For M=4

1. Function Code:

% Initialization of the function

function x = GaussQ( n)

% x = GaussQ( number)

%

% This function generates the value of Qfunction of a given number.

%

% Input: ( n ) = number

%

% Output : Q function applied value of n

% Date created: 07/06/2016 Author: Tamoghna Chattopadhyay

%

% -----------------------------------------------------------------

% Function body

x = 0.5 \* erfc(n/sqrt(2));

return;

1. Main Code:

clear all

clc

M = 4; %Number of possible messages

gdB = 0:.01:20; %gamma\_b in dB

g = 10 .^ (gdB/10); %actual value of gamma\_b

for i = 1:length(g)

Pb\_MPAM(i) = (2 \* (M-1) \* GaussQ ( sqrt( (6 \* g(i) \* log2(M))/ (M\*M - 1 )))) / (M \* log2(M));

Pb\_MPSK(i) = (2 \* GaussQ ( (sqrt( 2 \* g(i) \* log2(M))) \* sin (pi/M))) / log2(M);

Pb\_MQAM(i) = (4 \* GaussQ ( sqrt( (3 \* g(i) \* log2(M))/ (M - 1 )))) / log2(M);

end

%Obtain Plots

semilogy(gdB,Pb\_MPAM,'r.','LineWidth',3);

hold on;

grid;

semilogy(gdB,Pb\_MPSK,'b--','LineWidth',3);

semilogy(gdB,Pb\_MQAM,'k-','LineWidth',3);

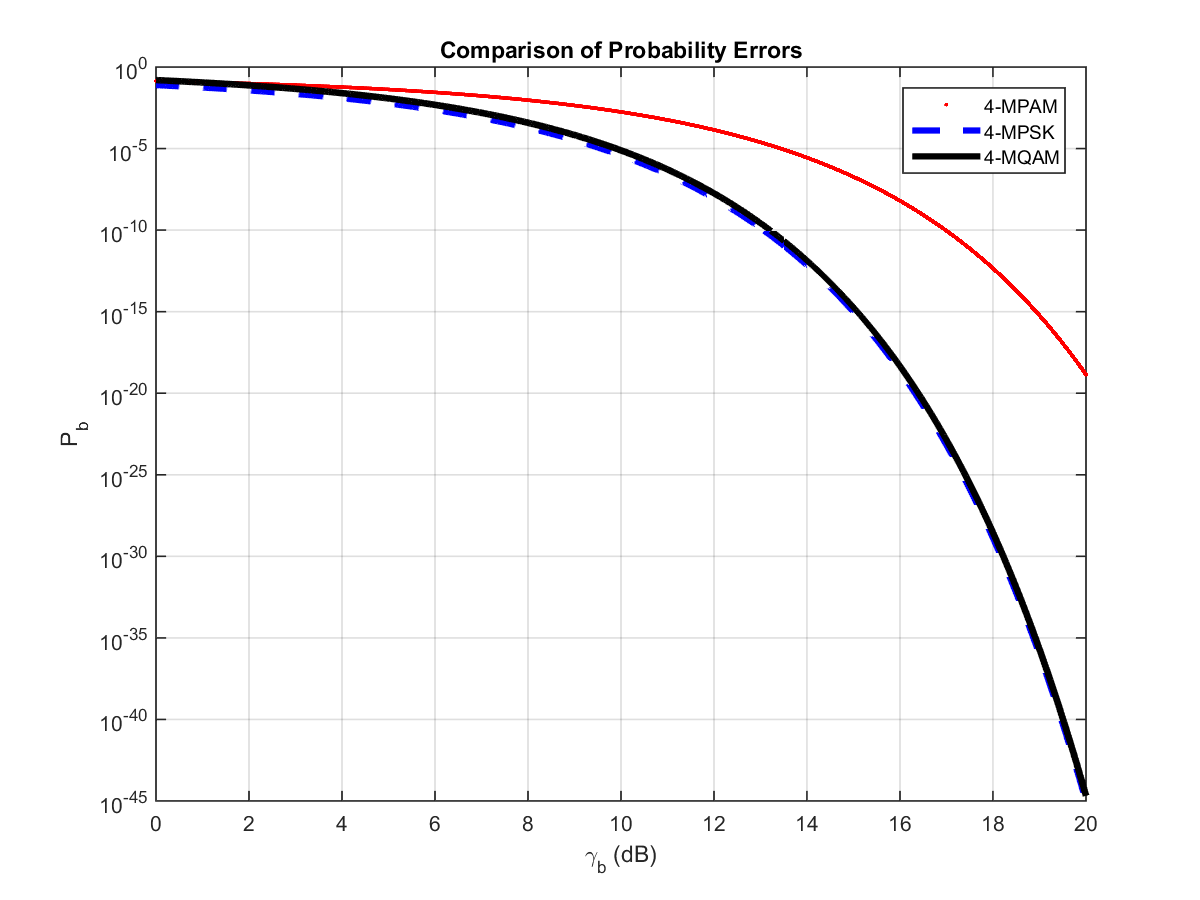
legend(strcat(num2str(M),'-MPAM'),strcat(num2str(M),'-MPSK'),strcat(num2str(M),'-MQAM'));

title('Comparison of Probability Errors');

xlabel('\gamma\_b (dB)');

ylabel('P\_b');

hold off;



For M=8

1. Function Code:

% Initialization of the function

function x = GaussQ( n)

% x = GaussQ( number)

%

% This function generates the value of Qfunction of a given number.

%

% Input: ( n ) = number

%

% Output : Q function applied value of n

% Date created: 07/06/2016 Author: Tamoghna Chattopadhyay

%

% -----------------------------------------------------------------

% Function body

x = 0.5 \* erfc(n/sqrt(2));

return;

1. Main Code:

clear all

clc

M = 8; %Number of possible messages

gdB = 0:.01:20; %gamma\_b in dB

g = 10 .^ (gdB/10); %actual value of gamma\_b

for i = 1:length(g)

Pb\_MPAM(i) = (2 \* (M-1) \* GaussQ ( sqrt( (6 \* g(i) \* log2(M))/ (M\*M - 1 )))) / (M \* log2(M));

Pb\_MPSK(i) = (2 \* GaussQ ( (sqrt( 2 \* g(i) \* log2(M))) \* sin (pi/M))) / log2(M);

Pb\_MQAM(i) = (4 \* GaussQ ( sqrt( (3 \* g(i) \* log2(M))/ (M - 1 )))) / log2(M);

end

%Obtain Plots

semilogy(gdB,Pb\_MPAM,'r.','LineWidth',3);

hold on;

grid;

semilogy(gdB,Pb\_MPSK,'b--','LineWidth',3);

semilogy(gdB,Pb\_MQAM,'k-','LineWidth',3);

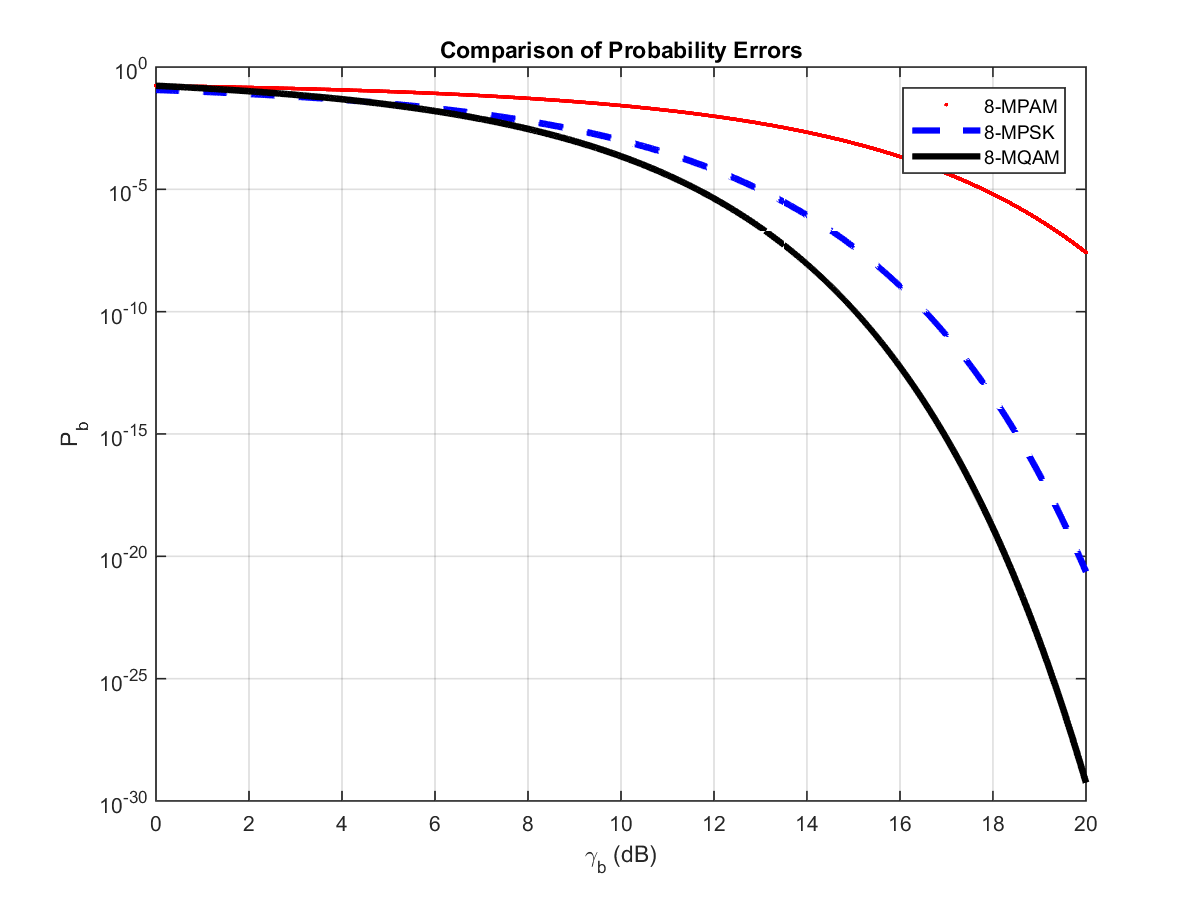
legend(strcat(num2str(M),'-MPAM'),strcat(num2str(M),'-MPSK'),strcat(num2str(M),'-MQAM'));

title('Comparison of Probability Errors');

xlabel('\gamma\_b (dB)');

ylabel('P\_b');

hold off;



For M=16

1. Function Code:

% Initialization of the function

function x = GaussQ( n)

% x = GaussQ( number)

%

% This function generates the value of Qfunction of a given number.

%

% Input: ( n ) = number

%

% Output : Q function applied value of n

% Date created: 07/06/2016 Author: Tamoghna Chattopadhyay

%

% -----------------------------------------------------------------

% Function body

x = 0.5 \* erfc(n/sqrt(2));

return;

1. Main Code:

clear all

clc

M = 16; %Number of possible messages

gdB = 0:.01:20; %gamma\_b in dB

g = 10 .^ (gdB/10); %actual value of gamma\_b

for i = 1:length(g)

Pb\_MPAM(i) = (2 \* (M-1) \* GaussQ ( sqrt( (6 \* g(i) \* log2(M))/ (M\*M - 1 )))) / (M \* log2(M));

Pb\_MPSK(i) = (2 \* GaussQ ( (sqrt( 2 \* g(i) \* log2(M))) \* sin (pi/M))) / log2(M);

Pb\_MQAM(i) = (4 \* GaussQ ( sqrt( (3 \* g(i) \* log2(M))/ (M - 1 )))) / log2(M);

end

%Obtain Plots

semilogy(gdB,Pb\_MPAM,'r.','LineWidth',3);

hold on;

grid;

semilogy(gdB,Pb\_MPSK,'b--','LineWidth',3);

semilogy(gdB,Pb\_MQAM,'k-','LineWidth',3);

legend(strcat(num2str(M),'-MPAM'),strcat(num2str(M),'-MPSK'),strcat(num2str(M),'-MQAM'));

title('Comparison of Probability Errors');

xlabel('\gamma\_b (dB)');

ylabel('P\_b');

hold off;

