



Trabalho Computacional 1

Sistema de Comunicações Digitais

INSTRUÇÕES

- O trabalho deverá ser enviado via sigaa. Ele deverá ser enviado zipado contendo os seguintes arquivos:
 - Relatório em formato pdf nomeado como "NOMEALUNO_MATRICULA.pdf". (Se for usado notebook como relatório, gere uma versão em PDF do notebook e nomeie-o como indicado)
 - Os códigos utilizados para gerar os resultados desta atividade.
 - Para os arquivos de programa, eles deverão conter comentários acerca de cada passo desenvolvido. Cada parâmetro definido, cada função criada, cada bloco desenvolvido. Um bom exemplo é o pseudo-código no final deste Homework.
- Nomear o arquivo compactado com sua matrícula.
- O trabalho poderá ser desenvolvido em qualquer linguagem/ambiente de programação.
- ATENÇÃO: **NÃO** será permitida a utilização de nenhuma função presente nos softwares tais como o MATLAB, com exceção das funções de plotagem e da função Q utilizada na probabilidade de erro teórica.

Em caso de dúvidas não existem em entrar em contato via e-mail: yurisales@gtel.ufc.br.

Boa sorte!

Problema 1: Considere a modulação M -QAM, em que o sinal em banda base é dado por:

$$s_m(t) = \left(\frac{A_m^{(\text{real})}}{\sqrt{2}} + \frac{jA_m^{(\text{imag})}}{\sqrt{2}} \right) g(t),$$

em que $g(t)$ é pulso transmitido, $A_m^{(\text{real})}$ e $A_m^{(\text{imag})}$ são as amplitudes da parte real e imaginária da forma de onda transmitida, respectivamente.

Considere $\int_{-\infty}^{\infty} |g(t)|^2 dt = \mathcal{E}_g = 2$. Suponha a transmissão de uma sequência de símbolos $\{s_m\}$ de tamanho $L = 264000$ bits. Após o filtro casado e desconsiderando qualquer ruído, temos na entrada do decisor:

$$z_m = s_m$$

- 1) Para $M = \{4, 16, 64\}$, determine a energia média \mathcal{E}_m de cada constelação;
- 2) Para $M = \{4, 16, 64\}$, determine a distância mínima d_{\min} entre dois símbolos;
- 3) Para $M = \{4, 16, 64\}$, implemente o modulador (mapeamento bit-símbolo) usando a codificação de Gray;
- 4) Para $M = \{4, 16, 64\}$, implemente o demodulador (mapeamento símbolo-bit).

Exemplo: O *scatterplot* da constelação 16-QAM é mostrado na Figura 1.

Dica 1: Uma constelação M -QAM pode ser vista como um produto cartesiano de duas constelações \sqrt{M} -PAM.

Dica 2: Para visualizar a constelação, pode-se usar a função *scatter* do MATLAB.

Dica 3: Você pode validar sua função de modulação e demodulação M -QAM, comparando a sequência original de bits gerada antes da modulação com aquela obtida após a demodulação. Se a implementação do modulador/demodulador estiver correta, ambas as sequências serão idênticas.

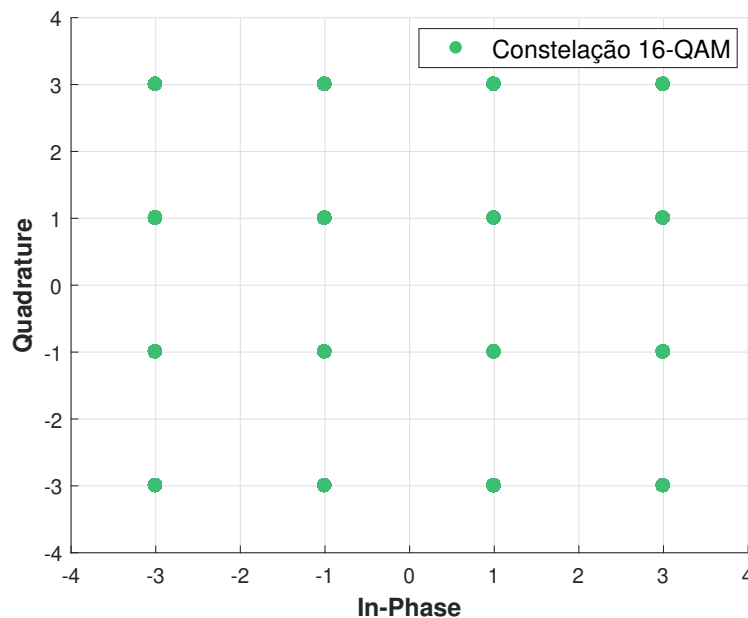


Fig. 1. Constelação 16-QAM.

Problema 2: Considere um sistema de comunicação digital M -QAM na presença de ruído additivo Gaussiano (canal AWGN). O sinal recebido após o filtro casado (MF) e amostrado é dado por:

$$y_m = s_m + n_m,$$

em que s_m é o símbolo transmitido e n_m representa o termo de ruído aditivo, modelado como uma variável aleatória Gaussiana complexa com média zero e variância N_0 , ou seja, $\mathcal{CN}(0, N_0)$. Para $M = \{4, 16, 64\}$, plote a curva da *probabilidade de erro teórica*, com o auxílio da função $\mathcal{Q}(\cdot)$, em função da razão \mathcal{E}_m/N_0 , expressa em dB, e definida por

$$(\mathcal{E}_m/N_0)_{\text{dB}} = 10\log_{10}\left(\frac{\mathcal{E}_m}{N_0}\right),$$

considerando a faixa de valores entre 0 e 20dB (utilize um passo de 2dB).

Exemplo: A curva de probabilidade de erro teórica para a modulação 4-QAM é mostrada na Figura 2.

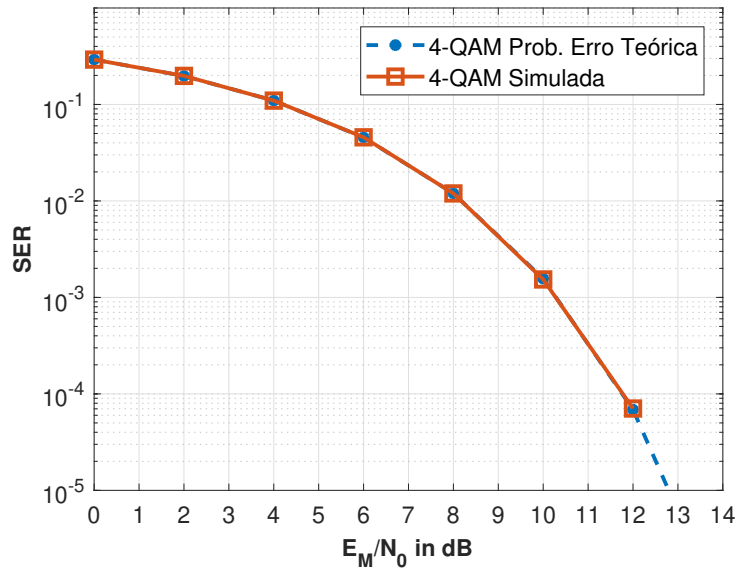


Fig. 2. Curva de probabilidade de erro teórica para a modulação 4-QAM.

Problema 3: Considerando ainda um sistema de comunicação digital M -QAM em canal AWGN:

- 1) Obtenha a curva da taxa de erro de símbolos (SER - *Symbol Error Rate*) simulada em função da razão \mathcal{E}_m/N_0 .

Plote as curvas de SER para $M = 4, 16$ e 64 na mesma figura;

- 2) Obtenha a curva da taxa de erro de bits (BER - *Bit Error Rate*) simulada em função da razão \mathcal{E}_m/N_0 .

Plote as curvas de BER para $M = 4, 16$ e 64 na mesma figura.

Dica: A taxa de erro de símbolo é calculada da seguinte forma:

$$R = \frac{\text{Número total de símbolos errados}}{\text{Número total de símbolos transmitidos}}.$$

Problema 4: Repita os Problemas 1, 2 e 3 considerando agora a modulação M -PSK, para $M = \{4, 8\}$. Considere um mapeamento de bit-símbolo usando a codificação de Gray (como no Problema 01). Para o M -PSK neste problema, assuma $\mathcal{E}_m = 1$.

Problema 5: Análise comparativa de diferentes técnicas modulações digitais.

- 1) Em uma única figura, sobreponha as curvas de SER das constelações utilizadas neste trabalho, isto é, $\{4, 16, 64\}$ -QAM e $\{4, 8\}$ -PSK;
- 2) Em uma única figura, sobreponha as curvas de BER das constelações utilizadas neste trabalho, isto é, $\{4, 16, 64\}$ -QAM e $\{4, 8\}$ -PSK.
- 3) Discuta livremente sobre os resultados observados na figuras acima, comparando o desempenho das diferentes modulações, à luz de questões tais como eficiência espectral, taxa de transmissão, e eficiência energética.

Ajustando a razão \mathcal{E}_m/N_0 nas simulações

Para plotar as curvas de SER e BER nas simulações, temos de ajustar a razão $(\mathcal{E}_m/N_0)_{\text{dB}}$ para cada valor de interesse. A razão em escala logarítmica (dB) permite uma melhor visualização das curvas de SER/BER em uma faixa mais ampla de valores. Por exemplo, $(\mathcal{E}_m/N_0)_{\text{dB}} = 30$ significa que a energia média do sinal transmitido \mathcal{E}_m é 1000 vezes maior que a densidade espectral do ruído N_0 . Já para $(\mathcal{E}_m/N_0)_{\text{dB}} = 10$, temos que \mathcal{E}_m é “apenas” 10 vezes maior que N_0 . Logo, temos a seguinte relação:

$$(\mathcal{E}_m/N_0)_{\text{dB}} = 10 \log_{10} (\mathcal{E}_m/N_0)_{\text{linear}},$$

ou seja,

$$(\mathcal{E}_m/N_0)_{\text{linear}} = 10^{(\mathcal{E}_m/N_0)_{\text{dB}}/10}$$

Logo, nos Problemas 2, 3 e 4, para simularmos a SER/BER para uma dada razão \mathcal{E}_m/N_0 (em dB), precisamos primeiramente converter esta razão para a escala linear. Em seguida, encontra-se a variância do termo de ruído associada à esta razão. Baseado na expressão acima, obtemos:

$$N_0 = \mathcal{E}_m \cdot 10^{-(\mathcal{E}_m/N_0)_{\text{dB}}/10}.$$

Note que a variância ruído por dimensão é dada por $\sigma_n^2 = N_0/2$, representando a potência média do ruído que afeta cada dimensão do sinal em banda base recebido (tanto a parte real como a parte imaginária) de forma igual. Portanto, ao gerarmos o termo de ruído nas simulações, pondera-se sua parte real e sua parte imaginária por um fator igual a $\sqrt{\sigma_n^2} = \sqrt{N_0/2}$, que corresponde ao valor médio de amplitude (desvio padrão) do ruído.

Exemplo de geração do termo de ruído:

```
noise= sqrt (No/2) * (randn (N,1)+1i*randn (N,1)) ; % ruido gaussiano complexo
```

Exemplo: Pseudo-código para 4-PAM

```
% Parametros 4-PAM
M = 4; % - Número de símbolos na constelação
L = 4096; % - Tamanho da sequencia transmitida (em bits)
K = log2 (M); % - Número de bits por símbolo
N = L/K; % - Número de símbolos a serem transmitidos
% Razão sinal-ruído
Em_No = 30; % escala dB

% Energia e distância mínima d constelação 4-PAM
Em = 5 * (c^2);
d_min = sqrt (12 * Em / (M^2 - 1)); % distancia mínima entre símbolos 2*c
```

```

c = d/2;

% gerando uma sequencia de bits
bits= randi(2,1,L)-1;

% mapeando os bits em símbolos (usando cod. gray)
symb= zeros(N,1); k=1;
for i=1:N
    symb(i)=pam4_mapping(bits(k:k+K-1),c);
    k=k+K;
end
% checando a constelação
% função: scatter(var_eixo_x,var_eixo_y,size_marker,[ R G B])
% RGB varia entre 0 e 1
% a opção 'filled' é para preencher o círculo (opcional)
scatter(symb,zeros(size(symb,1),1),75,[0.25 0.2 0.65],'filled')
xlabel('In-Phase')
ylabel('Quadrature')
title('4-PAM')
% legend('4-PAM constelação')
grid on

% gerando o termo de ruído
No= Es*10^(-Em_No/10);
noise= sqrt(No/2)*(randn(N,1)+1i*randn(N,1)); % ruido gaussiano complejo

% sinal recebido (após filtro casado e amostragem)
ym = symb + noise;

% TO DO: Construa o seu decisor slice

symb_dec=zeros(N,1);
bits_dec=zeros(1,L);
k=1;
for i=1:N
    [symb_dec(i),bits_dec(k:k+K-1)]= decisor_4PAM(real(rx_signal(i)),c);
    k=k+K;

```

```
end
```

```
% TO DO: Calculo de SER e BER  
SER = minha_função_ser();  
BER = minha_função_ber();
```