BER of BPSK in AWGN Channels

Student: Pei-Jin Su

Advisor: Kuei-Chiang Lai

2022/08/31 (Wed.)

Outline

- Simulation Scenarios
- Error Rate Curves: Theoretical vs. Simulated vs. Approximation
  - How do I know if my theoretical curve is correct?
  - Equal probability of the transmitted bits
  - Non-equal probability of the transmitted bits
- Scatter Plot and Histogram @ SNR = 3dB & SNR = 10dB
- My MATLAB Codes

Simulation Scenarios

- data bit $d = +1$ or -1 is transmitted.

$$P(d=+1) = P_1 \text{ and } P(d=-1) = 1 - P_1$$

- transmitted symbol

$$s = d \cdot \sqrt{2E_bT} = \begin{cases} \sqrt{2E_bT} & \text{, if } d = +1 \text{ is transmitted} \\ -\sqrt{2E_bT} & \text{, if } d = -1 \text{ is transmitted} \end{cases}$$

where T: bit interval (Assume $T = 1$ sec in my MATLAB code to simulate the BPSK system)

E_b: bit energy (Assume $E_b = 1$ J in my MATLAB code to simulate the BPSK system)

- additive Gaussian noise: $w \sim N(0, \sigma^2) \Rightarrow f_W(w) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{w^2}{2\sigma^2}}$

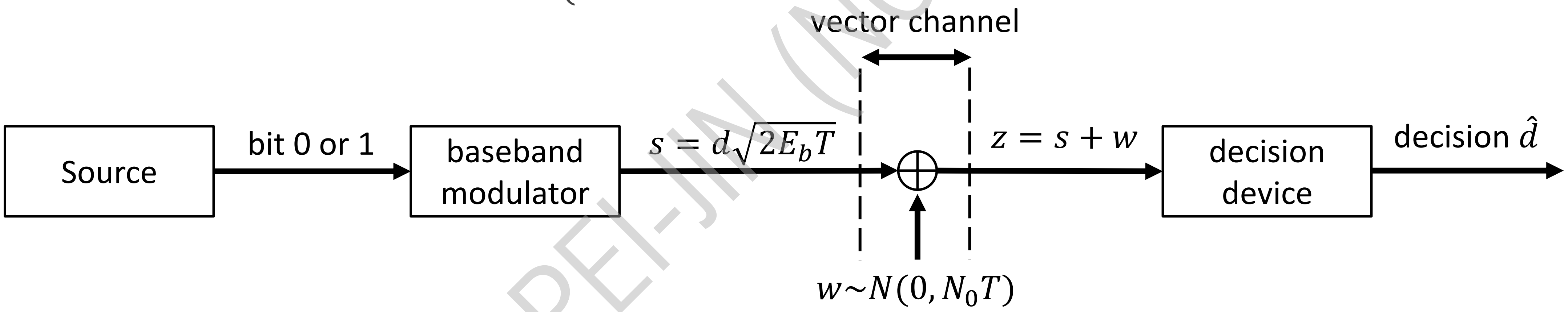
where $E[w] = 0$

$Var(w) = E[w^2] = \sigma^2 = N_0 T$

Simulation Scenarios

- discrete-time system model (Equivalent Baseband Model)

$$z = s + w = \begin{cases} N(\sqrt{2E_bT}, N_0T) & \text{, if } d = +1 \text{ is transmitted} \\ N(-\sqrt{2E_bT}, N_0T) & \text{, if } d = -1 \text{ is transmitted} \end{cases}$$



- definition of SNR

$$SNR_z \equiv \frac{E[|s|^2]}{E[|w|^2]} = \frac{2E_bT}{N_0T} = \frac{2E_b}{N_0}$$

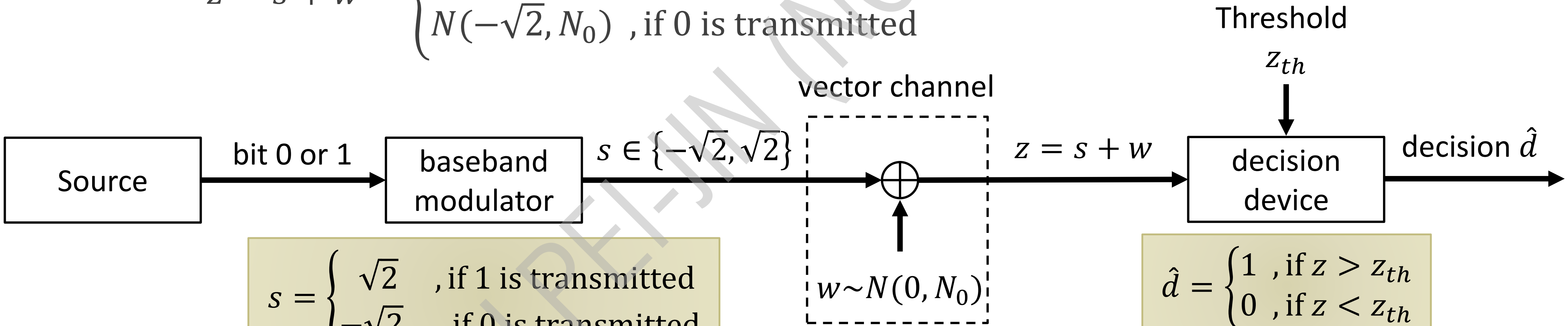
Simulation Scenarios

- system block diagram
  - Assume $T = 1$ sec and $E_b = 1$ J

$$z = s + w = \begin{cases} N(\sqrt{2}, N_0) & \text{, if 1 is transmitted} \\ N(-\sqrt{2}, N_0) & \text{, if 0 is transmitted} \end{cases}$$

ML detector: $z_{th} = 0$

MAP detector: $z_{th} = \frac{N_0 T}{2\sqrt{2E_b T}} \cdot \ln\left(\frac{1-P_1}{P_1}\right)$



$$s = \begin{cases} \sqrt{2} & \text{, if 1 is transmitted} \\ -\sqrt{2} & \text{, if 0 is transmitted} \end{cases}$$

$$\hat{d} = \begin{cases} 1 & \text{, if } z > z_{th} \\ 0 & \text{, if } z < z_{th} \end{cases}$$

$$N_0 = \frac{2}{SNR_z}$$

MAP criterion: $z_{th} = \frac{N_0}{2\sqrt{2}} \cdot \ln\left(\frac{1-P_1}{P_1}\right)$

- # bits simulated: 10^8

Error Rate Curves: Theoretical vs. Simulated vs. Approximation

- The theoretical bit error rate (BER) of the ML detector:

$$BER_{ML} = Q\left(\sqrt{SNR_z}\right) = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = Q\left(\sqrt{\frac{2}{N_0}}\right)$$

where $T = 1$ sec and $E_b = 1$ J

- Approximation of Q(x):

$$Q(x) \cong \frac{1}{2} e^{-\frac{x^2}{2}} \qquad \text{(Approximation 1)}$$

$$Q(x) \cong \frac{1}{\sqrt{2\pi x}} e^{-\frac{x^2}{2}} \qquad \text{(Approximation 2)}$$

Error Rate Curves: Theoretical vs. Simulated vs. Approximation

- The theoretical bit error rate (BER) of the MAP detector:

$$BER_{MAP} = P_1 \cdot Q\left(\sqrt{SNR_z} - \frac{z_{th}}{\sqrt{N_0 T}}\right) + (1 - P_1) \cdot Q\left(\sqrt{SNR_z} + \frac{z_{th}}{\sqrt{N_0 T}}\right)$$

$$= P_1 \cdot Q\left(\frac{\sqrt{2E_b T} - z_{th}}{\sqrt{N_0 T}}\right) + (1 - P_1) \cdot Q\left(\frac{\sqrt{2E_b T} + z_{th}}{\sqrt{N_0 T}}\right)$$

$$= P_1 \cdot Q\left(\frac{\sqrt{2} - z_{th}}{\sqrt{N_0}}\right) + (1 - P_1) \cdot Q\left(\frac{\sqrt{2} + z_{th}}{\sqrt{N_0}}\right)$$

where $T = 1$ sec and $E_b = 1$ J

- Why plotting theoretical curves?

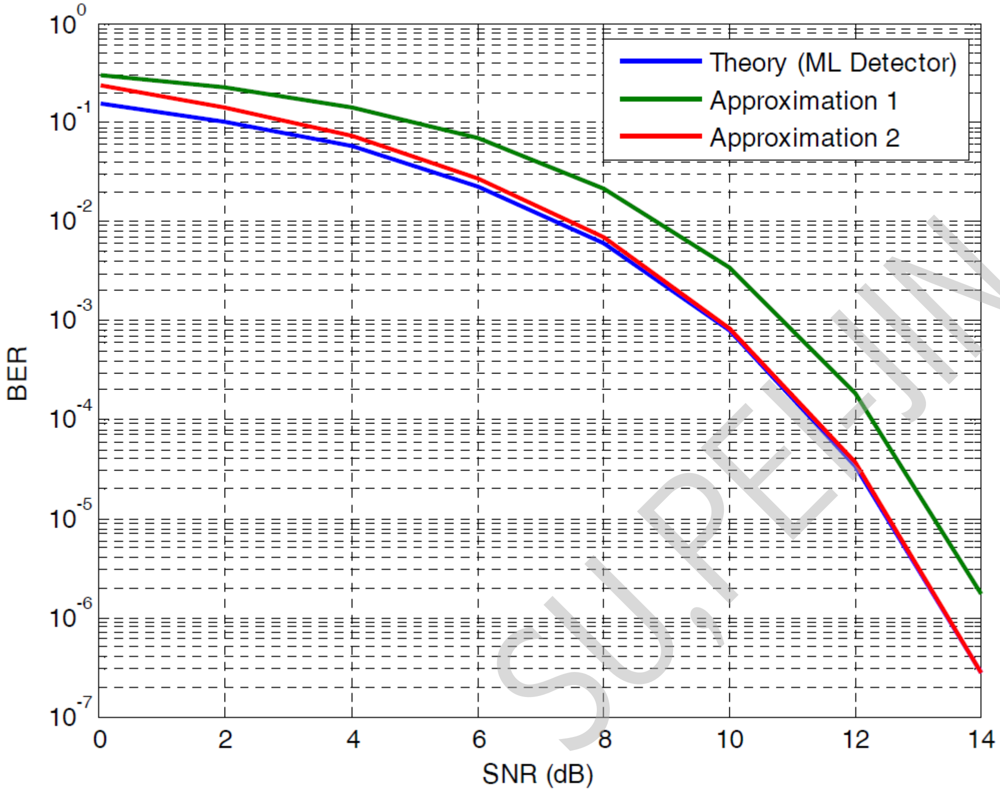
Ans: Compare my simulation result with theoretical BER curves.

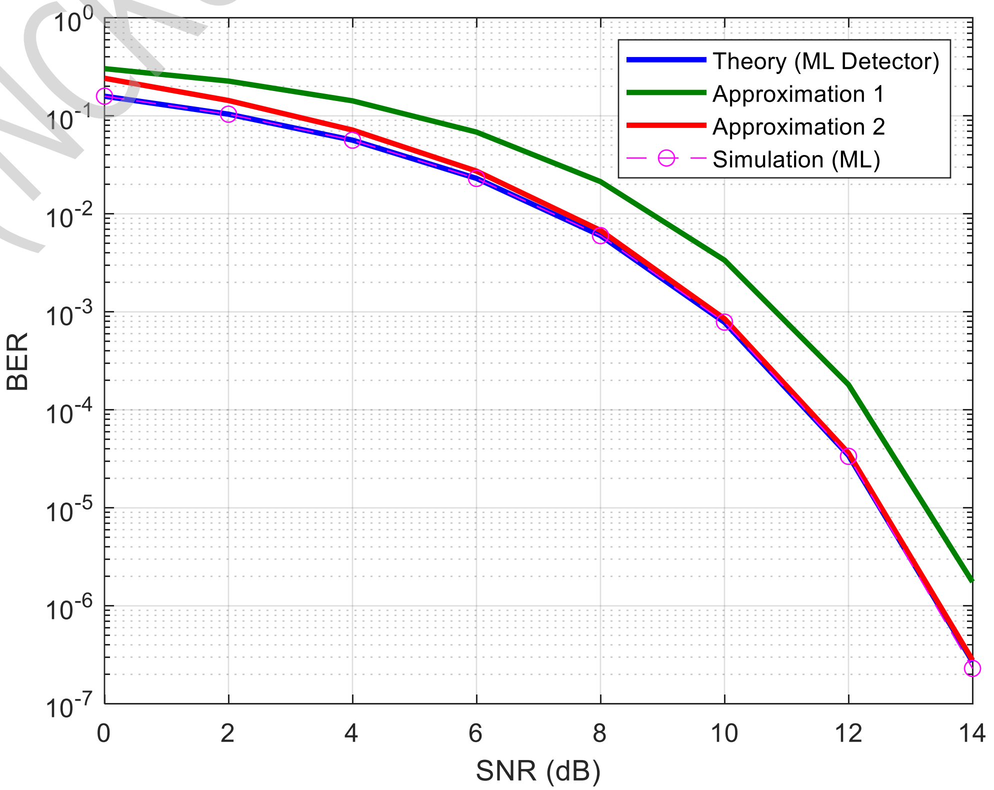
How do I know if my theoretical curve is correct?

- Appendix 1

Approximation 1: $Q(x) \cong \frac{1}{2} e^{-\frac{x^2}{2}}$

Approximation 2: $Q(x) \cong \frac{1}{\sqrt{2\pi}x} e^{-\frac{x^2}{2}}$





How do I know if my theoretical curve is correct?

- Appendix 2

Actual: $P_E = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = Q\left(\sqrt{2z}\right)$

Approximation (8.16): $P_E \cong \frac{1}{2\sqrt{\pi z}} e^{-z}$, $z \gg 1$

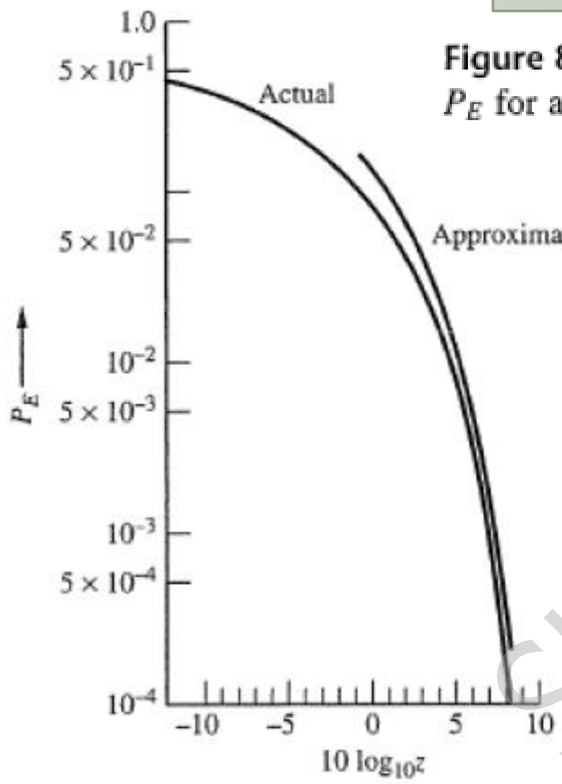
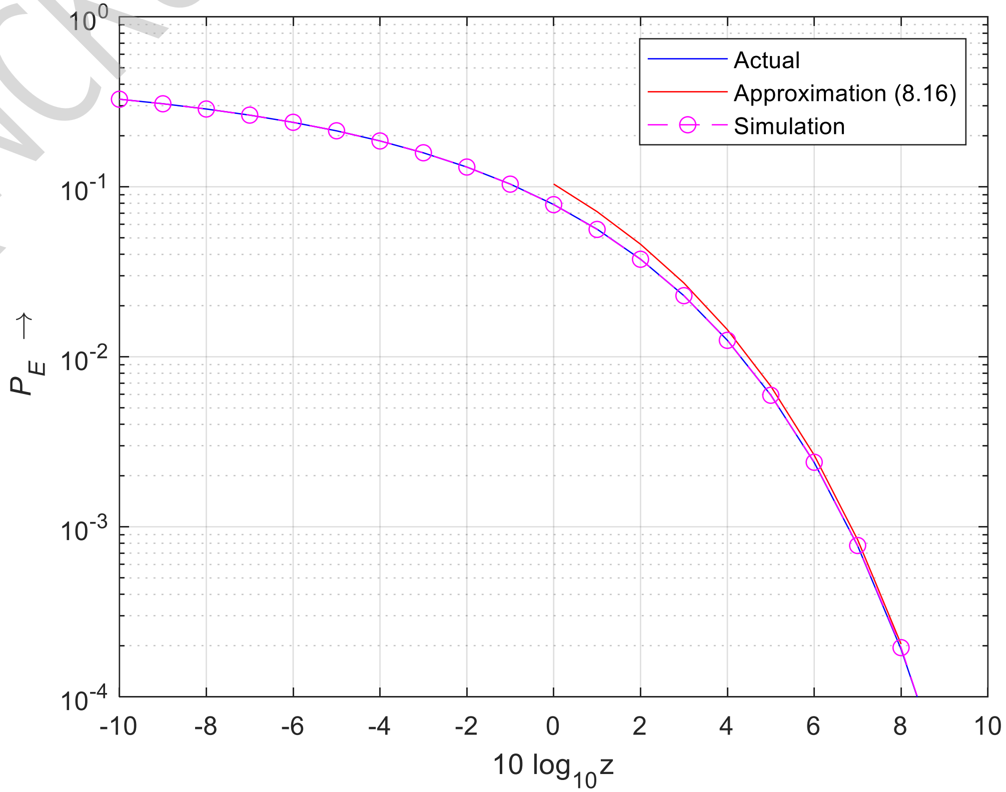


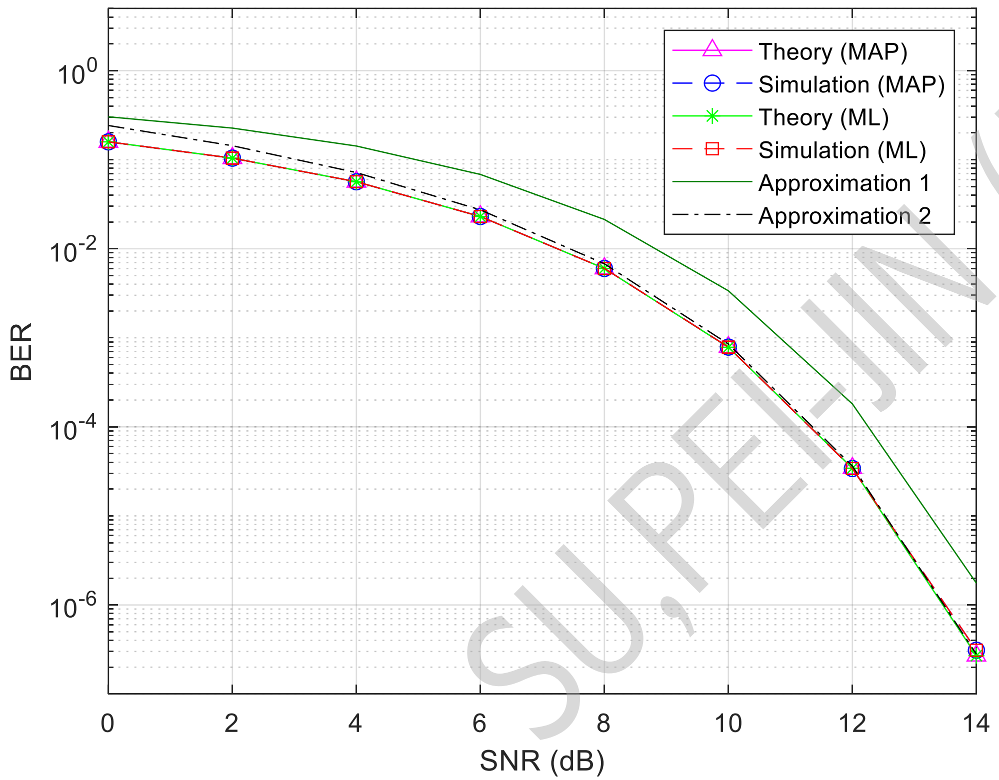
Figure 8.5
P_E for antipodal baseband digital signaling.

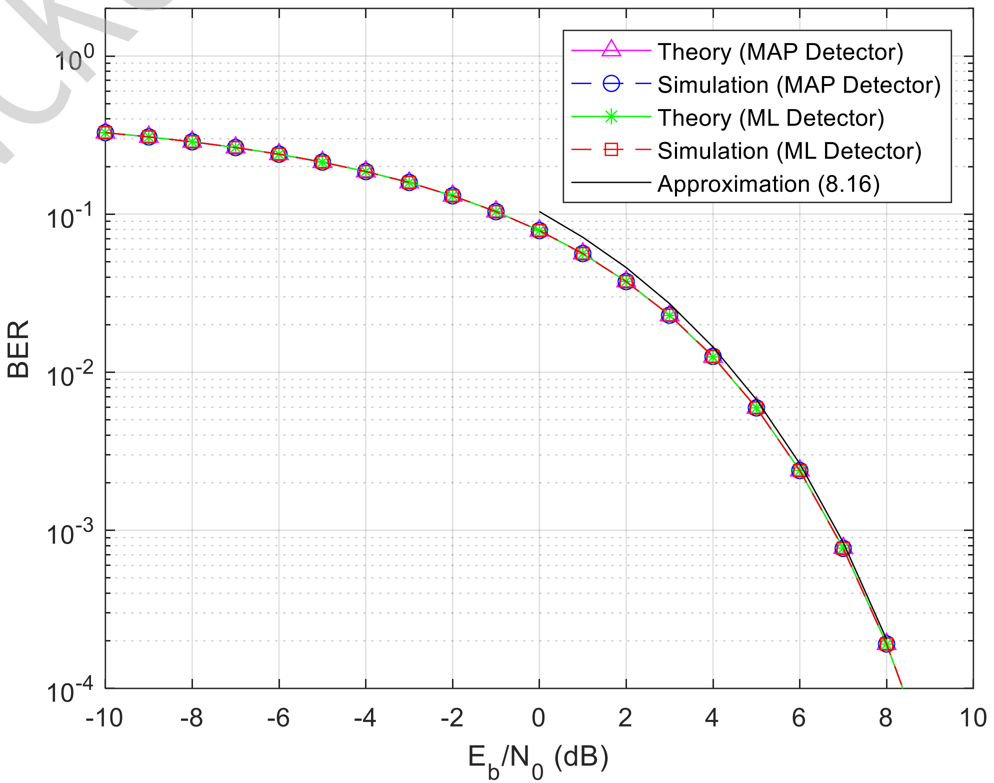
$z = \frac{E_b}{N_0}$



Equal probability of the transmitted bits

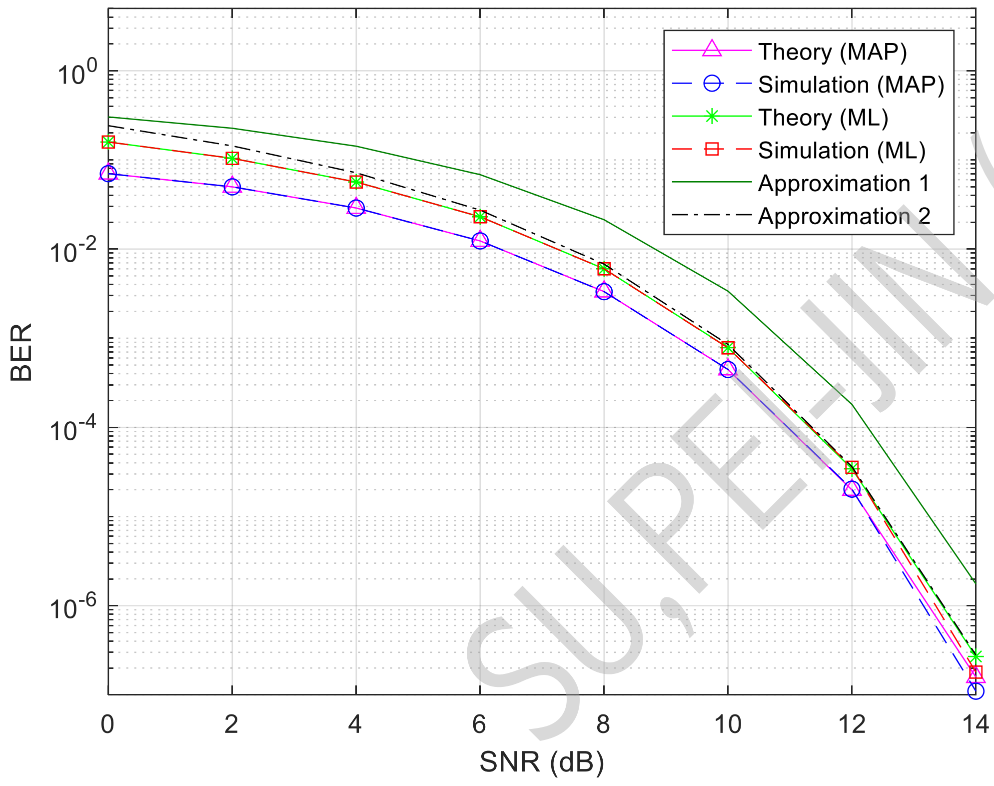
- $P(1) = P(s = \sqrt{2}) = P_1 = 0.5$ and $P(0) = P(s = -\sqrt{2}) = 1 - P_1 = 0.5$

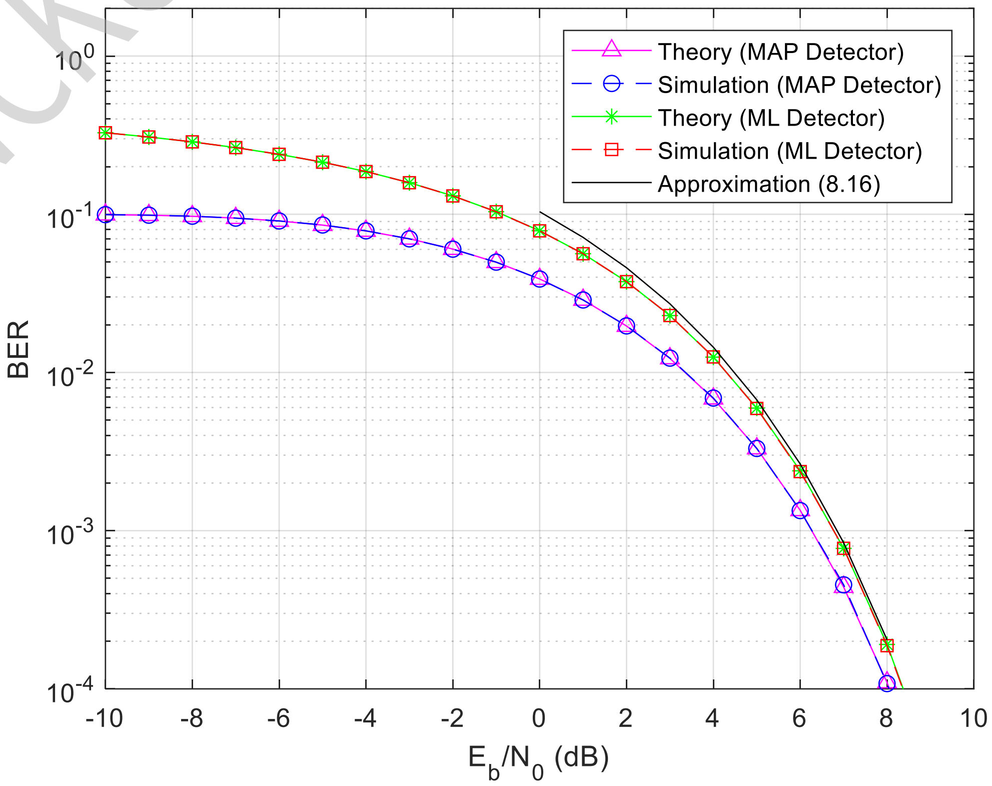




Non-equal probability of the transmitted bits

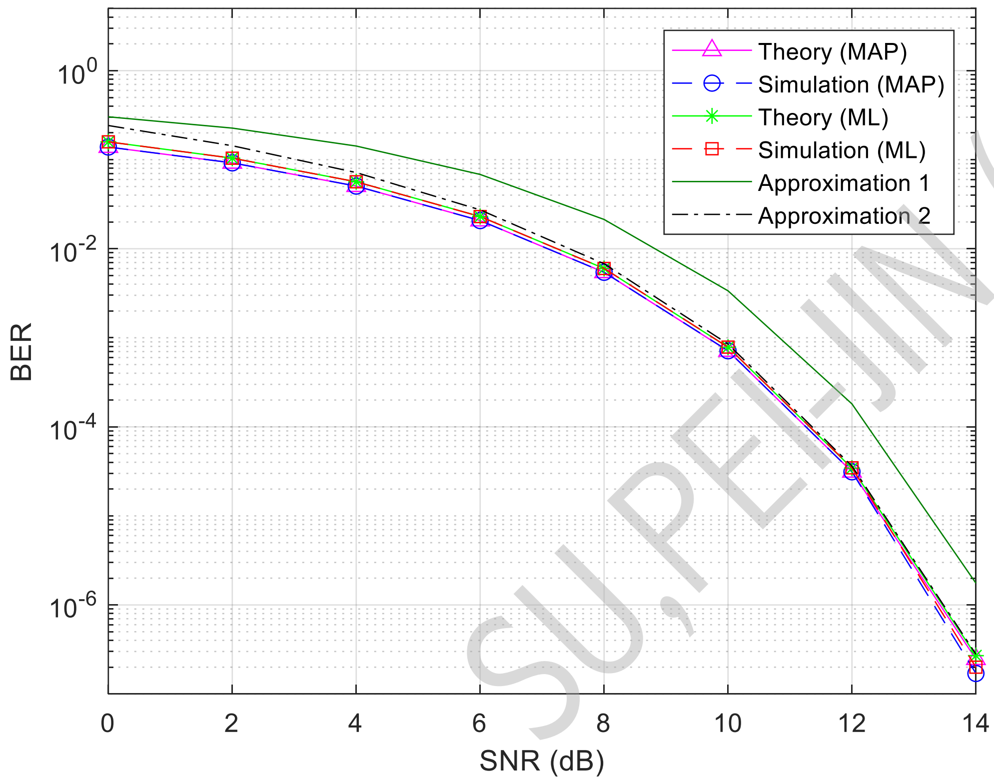
- $P(1) = P(s = \sqrt{2}) = P_1 = 0.1$ and $P(0) = P(s = -\sqrt{2}) = 1 - P_1 = 0.9$

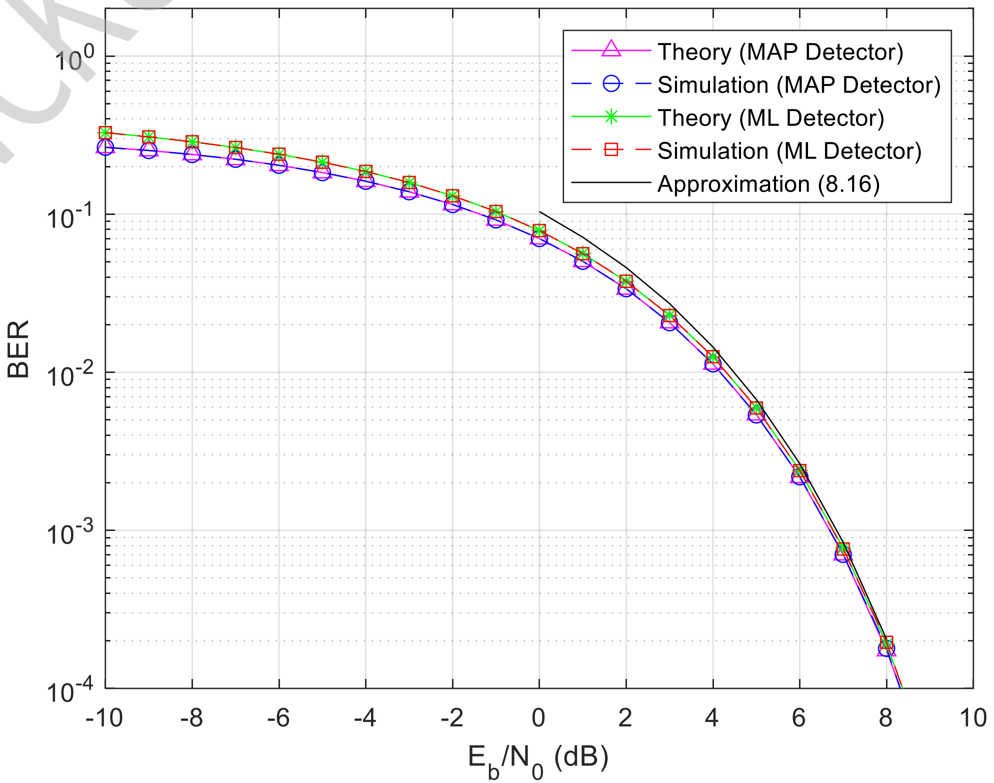




Non-equal probability of the transmitted bits

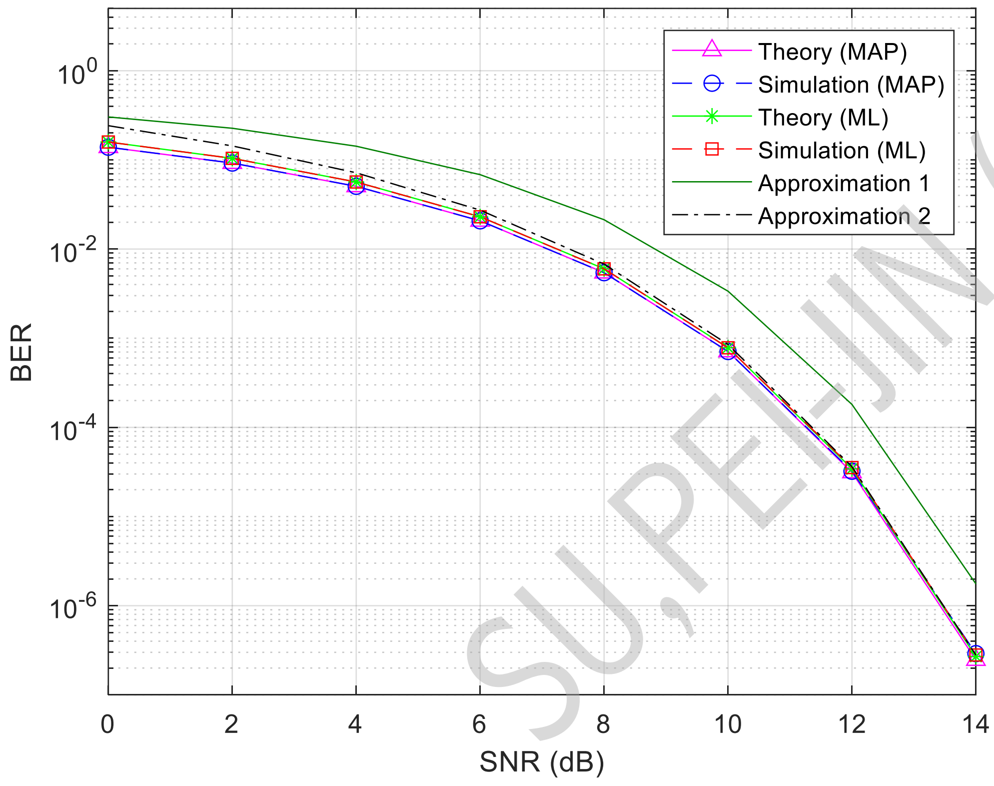
- $P(1) = P\left(s = \sqrt{2}\right) = P_1 = 0.3$ and $P(0) = P\left(s = -\sqrt{2}\right) = 1 - P_1 = 0.7$

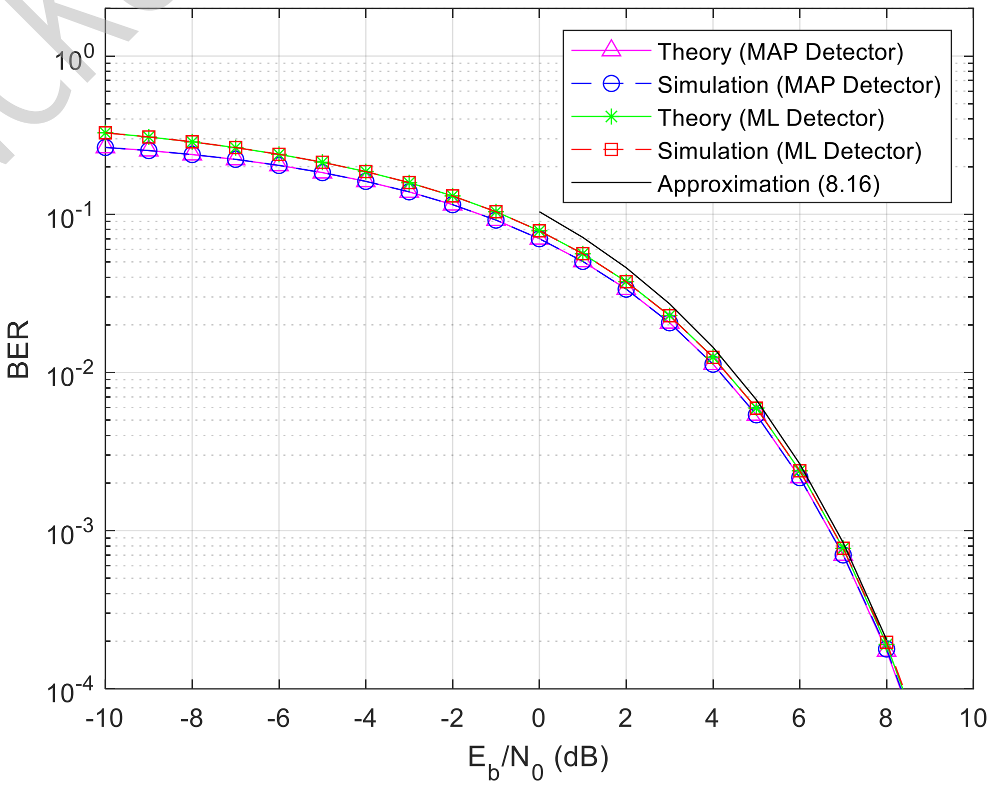




Non-equal probability of the transmitted bits

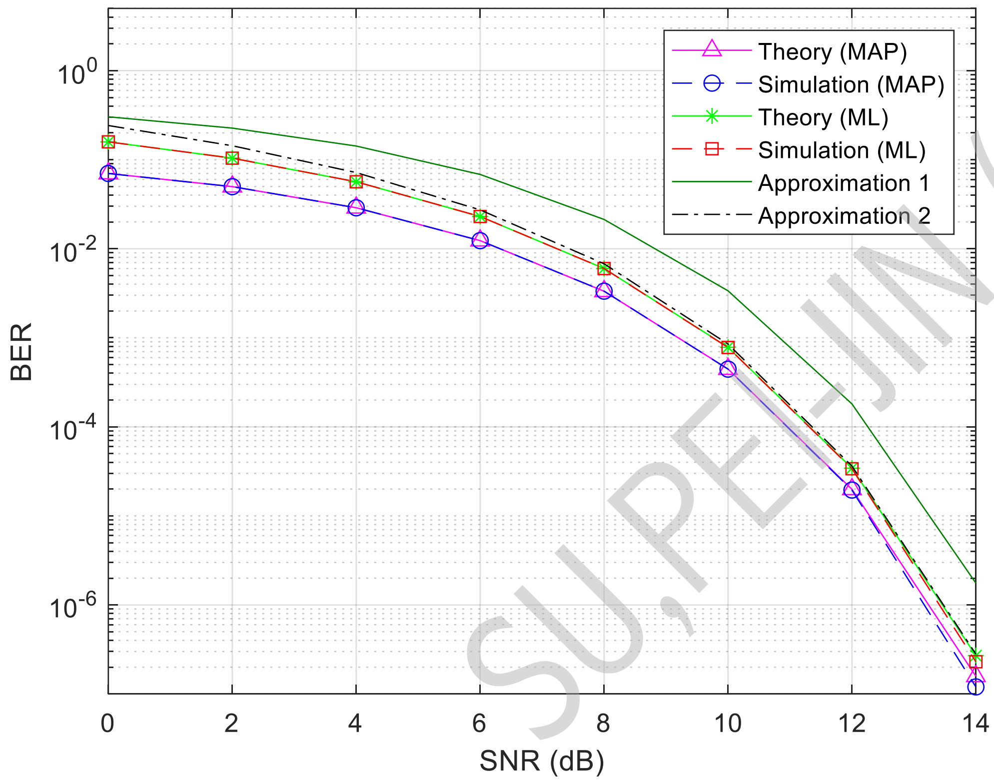
- $P(1) = P(s = \sqrt{2}) = P_1 = 0.7$ and $P(0) = P(s = -\sqrt{2}) = 1 - P_1 = 0.3$

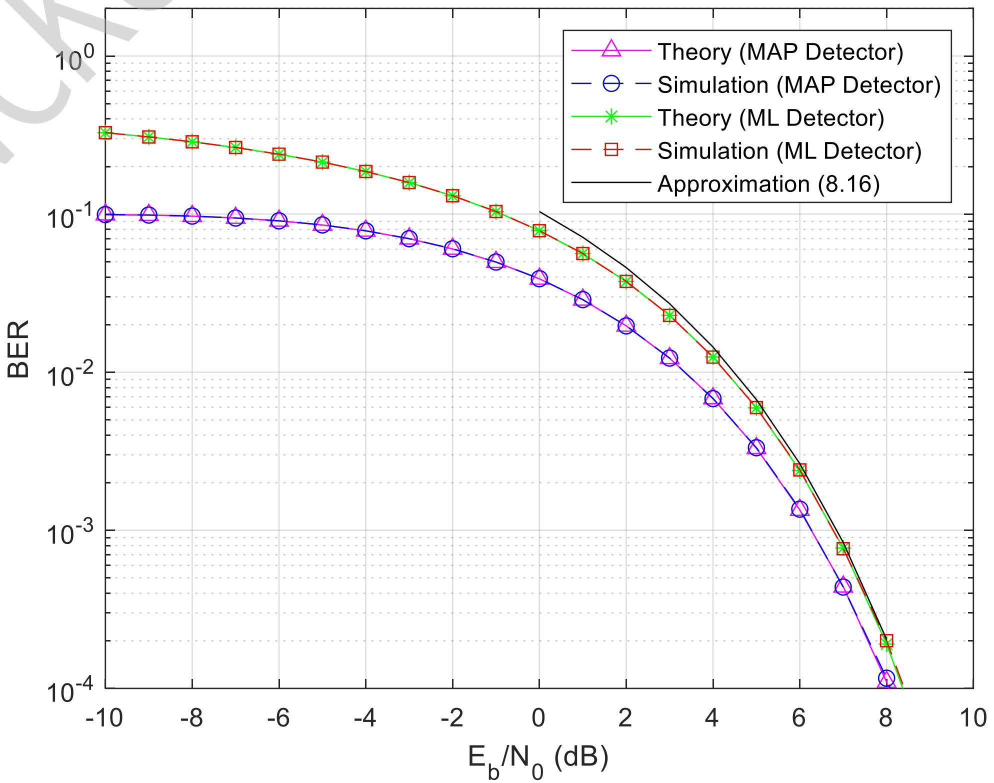




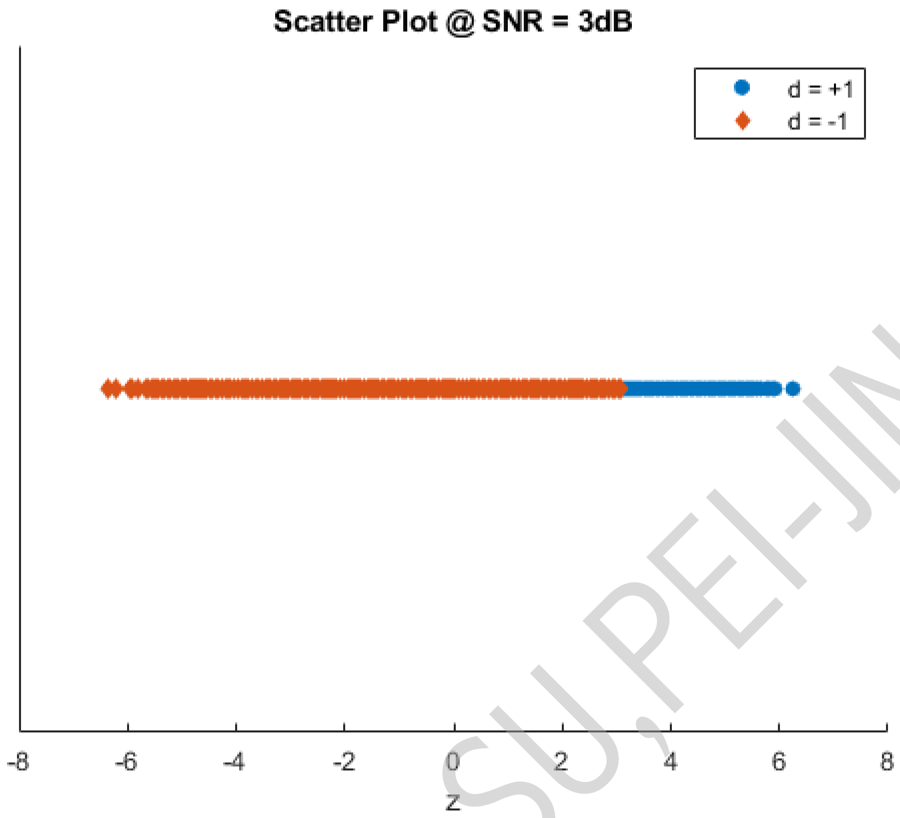
Non-equal probability of the transmitted bits

- $P(1) = P\left(s = \sqrt{2}\right) = P_1 = 0.9$ and $P(0) = P\left(s = -\sqrt{2}\right) = 1 - P_1 = 0.1$

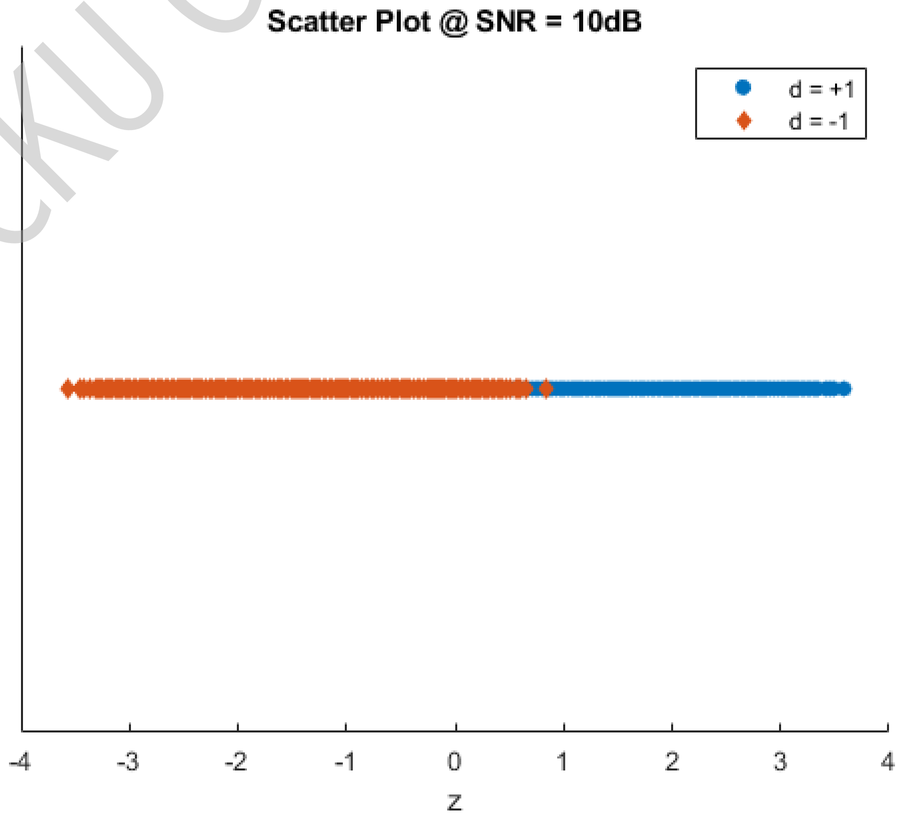




Scatter Plot @ SNR = 3dB & SNR = 10dB

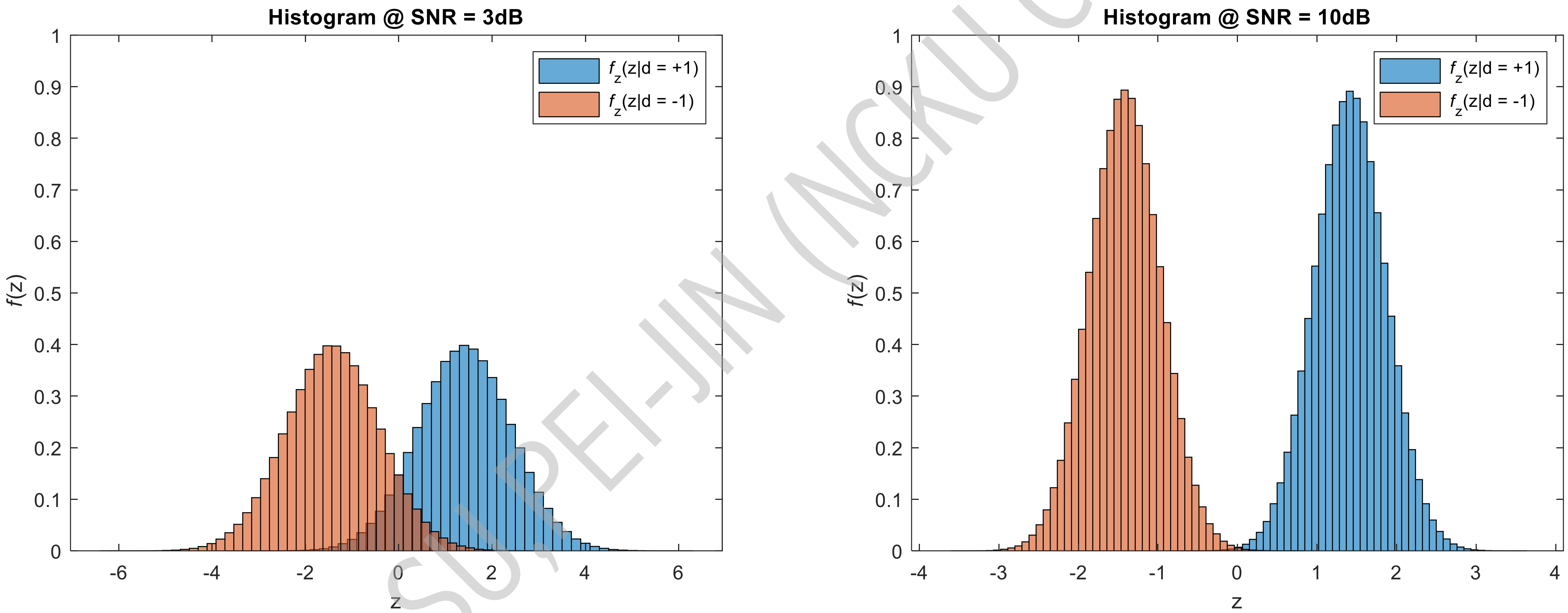


$$z = s + w = N\left(d \cdot \sqrt{2}, N_0\right)\ ,\ N_0 \approx 1.0024$$



$$z = s + w = N\left(d \cdot \sqrt{2}, N_0\right)\ ,\ N_0 \approx 0.2$$

Histogram @ SNR = 3dB & SNR = 10dB



$$z = s + w = N\left(d \cdot \sqrt{2}, N_0\right)\ ,\ N_0 \approx 1.0024$$

$$z = s + w = N\left(d \cdot \sqrt{2}, N_0\right)\ ,\ N_0 \approx 0.2$$

My MATLAB Codes

```
%%% BPSK Simulation Using MATLAB

clc;        %% Clear command window
clear;      %% Remove items from workspace

SNR_dB = 0:2:14;        %% SNR (= 2Eb/N0) in dB
EbOverN0_dB = SNR_dB - 10*log10(2);          %% Eb/N0 in dB
EbOverN0 = 10.^(EbOverN0_dB/10);             %% Eb/N0

Eb = 1;   %% Bit energy (J)
T = 1;    %% Bit interval (s)
num_bits = 10^8;        %% Number of data bit
N0 = Eb./EbOverN0;
sigma = sqrt(T*N0);
BPSK = [-sqrt(2*Eb*T),sqrt(2*Eb*T)];         %% Baseband modulator
P1 = 0.5;               %% The a priori probability: P(1) = 0.5
r_th = 0.5*log((1 - P1)/P1)*(T*N0)/sqrt(2*Eb*T);          %% The threshold value
num_error_MAP = zeros(size(EbOverN0_dB)); %% Number of error bits (MAP)
num_error_ML = zeros(size(EbOverN0_dB));  %% Number of error bits (ML)
```

My MATLAB Codes

```
% The theoretical BER of the MAP detector
MAP_constant_1 = sqrt(2*Eb*T) - r_th;
MAP_constant_2 = r_th + sqrt(2*Eb*T);
BER_theory_MAP = P1*qfunc(MAP_constant_1./sigma) + ...
                 (1-P1)*qfunc(MAP_constant_2./sigma);

% The theoretical BER of the ML detector
BER_theory_ML = qfunc(sqrt(2*EbOverN0));

% Approximation 1 of Q(x)
BER_approx_1 = 0.5*exp(-EbOverN0);

% Approximation 2 of Q(x)
BER_approx_2 = 0.5*exp(-EbOverN0)./sqrt(pi*EbOverN0);
```

My MATLAB Codes

```
for snrIdx = 1: length(EbOverN0_dB)
    tx_bit = rand(1, num_bits) > (1-P1);   %% Transmitted data bit
    tx_sym = BPSK(tx_bit(1:num_bits) + 1);                     %% Transmitted symbol

    % AWGN channel
    rx_sym = tx_sym + randn(1, length(tx_sym))*sigma(snrIdx);

    %%% MAP Detector %%%
    rx_bit_MAP = zeros(size(tx_bit));       %% Reset rx MAP decisions
    rx_bit_MAP(rx_sym > r_th(snrIdx)) = 1;                     %% MAP Decision of bit '1'
    err_pat_MAP = xor(tx_bit, rx_bit_MAP);                     %% Compare tx and rx bits (MAP)
    % Error counting (MAP criterion)
    num_error_MAP(snrIdx) = num_error_MAP(snrIdx) + sum(err_pat_MAP);

    %%% ML Detector %%%
    rx_bit_ML = zeros(size(tx_bit));         %% Reset rx ML decisions
    rx_bit_ML(rx_sym > 0) = 1;               %% ML Decision of bit '1'
    err_pat_ML = xor(tx_bit, rx_bit_ML);                       %% Compare tx and rx bits (ML)
    % Error counting (ML criterion)
    num_error_ML(snrIdx) = num_error_ML(snrIdx) + sum(err_pat_ML);
end
```

My MATLAB Codes

```
% The simulated BER of the optimal detector (i.e., MAP)
BER_sim_MAP = num_error_MAP/num_bits;

% The simulated BER of the ML detector
BER_sim_ML = num_error_ML/num_bits;

% Plot the error rate curves
clf;
semilogy(SNR_dB, BER_theory_MAP, '-^m');
hold on;
semilogy(SNR_dB, BER_sim_MAP, '--ob');
hold on;
semilogy(SNR_dB, BER_theory_ML, '-*g');
hold on;
semilogy(SNR_dB, BER_sim_ML, '--sr');
hold on;
```

My MATLAB Codes

```
semilogy(SNR_dB, BER_approx_1, 'Color', '#008000');
hold on;
semilogy(SNR_dB, BER_approx_2, '-.k');
axis([0,14,1e-7,5]);

legend('Theory (MAP)','Simulation (MAP)', ...
'Theory (ML)','Simulation (ML)', ...
'Approximation 1','Approximation 2');
xlabel('SNR (dB)');
ylabel('BER');
grid;
```

My MATLAB Codes

```
%%% Scatter Plot and Histogram @ SNR = 3dB and SNR = 10dB

clc;        %% Clear command window
clear;      %% Remove items from workspace

Eb = 1;  %% Bit energy (J)
T = 1;     %% Bit interval (s)
num_bits = 10^6;     %% Number of data bit
%%% AWGN channel (SNR = 3dB) %%%
SNR_3dB = 10.^(3/10);
N0_3dB = 2*Eb/SNR_3dB;          %% SNR = 2*Eb/N0
sigma_3dB = sqrt(N0_3dB*T);

% When data bit d = +1 is transmitted (Source bit: 1)
z_1_3dB = sqrt(2*Eb*T) + randn(1, num_bits)*sigma_3dB;
% When data bit d = -1 is transmitted (Source bit: 0)
z_0_3dB = -sqrt(2*Eb*T) + randn(1, num_bits)*sigma_3dB;
```

My MATLAB Codes

```
%%% AWGN channel (SNR = 10dB) %%%
SNR_10dB = 10.^(10/10);
N0_10dB = 2*Eb/SNR_10dB; %% SNR = 2*Eb/N0
sigma_10dB = sqrt(N0_10dB*T);

% When data bit d = +1 is transmitted (Source bit: 1)
z_1_10dB = sqrt(2*Eb*T) + randn(1, num_bits)*sigma_10dB;
% When data bit d = -1 is transmitted (Source bit: 0)
z_0_10dB = -sqrt(2*Eb*T) + randn(1, num_bits)*sigma_10dB;

%%% Plot the scatter plot and histogram @ SNR = 3dB %%%
clf;
y_Idx = zeros(1, num_bits);
scatter(z_1_3dB,y_Idx,'filled');
hold on;
scatter(z_0_3dB,y_Idx,'filled','d');
title('Scatter Plot @ SNR = 3dB');
yticks([]);
legend('d = +1','d = -1');
xlabel('z');
```

My MATLAB Codes

```
figure;
nbins = 50;
h1 = histogram(z_1_3dB,nbins,'Normalization','pdf');
hold on;
h2 = histogram(z_0_3dB,nbins,'Normalization','pdf');
ylim([0 1]);
title('Histogram @ SNR = 3dB');
legend('\it f\rm_z(z|d = +1)','\it f\rm_z(z|d = -1)');
xlabel('z');
ylabel('\it f\rm(z)');

%%% Plot the scatter plot and histogram @ SNR = 10dB %%%
figure;
scatter(z_1_10dB,y_Idx,'filled');
hold on;
scatter(z_0_10dB,y_Idx,'filled','d');
title('Scatter Plot @ SNR = 10dB');
yticks([]);
legend('d = +1','d = -1');
xlabel('z');
```

My MATLAB Codes

```
figure;
h3 = histogram(z_1_10dB,nbins,'Normalization','pdf');
hold on;
h4 = histogram(z_0_10dB,nbins,'Normalization','pdf');
x_axis_max = sqrt(2*Eb*T) + 6*sigma_10dB;
x_axis_min = -sqrt(2*Eb*T) - 6*sigma_10dB;
axis([x_axis_min x_axis_max 0 1]);
title('Histogram @ SNR = 10dB');
legend('\it f\rm_z(z|d = +1)','\it f\rm_z(z|d = -1)');
xlabel('z');
ylabel('\it f\rm(z)');
```