

AE4 - Transmissão Binária e Análise de Desempenho de Erro

Jessica de Souza

Engenharia de Telecomunicações, Instituto Federal de Santa Catarina

[<jessica.souzajds@gmail.com>](mailto:jessica.souzajds@gmail.com)

26 de abril de 2018

1. Introdução

O objetivo deste trabalho é aplicar os conhecimentos adquiridos sobre o tema de transmissão binária e análise de desempenho de erro. Para isto, foram realizadas algumas simulações de transmissão binária com variação de parâmetros tais como: tipo de sinalização NRZ unipolar e bipolar, relação sinal-ruído (SNR), recepção com e sem filtro casado. As próximas seções irão descrever o embasamento teórico utilizado para este trabalho e os resultados da simulação.

2. Conceitos Abordados

O objetivo deste trabalho é aplicar os conhecimentos adquiridos sobre o tema de transmissão binária e análise de desempenho de erro. Para isto, foram realizadas algumas simulações de transmissão binária com variação de parâmetros tais como: tipo de sinalização NRZ unipolar e bipolar, relação sinal-ruído (SNR), recepção com e sem filtro casado. As próximas seções irão descrever o embasamento teórico utilizado para este trabalho e os resultados da simulação.

2.1. Representação de dígitos binária

Os dígitos binários são uma forma de representar a informação de forma abstrata. Em um meio de transmissão estes dígitos são convertidos para pulsos elétricos, para que se possa transmiti-los através de um canal banda-base[1].

Quando uma modulação de pulso é aplicada à um símbolo binário, a forma de onda binária resultante é chamada de modulação de código de pulso (PCM). Existem diversos tipos de formas de onda PCM, desde binários, até “M-ary”, onde M indica o número de níveis [1]. Alguns tipos de formas de onda PCM podem ser citadas: Não retorna à zero (NRZ), retorna à zero (RZ), fase codificada e binária multinível. Além disso, as formas de onda podem ser classificadas como unipolar (apenas um nível: 0 e A) e bipolar (de -A até A) ou com mais níveis. Esta classificação faz com que o nível de tensão varie de acordo com o valor desejado.

A transmissão PCM possui vantagens como: Eliminação da componente DC do sistema, facilita a sincronização e detecção de erro, compressão de largura de banda, boa performance mesmo com ruído. Neste trabalho será mantido o foco na forma de onda PCM-NRZ unipolar e bipolar [1].

2.2. Representação de um canal AWGN

O ruído térmico é causado pelo movimento aleatório dos elétrons em elementos dissipadores de energia. Ele é considerado um ruído branco gaussiano aditivo (AWGN), que é uma combinação linear de formas de onda ortogonais que aparecem no receptor de um sistema de comunicação. Ele é modelado como uma função gaussiana com média 0, densidade espectral de potência $N_0/2$ e está presente em todas as faixas de frequência do espectro.

Um sinal que é transmitido em um canal AWGN, possui como componentes o sinal $s(t)$ somado com um ruído $n(t)$. Para o receptor fazer a análise da forma de onda recebida, ele possui um limiar de decisão para que se identifique os símbolos do sinal. Dependendo da relação sinal-ruído (SNR) e da interferência intersimbólica (ISI), a detecção dos símbolos pode ser muito ou pouco afetada, podendo ser medida através da taxa de erro de bit (BER). Os conceitos de SNR, ISI e BER serão abordados nas seções a seguir.

2.3. Relação sinal-ruído (SNR)

Um sinal captado em um receptor AWGN, é constituído por duas componentes: o sinal e o ruído. A relação sinal-ruído, é a razão entre a potência média do sinal de

entrada e a potência média do ruído filtrado [2], de forma a mensurar a quantidade de ruído contida em um sinal. A equação a seguir descreve como obter a SNR de um sinal.

$$\frac{Eb}{N0} = \frac{S}{N} \left(\frac{W}{R} \right)$$

Onde $Eb/ N0$ é o mesmo que S/N , porém em um formato normalizado pela largura de banda W e taxa de bit R .

2.4. Interferência intersimbólica (ISI)

Em um sistema de comunicação existem diversos componentes, como por exemplo: filtros (com seus elementos reativos como indutores e capacitores), cabo com reatância que pode distorcer o sinal de forma linear. Essas distorções de sinal aumentam a duração do pulso, de forma que ocorre uma sobreposição entre os sinais enviados no canal. Este fenômeno é chamado de interferência intersimbólica. A Figura 1 mostra como ocorre a ISI.

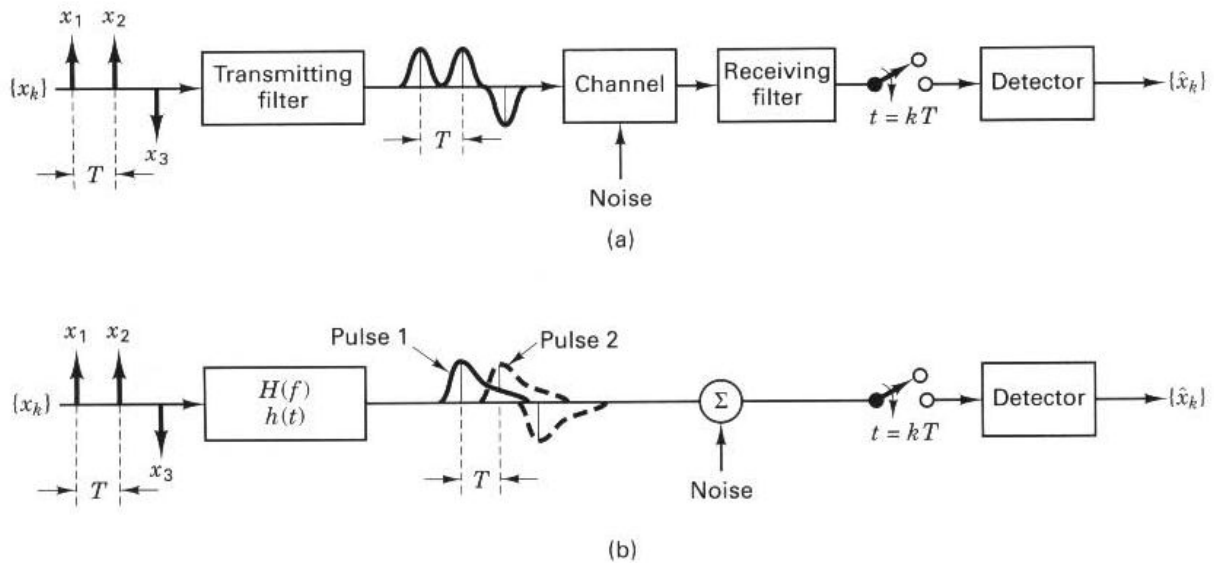


Figura 1 Interferência intersimbólica no processo de detecção.

2.5. Taxa de erro de bit

Os sistemas de comunicação contam com erros advindos de ruídos AWGN, interferências ISI, etc. Tais erros fazem com que em uma transmissão digital, os bits sejam detectados de forma errônea devido à grande variação dos bits [2]. É extremamente difícil possuir um sistema livre de erros, além de ser algo que requer investimentos de alto custo. Portanto, para que seja analisado qual a taxa de erro máxima permitida para tais sistemas, é utilizada a seguinte equação:

$$Pb = Q\left(\frac{a1-a2}{2\sigma_0}\right)$$

A equação de erro utilizando a função Q, nos permite mensurar o erro através da função gaussiana. Ela utiliza como parâmetros a diferença das amplitudes dos níveis a1 e a2, dividido por duas vezes o desvio padrão do ruído. Nota-se que a1 e a2 são onde ocorre o valor máximo da gaussiana. Para este relatório, contamos com a utilização das equações utilizadas em [1], para as sinalizações NRZ polar e bipolar, que estão descritas nas equações a seguir, respectivamente.

$$Pb_{unipolar} = Q\left(\sqrt{\frac{Eb}{N0}}\right) \quad Pb_{bipolar} = Q\left(\sqrt{\frac{2Eb}{N0}}\right)$$

Através do cálculo dessa probabilidade, podemos obter o gráfico de probabilidade de erro simbólica versus a relação Eb/N0, que mostra qual a probabilidade de erro de detecção de bit em relação da variação de Eb/N0.

2.6. Filtro casado

Para melhorar a detecção do sinal recebido, aplica-se um filtro no receptor com o mesmo comportamento do filtro no transmissor, porém invertido no tempo [2], este filtro é denominado filtro casado. Pode-se dizer que um filtro casado é uma função de transferência, com um atraso, o qual é o complexo conjugado da transformada de Fourier do sinal de entrada. A função principal do filtro casado é promover a maximização da relação sinal-ruído na saída do sistema [1].

3. Resultados e discussão

Os experimentos foram executados na ordem descrita na introdução. A seguir, o detalhamento de como foi realizado cada um deles. Foi utilizado o software Matlab para a realização dos experimentos.

3.1. Experimento 1

O primeiro experimento consistiu em simular uma transmissão binária cujos parâmetros estão descritos no código abaixo. O sinal gerado é um NRZ unipolar. Foi realizada a simulação da recepção com e sem filtro casado.

```
1 % Dados
2 info = [0 1 1 0 1 0 1 1 0 1 0]; % Informação, NRZ unipolar
3 SNR = 10; % dB
4 N = 100; % Fator de superamostragem
5 Rb = length(info); % Taxa de transmissao de bit
6 Fs = Rb*N; % Frequência de amostragem
7 t = [0:1/Fs:1-1/Fs];
```

Inicialmente foi gerado um filtro com o respectivo fator de superamostragem. Além disso, a informação gerada foi superamostrada para possuir mais pontos de pulso. O sinal foi passado pelo filtro gerado, de forma a possuir a forma de onda em pulso com duração N. Ao receber o sinal `info_hat`, foi inserido um ruído AWGN com a SNR desejada e a informação em bits foi obtida selecionando o sinal recebido em amostras no instante $T/2$. Para simular o filtro casado, antes de obter a `info_hat`, foi inserido um filtro igual ao filtro transmissor. O código abaixo mostra o desenvolvimento da simulação.

```

1 % Gerando o filtro
2 filtro_tx = ones(1,N); % pulso de 100 amostras
3 info_up = upsample(info,N); % insere 100 amostras a cada valor da info
4
5 % Passando o sinal pelo filtro
6 sinal_tx = filter(filtro_tx,1,info_up);
7
8 % Sinal recebido
9 sinal_rx = awgn(sinal_tx,SNR); % Ruído no receptor
10 Z_t = sinal_rx(N/2:N:end); % info hat
11
12 % Adicionando um filtro casado no receptor
13 filtro_rx = flip(filtro_tx);
14 sinal_rx_filt = filter(filtro_rx,1,sinal_rx)/N;
15 Z_t_filt = sinal_rx_filt(N:N:end); % info hat filtrado

```

A Figura 2 mostra as etapas de transmissão da informação. Em (a) o sinal superamostrado, logo após em (b) o sinal está filtrado e pronto para transmitir. Em (c) podemos observar o sinal recebido com ruído (SNR=10) e em (d) o sinal recebido e filtrado com o filtro casado. O sinal recebido em (d), deve ser analisada a sequência de bits recebida ao final de cada período T , que é onde o fim da transição de bit ocorre.

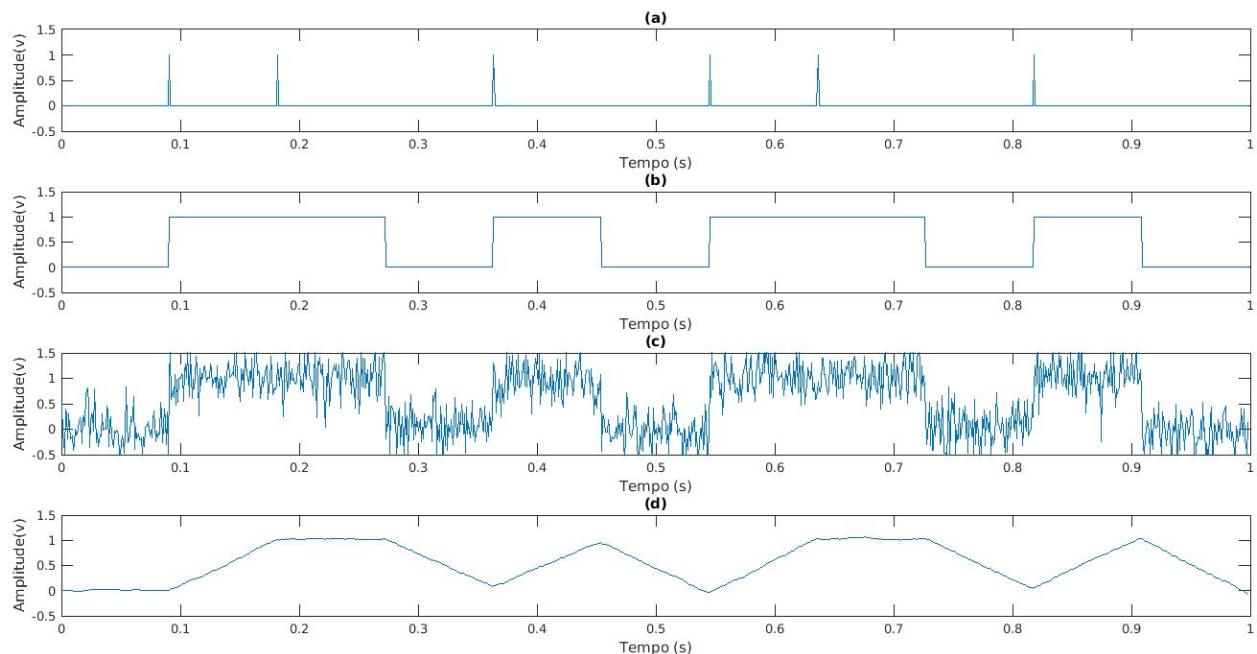


Figura 2 Resultados da simulação. (a) Sinal superamostrado, (b) sinal transmitido, (c) sinal recebido sem filtro e (d) sinal recebido com filtro.

3.2. Experimento 2

O segundo experimento consistiu em realizar uma série de simulações, para realizar a simulação de desempenho de erro através de um gráfico de probabilidade de erro de bit (Pb) versus a SNR em 4 cenários diferentes.

3.2.1. Cenário 1

O primeiro cenário é realizar a transmissão utilizando sinalização NRZ unipolar com as amplitudes e 1V e 2V, ambos sem utilizar filtro casado. Abaixo o código referente aos dados deste cenário.

```
1 % Dados
2 N = 100;      % Fator de superamostragem
3 SNR_max=15;   % Razao Sinal-Ruido
4 Rb = 10000;
5 A1 = 1;
6 A2 = 2;
7 limiar1 = 0.5;
8 limiar2 = 1;
9 Fs = Rb;
10 t = [0:1/Fs:1-1/Fs];
```

Foi gerada informação de forma aleatória utilizando a função randint(). Após isso, a informação foi superamostrada, passada pelo filtro transmissor, de forma a possuir a forma de onda em pulso com duração N. A seguir o código referente a estes passos.

```
11 % Gerando a informação
12 info= randint(1,Rb);
13
14 % Gerando os níveis do sinal;
15 info_nrz1 = info * A1;
16 info_nrz2 = info * A2;
17
18 % Criando o filtro que irá formatar a informação
19 filtro_nrz = ones(1,N);
20
21 % Superamostrando a informação 1 e 2 em N
22 info_up1 = upsample(info_nrz1,N);
23 info_up2 = upsample(info_nrz2,N);
24
25 % Filtrando os sinais
26 sinal_tx1=filter(filtro_nrz,1,info_up1);
27 sinal_tx2=filter(filtro_nrz,1,info_up2);
```

Para simular a recepção do sinal, foi criado um loop com variação da SNR, de forma a obter a probabilidade de erro de bit. O código a seguir mostra como foi realizado a simulação da recepção do sinal transmitido.

```

28 % Verificação de erro em função da SNR
29 for SNR = 0 : SNR_max
30     % Adiciona ruído branco gaussiano no sinal com dada SNR
31     sinal_rx1 = awgn(sinal_tx1,SNR);
32     sinal_rx2 = awgn(sinal_tx2,SNR);
33
34     % Amostra o sinal para cada pulso iniciar no centro de cada bit
35     Z_T1 = sinal_rx1(N/2:N:end);
36     Z_T2 = sinal_rx2(N/2:N:end);
37
38     % Compara o sinal com limiar para transforma-lo em binario
39     info_hat1 = (Z_T1 > limiar1);
40     info_hat2 = (Z_T2 > limiar2);
41
42     % Probabilidade de erro de bit do sinal Rx
43     Pb1(SNR+1) = sum(xor(info_nrz1,info_hat1))/length(info_hat1);
44     Pb2(SNR+1) = sum(xor(info_nrz2,info_hat2))/length(info_hat2);
45 end

```

A Figura 3 mostra o sinal transmitido com diferentes amplitudes e seus respectivos sinais recebidos.

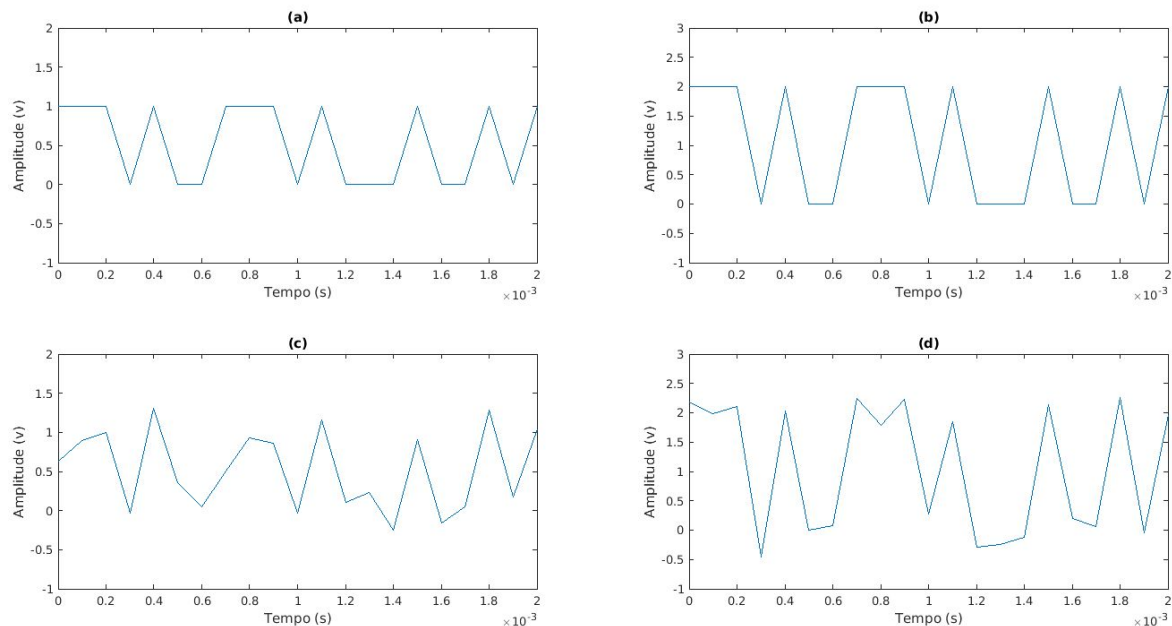


Figura 3 Informação gerada com (a) amplitude de 1V, (b) amplitude de 2V. Informação recebida com SNR=15 em (c) amplitude de 1V e (d) amplitude de 2V.

Por fim, a Figura 4 mostra o gráfico de probabilidade de erro de bit versus SNR, podemos observar que a amplitude de 2V mostrou um melhor desempenho em relação a de 1V pois a curva de queda da probabilidade possui um decaimento mais rápido em relação à variação da SNR.

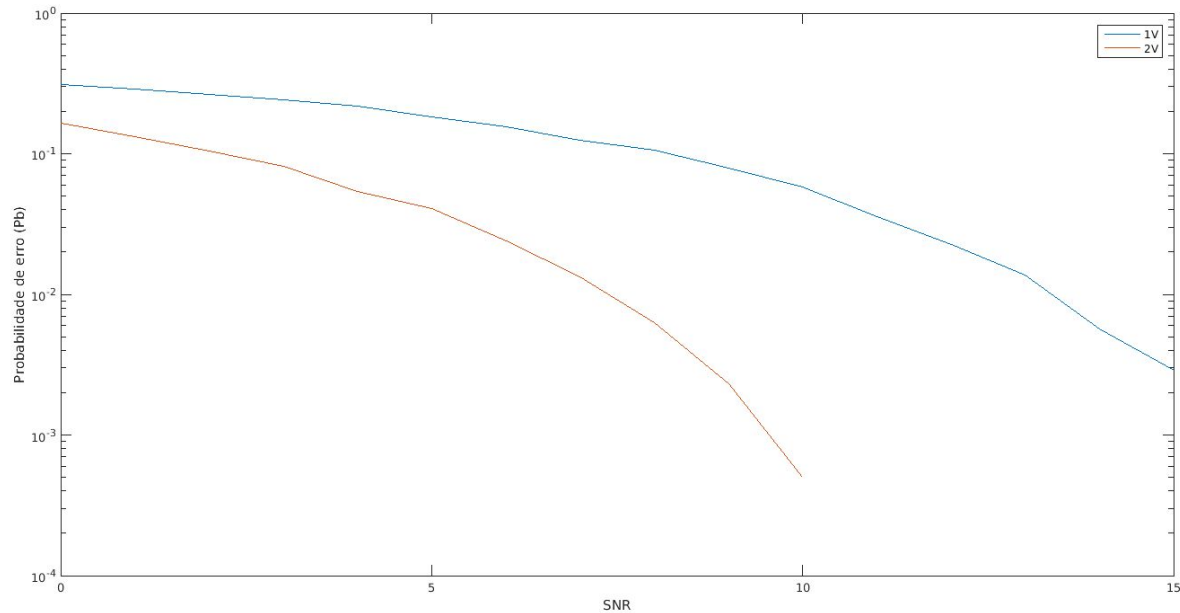


Figura 4 SNR x Pb do cenário 1

3.2.2. Cenário 2

O segundo cenário consistiu em simular uma sinalização NRZ unipolar com amplitude de 1V, verificando a diferença entre os sinais recebidos com e sem filtro casado. Abaixo o código referente à definição dos dados.

```

46 % Dados
47 N = 10;      % Fator de superamostragem
48 SNR_max=15;  % Razao Sinal-Ruido
49 Rb = 10000;
50 A = 1;
51 limiar = A/2;
52 Fs = Rb;
53 t = [0:1/Fs:1-1/Fs];

```

As etapas são semelhantes ao cenário 1, porém foi necessário gerar, superamostrar e filtrar apenas uma vez. A seguir o código referente à esta etapa.

```

54 info= randint(1,Rb);           % Gerando a informação
55 info_nrz1 = info * A;          % Gerando os níveis do sinal;
56 filtro_nrz = ones(1,N);        % Criando o filtro que irá formatar a informação
57 info_up1 = upsample(info_nrz1,N); % Superamostrando a informação em Nsamp
58
59 sinal_tx1=filter(filtro_nrz,1,info_up1); % Filtrando o sinal
60
61 filtro_rx = flip(filtro_nrz);  % Filtro do receptor

```

Para simular a recepção do sinal, foi criado um loop com variação da SNR, de forma a obter a probabilidade de erro de bit. Foram criadas duas informações recebidas: com e sem filtro casado no receptor. O código a seguir mostra como foi realizado a simulação da recepção do sinal transmitido.

```

62 % Verificação de erro de bit em função da SNR
63 for SNR = 0 : SNR_max
64     % Adiciona ruído branco gaussiano no sinal com dada SNR
65     sinal_rx1 = awgn(sinal_tx1,SNR);
66     sinal_rx2 = filter(filtro_rx,1,sinal_rx1)/N;
67
68     % Amostra o sinal para cada pulso iniciar no centro de cada bit
69     Z_T1 = sinal_rx1(N/2:N:end);
70     Z_T2 = sinal_rx2(N:N:end);
71
72     % Compara o sinal com limiar para transforma-lo em binário
73     info_hat1 = (Z_T1 > limiar);
74     info_hat2 = (Z_T2 > limiar);
75
76     % Probabilidade de erro dos sinais Rx
77     Pb1(SNR+1) = sum(xor(info_nrz1,info_hat1))/length(info_hat1);
78     Pb2(SNR+1) = sum(xor(info_nrz1,info_hat2))/length(info_hat2);
79 end

```

A Figura 5 mostra o sinal transmitido e seus respectivos sinais recebidos sem e com filtro casado. Note que a baixa variação do sinal se deve ao fato de o plot gerado ter sido realizado com a SNR máxima do loop.

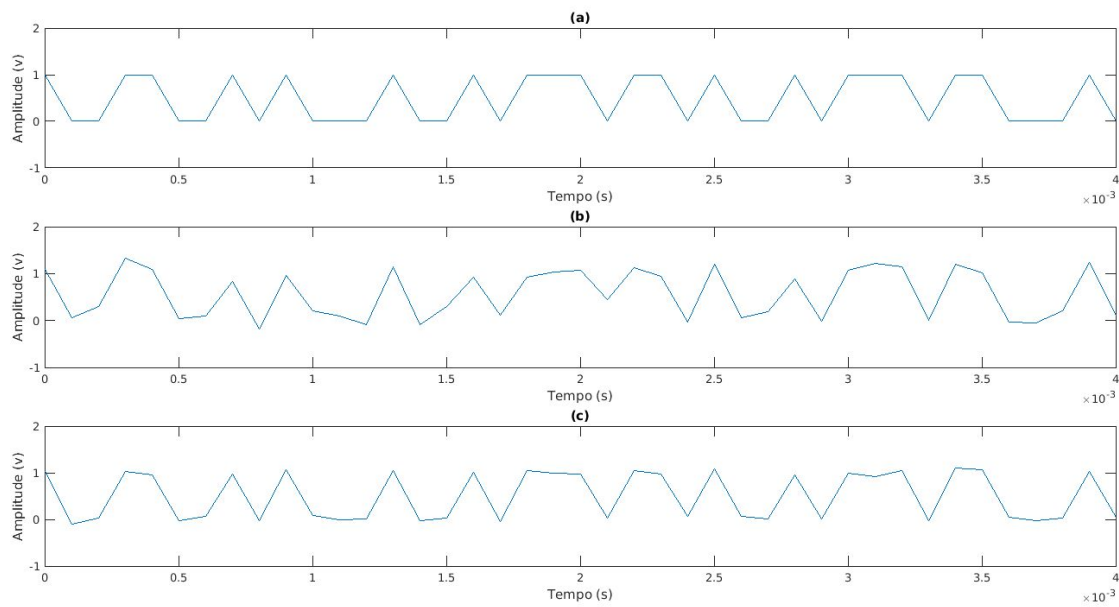


Figura 5 (a) sinal transmitido, (b) sinal recebido sem filtro casado e (c) sinal recebido com o filtro casado.

Por fim, a Figura 6 mostra o gráfico de probabilidade de erro de bit versus SNR. podemos observar que o sinal recebido com o filtro casado mostrou um melhor desempenho em relação ao sinal recebido sem filtro casado, pois a curva de queda da probabilidade possui um decaimento mais rápido em relação à variação da SNR.

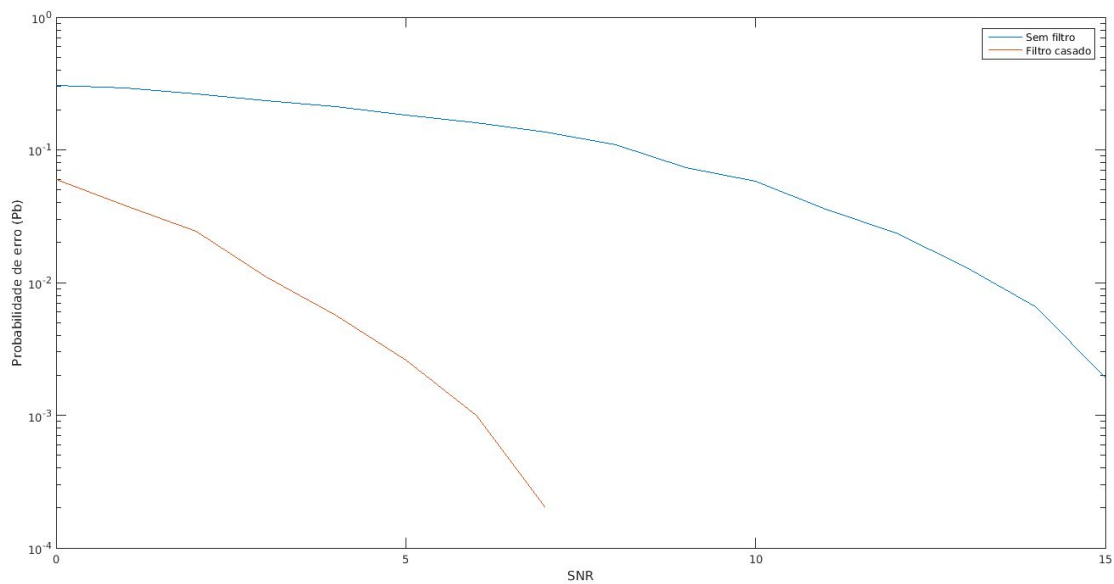


Figura 6 SNR x Pb do cenário 2

3.2.3. Cenário 3

No terceiro cenário, foi realizada a simulação de uma transmissão utilizando a sinalização NRZ unipolar e bipolar, ambas com filtro casado. Abaixo o código referente aos dados iniciais do sistema.

```
80 % Dados
81 N = 1;          % Fator de superamostragem
82 SNR_max=15;    % Razao Sinal-Ruido
83 Rb = 100000;
84 A = 1;
85 limiar = A/2;
86 Fs = Rb;
87 t = [0:1/Fs:1-1/Fs];
```

As etapas são semelhantes ao cenário 1 e 2, porém foi necessário gerar os sinais unipolar e bipolar, superamostrar e filtrar separadamente. A seguir o código referente à esta etapa.

```
88 % Gerando a informação
89 info= randint(1,Rb);
90 info_nrz_uni = info * A;
91 info_nrz_bi = (info*2)-1;
92
93 % Criando o filtro que irá formatar a informação
94 filtro_nrz = ones(1,N);
95
96 % Superamostrando a informação em N
97 info_up_uni = upsample(info_nrz_uni,N);
98 info_up_bi = upsample(info_nrz_bi,N);
99
100 % Filtrando o sinal
101 sinal_tx_uni=filter(filtro_nrz,1,info_up_uni);
102 sinal_tx_bi=filter(filtro_nrz,1,info_up_bi);
103
104 % Filtro do receptor
105 filtro_rx = filtro_nrz;
```

Para simular a recepção do sinal, foi criado um loop com variação da SNR, de forma a obter a probabilidade de erro de bit. Foram criadas duas informações recebidas: sinal unipolar e bipolar. O código a seguir mostra como foi realizado a simulação da recepção do sinal transmitido.

```

106 % Verificação de erro em função da SNR
107 for SNR = 0 : SNR_max
108     % Adiciona ruído branco gaussiano no sinal com dada SNR
109     sinal_rx_uni = awgn(sinal_tx_uni,SNR);
110     sinal_rx_bi = awgn(sinal_tx_bi,SNR);
111
112     % Adiciona ruído branco gaussiano no sinal com dada SNR
113     sinal_rx_uni_filt = filter(filtro_rx,1,sinal_rx_uni)/N;
114     sinal_rx_bi_filt = filter(filtro_rx,1,sinal_rx_bi)/N;
115
116     % Amostra o sinal para cada pulso iniciar no fim de cada bit
117     Z_T_uni = sinal_rx_uni_filt(N:N:end);
118     Z_T_bi = sinal_rx_bi_filt(N:N:end);
119
120     % Compara o sinal com limiar para transforma-lo em binário
121     info_hat_uni = (Z_T_uni > limiar);
122     info_hat_bi = (Z_T_bi > limiar);
123
124     % Probabilidade de erro do sinal Rx
125     Pb1(SNR+1) = sum(xor(info_nrz_uni,info_hat_uni))/length(info_hat_uni);
126     Pb2(SNR+1) = sum(xor(info_nrz_uni,info_hat_bi))/length(info_hat_bi);
127 end

```

A Figura 7 mostra o sinal transmitido e seus respectivos sinais recebidos no tipo NRZ unipolar e bipolar. Note que a baixa variação do sinal se deve ao fato de o plot gerado ter sido realizado com a SNR máxima do loop.

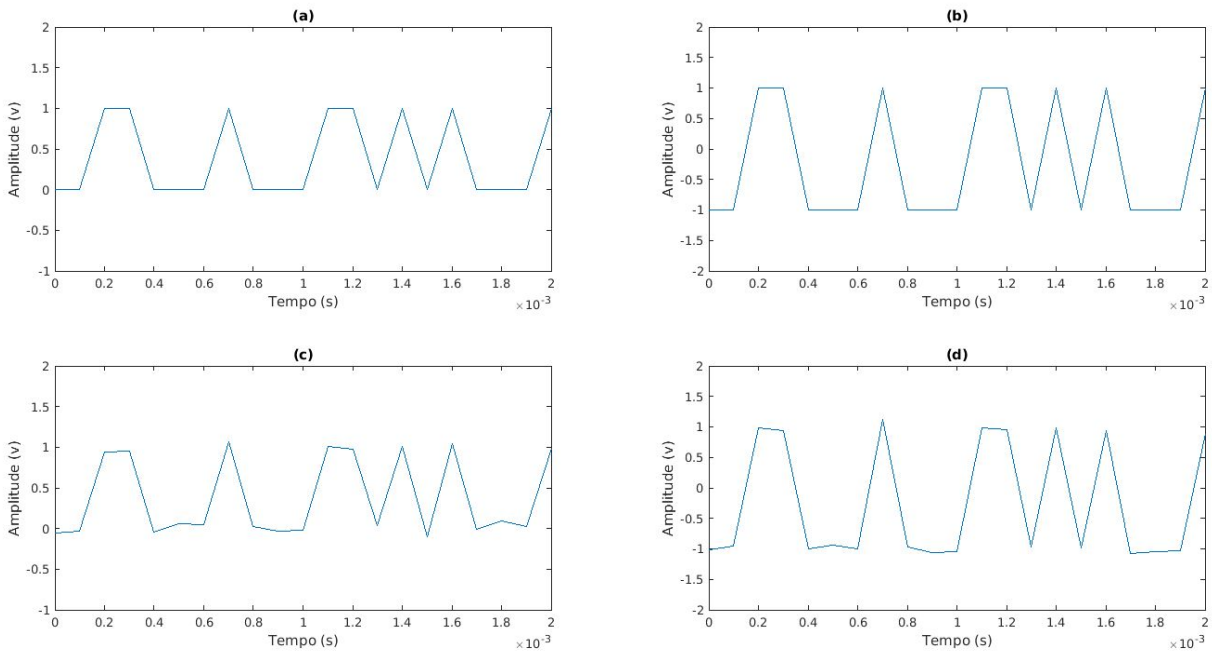


Figura 7 (a) sinal transmitido unipolar, (b) sinal transmitido bipolar (c) sinal recebido unipolar e (d) sinal recebido bipolar, ambos com SNR=15.

Por fim, a Figura 8 mostra o gráfico de probabilidade de erro de bit versus SNR. podemos observar que o sinal recebido NRZ bipolar mostrou um melhor desempenho em relação ao sinal recebido NRZ unipolar, apesar da curva de queda da probabilidade possui um decaimento tão lento quanto, a taxa de erro de bit é menor na mesma proporção.

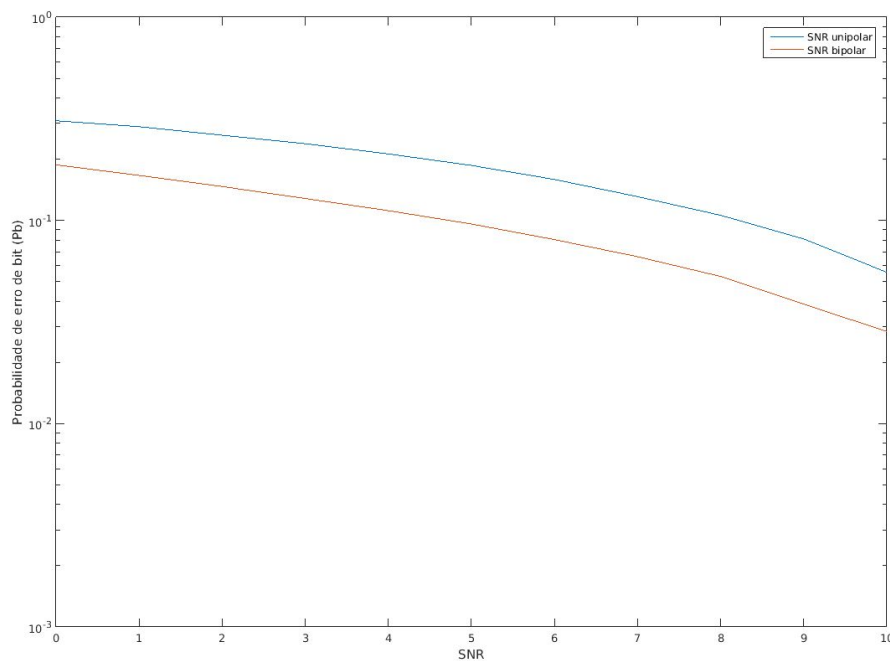


Figura 8 SNR x Pb do cenário 3

3.2.4. Cenário 4

No último cenário, foi realizada a mesma simulação que o cenário 3, porém foi realizada a variação de E_b/N_0 ao invés de variar a SNR, para que se possa comparar os valores da simulação teórica e prática. Não é necessário colocar o código de parâmetros iniciais e geração de informação e filtros, pois estes são iguais aos do cenário 3, sendo a única diferença o loop que simula a recepção do sinal. A probabilidade de erro teórica foi calculada utilizando a função Q, citada na seção 2.5, onde E_b/N_0 foi calculado em seu valor linear e não em dB. Já a probabilidade de erro prática foi calculada normalmente como nos outros cenários. A seguir, o código desenvolvido.

A Figura 9 mostra um comparativo da simulação utilizando SNR e utilizando as equações 3.73 e 3.76 do livro em [1]. Os valores teóricos (E_b/N_0) mostraram um melhor

desempenho do que os práticos (SNR), além de a bipolar ter tido um bom desempenho em ambas.

```

172 % Verificação de erro em função da SNR
173 for EBN0 = 0 : Eb_N0
174     %Pratica
175     sinal_rx_uni = awgn(sinal_tx_uni,EBN0);
176     sinal_rx_bi = awgn(sinal_tx_bi,EBN0);
177
178     sinal_rx_uni_filt = filter(filtro_rx,1,sinal_rx_uni)/N;
179     sinal_rx_bi_filt = filter(filtro_rx,1,sinal_rx_bi)/N;
180
181     % Amostra o sinal para cada pulso iniciar no fim de cada bit
182     Z_T_uni = sinal_rx_uni_filt(N:N:end);
183     Z_T_bi = sinal_rx_bi_filt(N:N:end);
184
185     % Compara o sinal com limiar para transforma-lo em binario
186     info_hat_uni = (Z_T_uni > limiar);
187     info_hat_bi = (Z_T_bi > limiar);
188
189     % Probabilidade de erro do sinal Rx
190     Pb1(EBN0+1) = qfunc(sqrt(10^(EBN0/10))); %unipolar
191     Pb2(EBN0+1) = qfunc(sqrt(2*10^(EBN0/10))); %bipolar
192
193     %SNR (pratica)
194     Pb12(EBN0+1) = sum(xor(info_nrz_uni,info_hat_uni2))/length(info_hat_uni2);
195     Pb22(EBN0+1) = sum(xor(info_nrz_uni,info_hat_bi2))/length(info_hat_bi2);
196 end

```

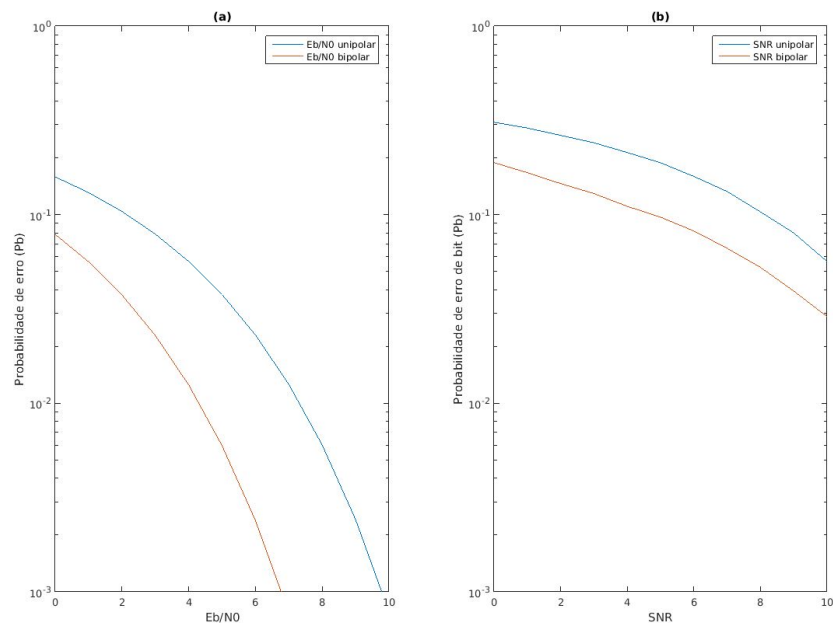


Figura 9 SNR x Pb do cenário 4. (a) simulação teórica e (b) simulação prática.

4. Conclusão

Este trabalho teve como objetivo realizar a simulação de diversas transmissões binárias, de forma a verificar o desempenho e analisar transmissões do tipo NRZ com variações de tipo (unipolar e bipolar) e com e sem a utilização de filtro casado no receptor. Além disso, foi realizado o cálculo de desempenho de erro no receptor através da variação da SNR no canal.

No primeiro experimento o foco foi em realizar a transmissão do sinal correta e verificar a recepção com e sem filtro casado. O filtro casado obteve um melhor desempenho pois removeu o ruído e com isso os possíveis erros de detecção de limiar.

No segundo experimento, foi possível verificar os diferentes cenários possíveis para uma transmissão binária e quais os tipos que apresentam uma menor probabilidade de erro de bit. Na primeira transmissão a amplitude maior obteve um melhor desempenho pois a energia média do sinal é melhor e isso reduziu a P_b consideravelmente. Na segunda transmissão o sinal com filtro casado teve um melhor desempenho, até mesmo que o da transmissão com amplitude maior do cenário 1, pois o filtro casado fez com que o sinal ficasse mais limpo, com um patamar melhor definido e próximo ao sinal original transmitido. Na terceira transmissão foi realizada a transmissão com filtro casado de sinais unipolar e bipolar. Este foi o melhor cenário entre os 3 realizados nesta questão pois além de haver um filtro casado, o sinal bipolar obteve um melhor desempenho, já que o limiar de decisão era maior entre os dois e a energia necessária para transmitir este dado é menor do que uma sinalização unipolar.

5. Referências

[1] SKLAR, B. Digital Communications: Fundamentals and Applications. 2. edition. Prentice Hall. 1029 p. ISBN 0-13-084788-7.

[2] HAYKIN, S. Sistemas de comunicação, 5ª edição. Bookman. 2010. 512 p.