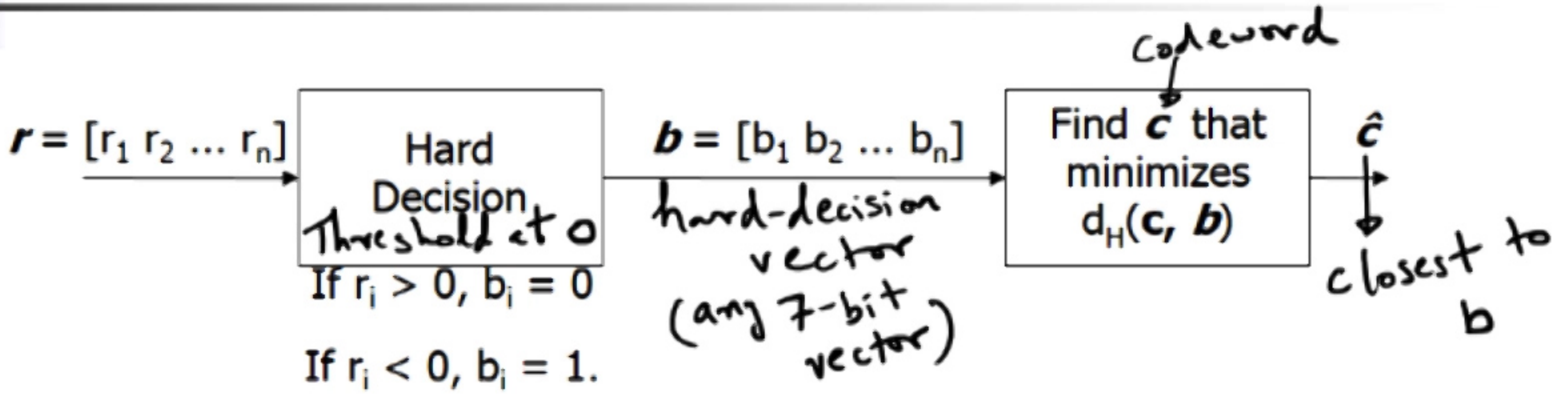
Message	Codeword
0100	0100111
1001	1001110
0011	0011101
0111	0111010
1110	1110100
1101	1101001
1010	1010011
1111	1111111

Remarks

- 16 codewords
- Cyclic
- Message appears in the codeword

first 4 bits are equal to msg.

Hard Decision Decoder

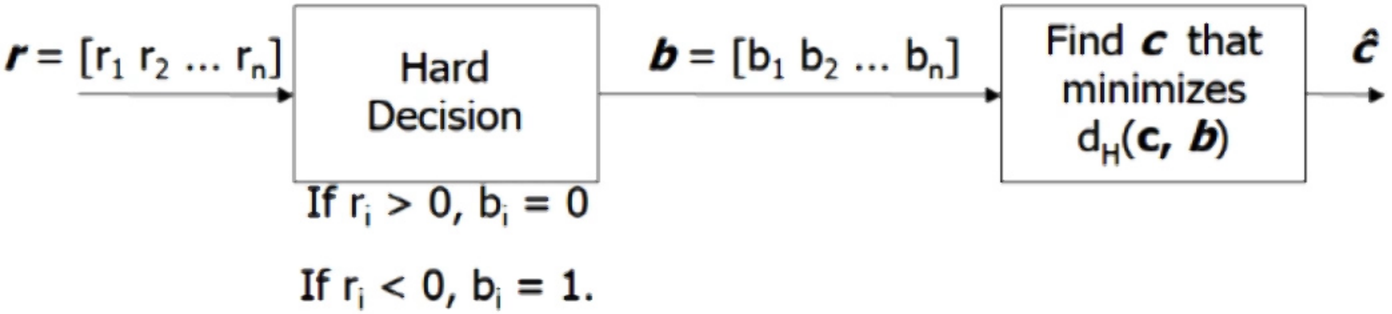


- Find codeword closest in Hamming distance
  - Find distance of $\boldsymbol{b}$ from 2^k codewords
  - Complexity increases exponentially with k

$$u = [u_1\ \ u_2\ \cdots\ u_n]$$
$$v = [v_1\ \ v_2\ \cdots\ v_n]$$
} binary

$d_H(u, v) =$ number of places in which u and v differ

Example: (7,4) Hamming Code



Message	Codeword	
0000	0000000	4
0001	0001011	5
0010	0010110	
0101	0101100	
1011	1011000	
0110	0110001	
1100	1100010	
1000	1000101	1

Message	Codeword	
0100	0100111	
1001	1001110	
0011	0011101	
0111	0111010	
1110	1110100	
1101	1101001	
1010	1010011	2
1111	1111111	

- $b = [1010101]$, $\hat{c} = ?$ 1000101
- $b = [0110110]$, $\hat{c} = ?$

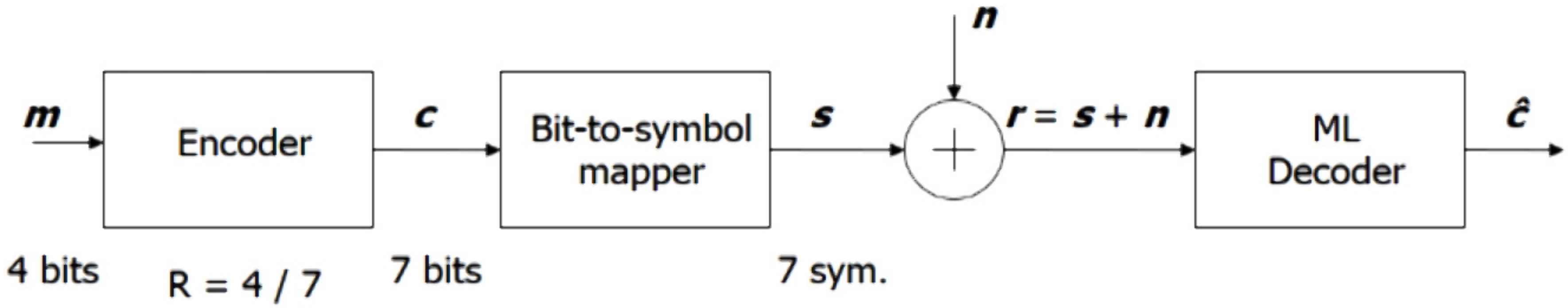
Maximum-Likelihood Decoder

Soft-decision decoder

$$\boldsymbol{r} = [r_1\ r_2\ \dots\ r_n] \longrightarrow \boxed{\text{Find } \boldsymbol{c} \text{ that minimizes } |\boldsymbol{r} - (1-2\boldsymbol{c})|^2} \longrightarrow \hat{\boldsymbol{c}}$$

- Find codeword closest in Euclidean distance
  - Find distances with 2^k code-symbol vectors
  - Complexity more than hard decision decoder
- Example: n=3 Repetition code; $\boldsymbol{r} = [1.9\ -0.6\ -0.2]$
  - ML decoder; $\hat{\boldsymbol{c}} = [1\ 1\ 1]$
    - $|\boldsymbol{r} - [1\ 1\ 1]|^2 = (0.9)^2 + (1.6)^2 + (1.2)^2 = 4.81$
    - $|\boldsymbol{r} - [-1\ -1\ -1]|^2 = (1.9)^2 + (0.4)^2 + (0.8)^2 = 4.41$
  - Hard decision decoder; $\hat{\boldsymbol{c}} = [0\ 0\ 0]$

Example: (7,4) Hamming Code



Maximum likelihood decoding: (soft-decision decoding)

1. Suppose $\boldsymbol{r}$ = [1.1 1.3 0.8 1.5 0.2 0.6 1.2]. $\hat{\boldsymbol{c}}$ = ?

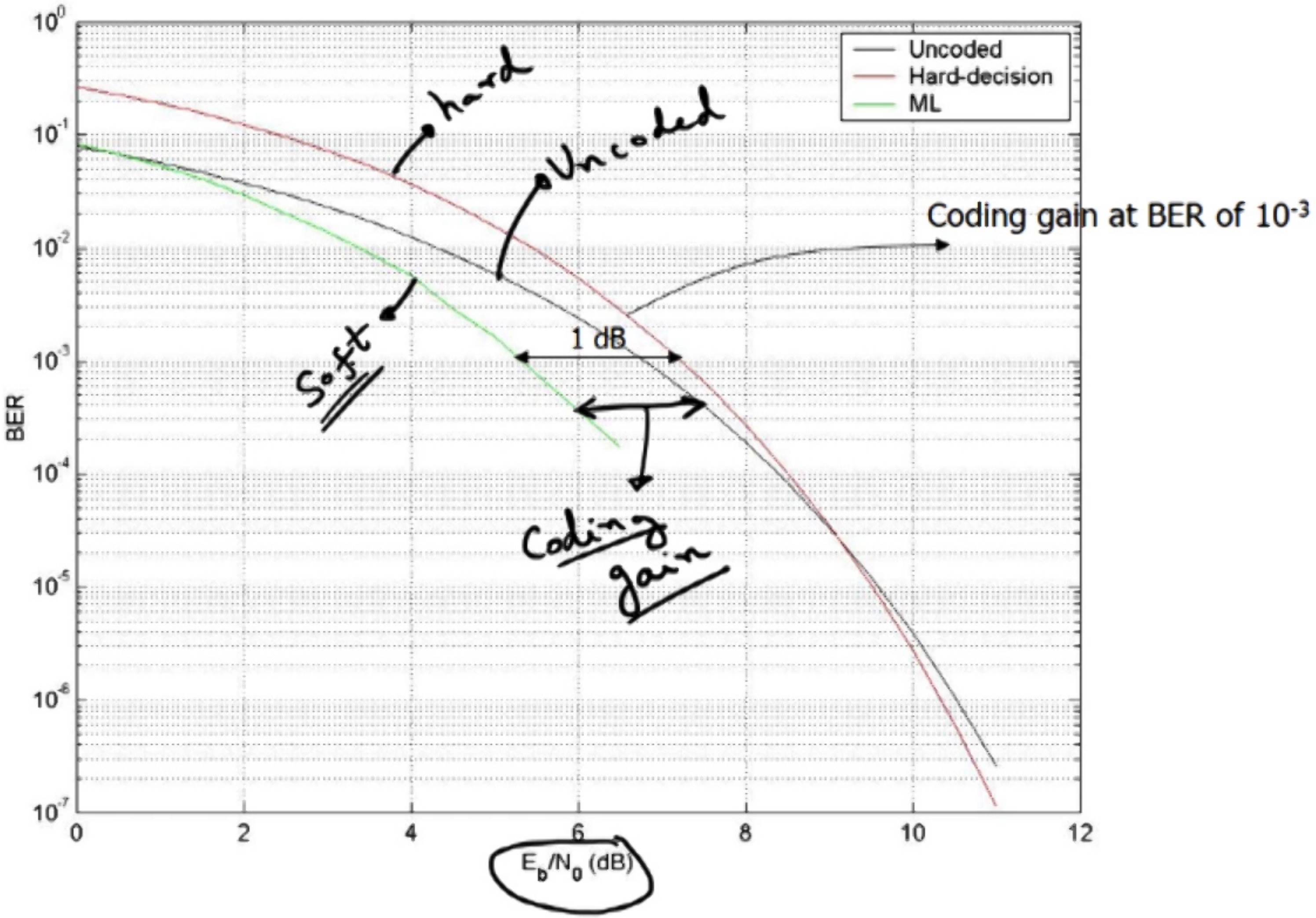
2. Suppose $\boldsymbol{r}$ = [1.1 1.3 0.8 1.5 -0.2 0.6 1.2]. $\hat{\boldsymbol{c}}$ = ?

3. Suppose $\boldsymbol{r}$ = [1.1 1.3 -0.8 1.5 -0.2 0.6 –0.1]. $\hat{\boldsymbol{c}}$ = ?

more complex

BER Curves for Hamming Code

$$\frac{E_b}{N_0} = \frac{7}{2\times 4\sigma^2} = \frac{7}{8\sigma^2}$$



Summary

- Error-correcting codes provide significant coding gains
  - Coding gain has to be calculated using the BER vs. E_b/N_0 plot
  - Longer codes provide better coding gains
  - Need to find good codes
  - Good decoders are important
- Efficient implementation of encoding, error detection and error correction are most important in practice