

# Aula 3

ANA GONÇALVES

Universidade Federal de Minas Gerais

## I. TUTORIAL 3 - LETRA A

No ambiente MATLAB, esta equação diferencial ordinária de 1a. ordem pode ser resolvida usando variáveis simbólicas com a função `dsolve`. O script abaixo obtém a solução para um degrau unitário de entrada e permite variar os valores de  $R$  e  $C$ . O degrau unitário representa, por exemplo, uma bateria de 1 volt ligada à entrada em  $t = 0$ . Valores numéricos da saída são obtidos com a função `subs`.

Pode-se escrever a equação como:

$$x(t) = R * i(t) + y(t) \longrightarrow x(t) = RC \frac{dy(t)}{dt} + y(t) \quad (1)$$

A resposta dele ao degrau unitário,  $x(t) = u(t)$  é

$$y(t) = 1 * (1 - e^{t/RC}) \quad (2)$$

, com  $t \geq 0$  e  $RC$  é a constante de tempo do circuito

```

1 %===== Parte 1 =====
2 % equacao diferencial/resposta ao degrau
3
4 clear          % limpa workspace
5 clc           % limpa memoria da janela de comando
6 close all     % fecha figuras abertas
7
8 R = 1e3;      % resistencia, ohm
9 C = 100e-9;   % capacitancia, farad, RC = 100 microsegundos
10 T = 1e-6;    % intervalo de discretizacao para plotar graficos
11
12 syms y(t)    % define a variavel simbolica
13 Dy = diff(y); % D: dy(t)/dt
14
15 % resolve a equacao com entrada x(t) = 1 (degrau) e cond iniciais nulas
16 y = dsolve(1 == R*C*Dy + y, y(0)==0);
17
18 t1 = 0: T: 5*R*C; % t1: vetor com coordenadas temporais
19 y1 = subs(y, t, t1); % y1: resposta calculada nos instantes em t1
20
21 figure(1);
22 subplot(2,2,1); % figure(1): formato 2x2, 4 paines
23 plot(t1,y1);
24 xlabel('tempo (s)');
25 ylabel('tensao saida (V)');
26 title('Resposta ao degrau');
27 grid

```

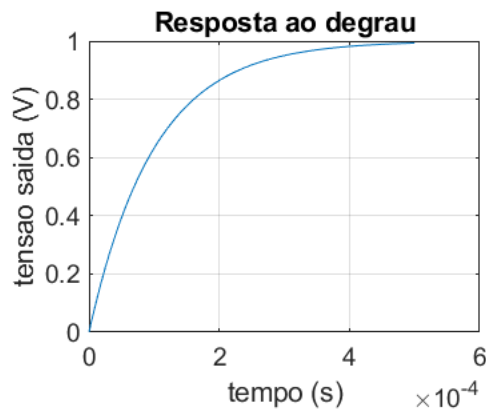


Figura 1: Gráfico gerado

## II. TUTORIAL 3 - LETRA B

Na letra b, muda-se apenas a linha 18 para  $1^{\circ}R^{\circ}C$  e depois para  $10^{\circ}R^{\circ}C$ , os resultados estão disponibilizados nas imagens abaixo.

A constante de tempo interfere diretamente na velocidade de resposta do sistema de primeira ordem e de forma inversamente proporcional. Quanto menor a constante de tempo maior é a velocidade de resposta do sistema pois o mesmo atinge a resposta de regime permanente num intervalo de tempo menor (e vice versa).

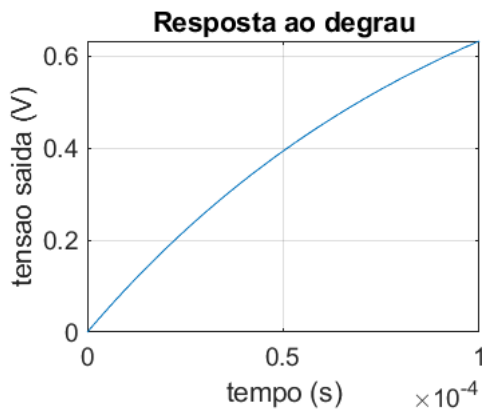


Figura 2: Gráfico gerado

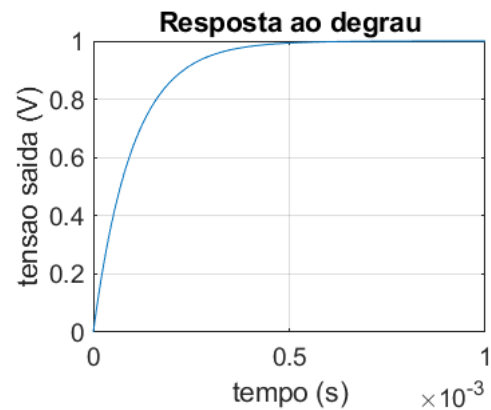


Figura 3: Gráfico gerado

## III. TUTORIAL 3 - LETRA C

A resposta impulsiva  $h(t) = \frac{1}{RC}e^{t/RC}$ ,  $t \geq 0$ , pode ser obtida derivando a resposta ao degrau. Com a resposta impulsiva, por sua vez, pode-se obter a resposta a uma entrada qualquer por meio da convolução. Acrescenta-se o código abaixo no código da letra a.

```
1 % resposta impulsiva
2 subplot(2,2,2);
3 h = diff(y);           %h: resposta impulsiva h(t)
4 h1 = subs(h,t,t1);     %h1: resposta impulsiva calculada em t1
5 plot(t1, h1);
6 xlabel('tempo (s)');
7 ylabel('tensão saída (V)');
```

```

8 title('Resposta impulsiva');
9 grid

```



Figura 4: Gráfico gerado

#### IV. TUTORIAL 3 - LETRA D

O script baseia-se numa versão aproximada da integral de convolução.

$$s(t)|_{t=nT} = s(nT) \approx \sum_{k=-\infty}^{\infty} x[n] * h[n-k] * T \quad (3)$$

```

1 %===== Parte 1 =====
2 % equacao diferencial/resposta ao degrau
3
4 clear          % limpa workspace
5 clc           % limpa memoria da janela de comando
6 close all     % fecha figuras abertas
7
8 R = 1e3;      % resistencia, ohm
9 C = 100e-9;   % capacitancia, farad, RC = 100 microsegundos
10 T = 1e-6;    % intervalo de discretizacao para plotar graficos
11
12 syms y(t)    % define a variavel simbolica
13 Dy = diff(y); % D: dy(t)/dt
14
15 % resolve a equacao com entrada x(t) = 1 (degrau) e cond iniciais nulas
16 y = dsolve(1 == R*C*Dy + y, y(0)==0);
17
18 t1 = 0: T: 5*R*C; % t1: vetor com coordenadas temporais
19 y1 = subs(y, t, t1); % y1: resposta calculada nos instantes em t1
20
21 figure(1);
22 subplot(2,2,1); % figure(1): formato 2x2, 4 paines
23 plot(t1,y1);
24 xlabel('tempo (s)');
25 ylabel('tensao saida (V)');
26 title('Resposta ao degrau');
27 grid
28
29 % resposta impulsiva
30 subplot(2,2,2);

```

```

31 h = diff(y);           %h: resposta impulsiva h(t)
32 h1 = subs(h,t,t1);     %h1: resposta impulsiva calculada em t1
33 plot(t1, h1);
34 xlabel('tempo (s)');
35 ylabel('tensao saida (V)');
36 title('Resposta impulsiva');
37 grid
38
39 % resposta a um pulso retangular
40 Lu = 50;                % duracao do pulso
41 t2 = linspace(0, R*C, Lu); % coordenada para pulso com 100 amostras
42 u = ones(1, Lu);        % u: pulso com 100 amostras, formato double
43 h1d = double(h1);        % converte h1 para formato double
44 s = conv(u, h1d)*T;      % s: sauda por convolucao; u e h1d com mesmo formato
45 % multiplicacao por T: aproximacao da integral (ver roteiro)
46
47 Ls = length(s);         % Note: Ls = length(t2) + length(h1) - 1
48
49 t3 = linspace(0,Ls*T, Ls); % para uniformiza tamanho de subplots
50 subplot(2,2,3);
51 hold on;                % pulso de entrada e saida serao sobrepostos
52 plot(t3, [u zeros(1, Ls - Lu)]); % plota pulso de entrada
53
54 % subplot(2,2,4);
55 plot(t3,s);              % plota saida
56 hold off;

```

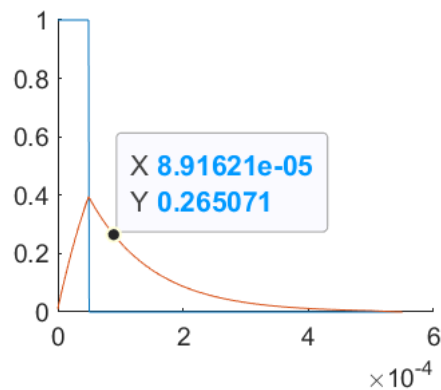


Figura 5: Gráfico gerado

## V. TUTORIAL 3 - LETRA E

Foi escolhido  $L = 20$  e  $L = 80$ .

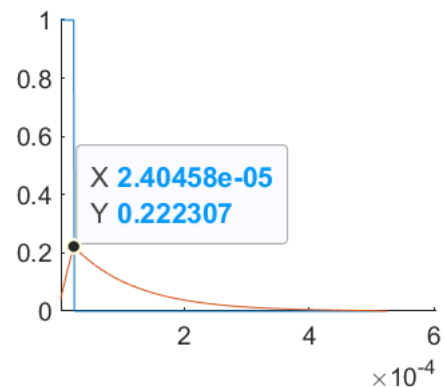


Figura 6: Gráfico gerado

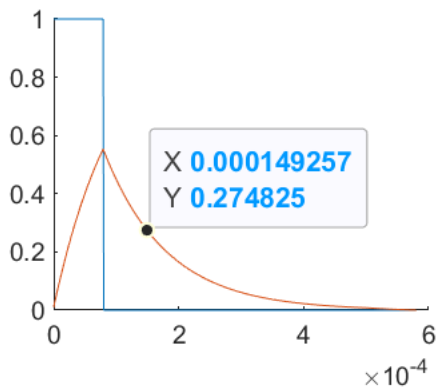


Figura 7: Gráfico gerado

## VI. TUTORIAL 3 - LETRA F

A função `gensig` cria sinais periódicos. Há outras funções para gerar sinais no MATLAB. O exemplo abaixo cria um pulso senoidal que é, em seguida, processado (convoluído) com a resposta impulsiva do circuito.

```

1  %===== Parte 1 =====
2  % equacao diferencial/resposta ao degrau
3
4  clear          % limpa workspace
5  clc            % limpa memoria da janela de comando
6  close all      % fecha figuras abertas
7
8  R = 1e3;        % resistencia, ohm
9  C = 100e-9;    % capacitancia, farad, RC = 100 microsegundos
10 T = 1e-6;      % intervalo de discretizacao para plotar graficos
11
12 syms y(t)      % define a variavel simbolica
13 Dy = diff(y); % D: dy(t)/dt
14
15 % resolve a equacao com entrada x(t) = 1 (degrau) e cond iniciais nulas
16 y = dsolve(1 == R*C*Dy + y, y(0)==0);
17
18 t1 = 0: T: 5*R*C; % t1: vetor com coordenadas temporais
19 y1 = subs(y, t, t1); % y1: resposta calculada nos instantes em t1
20
21 figure(1);
22 subplot(2,2,1); % figure(1): formato 2x2, 4 paines
23 plot(t1,y1);
24 xlabel('tempo (s)');
25 ylabel('tensao saida (V)');
26 title('Resposta ao degrau');
27 grid
28
29 % resposta impulsiva
30 subplot(2,2,2);
31 h = diff(y); %h: resposta impulsiva h(t)
32 h1 = subs(h,t,t1); %h1: resposta impulsiva calculada em t1
33 plot(t1, h1);
34 xlabel('tempo (s)');
35 ylabel('tensao saida (V)');
36 title('Resposta impulsiva');
37 grid
38
39 % resposta a um pulso retangular

```

```

40 Lu = 80; % duracao do pulso
41 t2 = linspace(0, R*C, Lu); % coordenada para pulso com 100 amostras
42 u = ones(1, Lu); % u: pulso com 100 amostras, formato double
43 h1d = double(h1); % converte h1 para formato double
44 s = conv(u, h1d)*T; % s: sauda por convolucao; u e h1d com mesmo formato
45 % multiplicacao por T: aproximacao da integral (ver roteiro)
46
47 Ls = length(s); % Note: Ls = length(t2) + length(h1) - 1
48
49 t3 = linspace(0,Ls*T, Ls); % para uniformiza tamanho de subplots
50 subplot(2,2,3);
51 hold on; % pulso de entrada e saida serao sobrepostos
52 plot(t3, [u zeros(1, Ls - Lu)]); % plota pulso de entrada
53
54 % subplot(2,2,4);
55 plot(t3,s); % plota saida
56 hold off;
57
58 %===== Parte 3 =====
59 % Resposta a um pulso senoidal
60 figure(2)
61
62 periodo = R*C; dur = 10*R*C;
63 [x, tx] = gensig('sin', periodo, dur, T);
64
65 s = conv(x,h1d)*T; %s: saida por convolucao; u e h1c com mesmo formato
66 Ls = length(s); % Note: Ls = length(t2) + length(h1) - 1
67 t3 = linspace(0,Ls*T, Ls);
68
69 plot(tx,x);
70 hold on
71 plot(t3,s);
72 grid;
73 legend('entrada', 'saida');
74 title('Sistema de 1a. ordem');
75 hold off

```

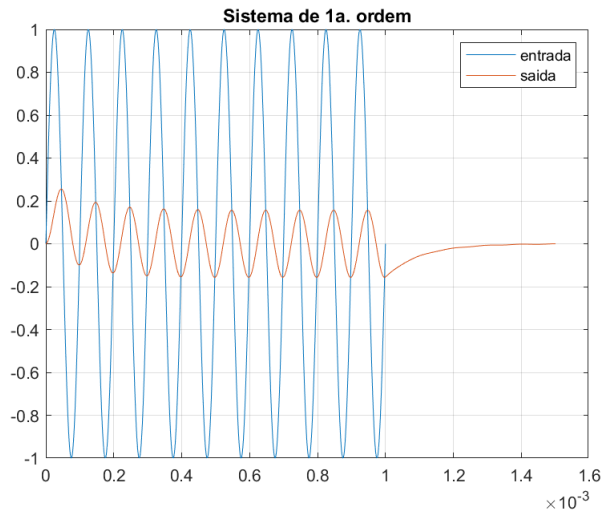


Figura 8: Gráfico gerado

## VII. TUTORIAL 3 - LETRA G

A resposta ao impulso leva a uma saída de comportamento senoidal mas que, após certo intervalo de tempo, decai em forma exponencial. Assim, quanto maior o valor do impulso, maior o tempo do comportamento em senoide até que o capacitor comece a descarregar.

## VIII. TUTORIAL 3 - LETRA H

Muda-se o período primeiramente para  $R \cdot C / 10$  e depois para  $10 \cdot R \cdot C$ .

A senóide eterna trata-se de uma aproximação idealizada, em vista de considerar todos os instantes de tempo ( $-\infty < t < \infty$ ). O modelo torna-se mais preciso, à medida que os tempos de observação sejam longos comparados com o seu período  $T = 2\pi/\omega_0$ .

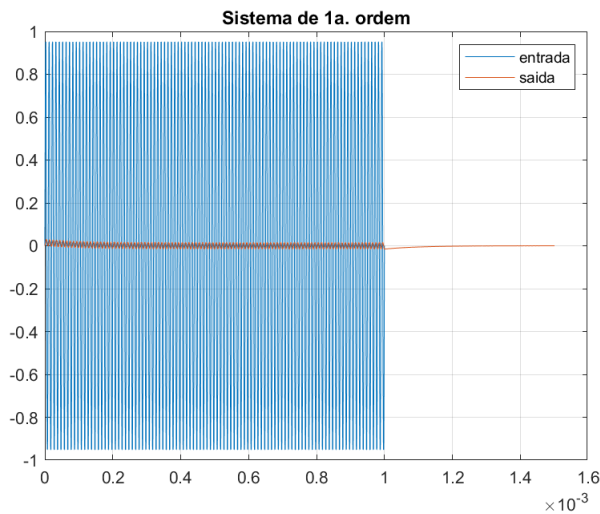


Figura 9: Gráfico gerado

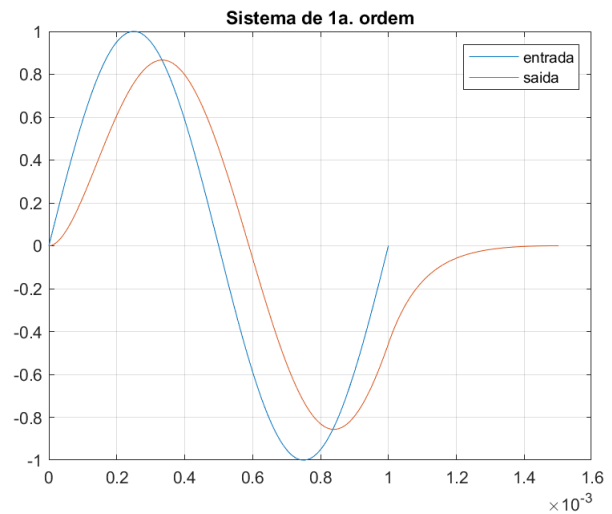


Figura 10: Gráfico gerado

## IX. TUTORIAL 3 - LETRA I

```

1 %===== Parte 1 =====
2 % equacao diferencial/resposta ao degrau
3
4 clear          % limpa workspace
5 clc           % limpa memoria da janela de comando
6 close all     % fecha figuras abertas
7
8 R = 1e3;      % resistencia, ohm
9 C = 100e-9;   % capacitancia, farad, RC = 100 microsegundos
10 T = 1e-6;    % intervalo de discretizacao para plotar graficos
11
12 syms y(t)    % define a variavel simbolica
13 Dy = diff(y); % D: dy(t)/dt
14
15 % resolve a equacao com entrada x(t) = 1 (degrau) e cond iniciais nulas
16 y = dsolve(1 == R*C*Dy + y, y(0)==0);
17
18 t1 = 0: T: 5*R*C; % t1: vetor com coordenadas temporais
19 y1 = subs(y, t, t1); % y1: resposta calculada nos instantes em t1
20

```

```

21 figure(1);
22 subplot(2,2,1);      % figure(1): formato 2x2, 4 paines
23 plot(t1,y1);
24 xlabel('tempo (s)');
25 ylabel('tensao saida (V)');
26 title('Resposta ao degrau');
27 grid
28
29 % resposta impulsiva
30 subplot(2,2,2);
31 h = diff(y);          %h: resposta impulsiva h(t)
32 h1 = subs(h,t,t1);    %h1: resposta impulsiva calculada em t1
33 plot(t1, h1);
34 xlabel('tempo (s)');
35 ylabel('tensao saida (V)');
36 title('Resposta impulsiva');
37 grid
38
39 % resposta a um pulso retangular
40 Lu = 80;              % duracao do pulso
41 t2 = linspace(0, R*C, Lu); % coordenada para pulso com 100 amostras
42 u = ones(1, Lu);      % u: pulso com 100 amostras, formato double
43 h1d = double(h1);     % converte h1 para formato double
44 s = conv(u, h1d)*T;    % s: sauda por convolucao; u e h1d com mesmo formato
45 % multiplicacao por T: aproximacao da integral (ver roteiro)
46
47 Ls = length(s);       % Note: Ls = length(t2) + length(h1) - 1
48
49 t3 = linspace(0,Ls*T, Ls); % para uniformiza tamanho de subplots
50 subplot(2,2,3);
51 hold on;              % pulso de entrada e saida serao sobrepostos
52 plot(t3, [u zeros(1, Ls - Lu)]); % plota pulso de entrada
53
54 % subplot(2,2,4);
55 plot(t3,s);           % plota saida
56 hold off;
57
58 %===== Parte 3 =====
59 % Resposta a um pulso senoidal
60 figure(2)
61
62 periodo = 10*R*C; dur = 10*R*C;
63 [x, tx] = gensig('sin', periodo, dur, T);
64
65 s = conv(x,h1d)*T;    %s: saida por convolucao; u e h1c com mesmo formato
66 Ls = length(s);      % Note: Ls = length(t2) + length(h1) - 1
67 t3 = linspace(0,Ls*T, Ls);
68
69 plot(tx,x);
70 hold on
71 plot(t3,s);
72 grid;
73 legend('entrada', 'saida');
74 title('Sistema de 1a. ordem');
75 hold off
76
77 % ===== Parte 4 =====
78 %  $H(s) = 1/(sRC + 1)$ 
79 Num = 1;
80 Den = [R*C 1];
81 H = tf(Num, Den); % veja H na linha de comando
82
83 figure(3)
84 subplot(2,2,1); step(H); grid;

```



```

85 subplot(2,2,2); impulse(H); grid;
86 subplot(2,2,3); pzmap(H);

```

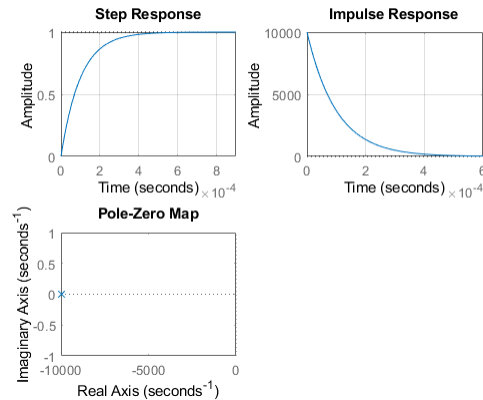


Figura 11: Gráfico gerado

Nota-se que o comportamento exponencial destas curvas valida os procedimentos realizados nos itens anteriores.

## X. TUTORIAL 3 - LETRA J

```

1  %===== Parte 1 =====
2  % equacao diferencial/resposta ao degrau
3
4  clear          % limpa workspace
5  clc           % limpa memoria da janela de comando
6  close all     % fecha figuras abertas
7
8  R = 1e3;      % resistencia, ohm
9  C = 100e-9;   % capacitancia, farad, RC = 100 microsegundos
10 T = 1e-6;     % intervalo de discretizacao para plotar graficos
11
12 syms y(t)     % define a variavel simbolica
13 Dy = diff(y); % D: dy(t)/dt
14
15 % resolve a equacao com entrada x(t) = 1 (degrau) e cond iniciais nulas
16 y = dsolve(1 == R*C*Dy + y, y(0)==0);
17
18 t1 = 0: T: 5*R*C; % t1: vetor com coordenadas temporais
19 y1 = subs(y, t, t1); % y1: resposta calculada nos instantes em t1
20
21 figure(1);
22 subplot(2,2,1); % figure(1): formato 2x2, 4 paines
23 plot(t1,y1);
24 xlabel('tempo (s)');
25 ylabel('tensao saida (V)');
26 title('Resposta ao degrau');
27 grid
28
29 % resposta impulsiva
30 subplot(2,2,2);
31 h = diff(y); %h: resposta impulsiva h(t)
32 h1 = subs(h,t,t1); %h1: resposta impulsiva calculada em t1
33 plot(t1, h1);

```

```

34 xlabel('tempo (s)');
35 ylabel('tensao saida (V)');
36 title('Resposta impulsiva');
37 grid
38
39 % resposta a um pulso retangular
40 Lu = 80; % duracao do pulso
41 t2 = linspace(0, R*C, Lu); % coordenada para pulso com 100 amostras
42 u = ones(1, Lu); % u: pulso com 100 amostras, formato double
43 h1d = double(h1); % converte h1 para formato double
44 s = conv(u, h1d)*T; % s: saida por convolucao; u e h1d com mesmo formato
45 % multiplicacao por T: aproximacao da integral (ver roteiro)
46
47 Ls = length(s); % Note: Ls = length(t2) + length(h1) - 1
48
49 t3 = linspace(0,Ls*T, Ls); % para uniformiza tamanho de subplots
50 subplot(2,2,3);
51 hold on; % pulso de entrada e saida serao sobrepostos
52 plot(t3, [u zeros(1, Ls - Lu)]); % plota pulso de entrada
53
54 % subplot(2,2,4);
55 plot(t3,s); % plota saida
56 hold off;
57
58 %===== Parte 3 =====
59 % Resposta a um pulso senoidal
60 figure(2)
61
62 periodo = 10*R*C; dur = 10*R*C;
63 [x, tx] = gensig('sin', periodo, dur, T);
64
65 s = conv(x,h1d)*T; %s: saida por convolucao; u e h1c com mesmo formato
66 Ls = length(s); % Note: Ls = length(t2) + length(h1) - 1
67 t3 = linspace(0,Ls*T, Ls);
68
69 plot(tx,x);
70 hold on
71 plot(t3,s);
72 grid;
73 legend('entrada', 'saida');
74 title('Sistema de 1a. ordem');
75 hold off
76
77 % ===== Parte 4 =====
78 %  $H(s) = 1/(sRC + 1)$ 
79 Num = 1;
80 Den = [R*C 1];
81 H = tf(Num, Den); % veja H na linha de comando
82
83 figure(3)
84 subplot(2,2,1); step(H); grid;
85 subplot(2,2,2); impulse(H); grid;
86 subplot(2,2,3); pzmap(H);
87
88 figure(4)
89 bode(H); grid;

```

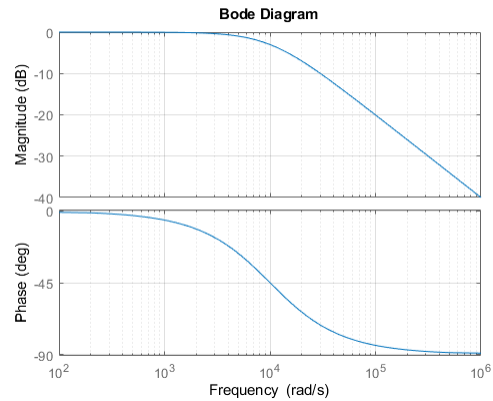


Figura 12: Gráfico gerado

Pelo diagrama de Bode, nota-se que a frequência de corte é de  $10^3$  rad/s. Assim, o valor de defasagem é de  $-45^\circ$  e o valor de magnitude é de  $-3\text{dB}$ , aproximadamente. Posteriormente, por meio do cálculo da função de transferência, encontrasse  $|H(j\omega)| = \frac{\sqrt{2}}{2}$

## XI. TUTORIAL 3 - LETRA K

```

1 % Sistema de 1a. ordem em tempo discreto
2 close all
3 clear
4 clc
5
6 alfa = 0.1;
7 y_ant = 0;      % memoriza saida anterior
8 N = 10000;      % horizonte de simulacao
9 f1 = 1/40       % freq de interesse
10 A1 = 1;         % amplitude do sinal de interesse
11 f2 = 1/5;       % freq do ruido a ser filtrado
12 A2 = 0.5;       % amplitude do ruido a ser filtrado
13 fs = 10e3       % freq de amostragem (para reproducao)
14 t = 0:N-1;
15
16 x = A1*cos(2*pi*t*f1) + A2*sin(2*pi*t*f2); % entrada
17 y = linspace(1,N); % saida
18
19 % implementacao equacao a diferencas
20 for i=1:N
21     y(i) = alfa*x(i) + (1-alfa)*y_ant;
22     y_ant = y(i);
23 end
24
25 figure(1);
26 subplot(2,1,1);
27 hold on
28 plot(x); plot(y);
29 xlabel('n');
30 legend('x[n]', 'y[n]'); title('Equacao a diferencas finitas');
31 hold off

```

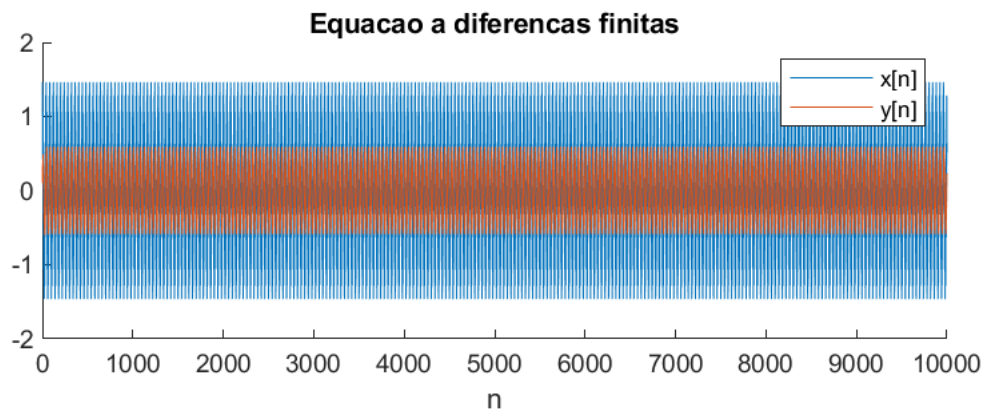


Figura 13: Gráfico gerado

Com  $N = 1000$

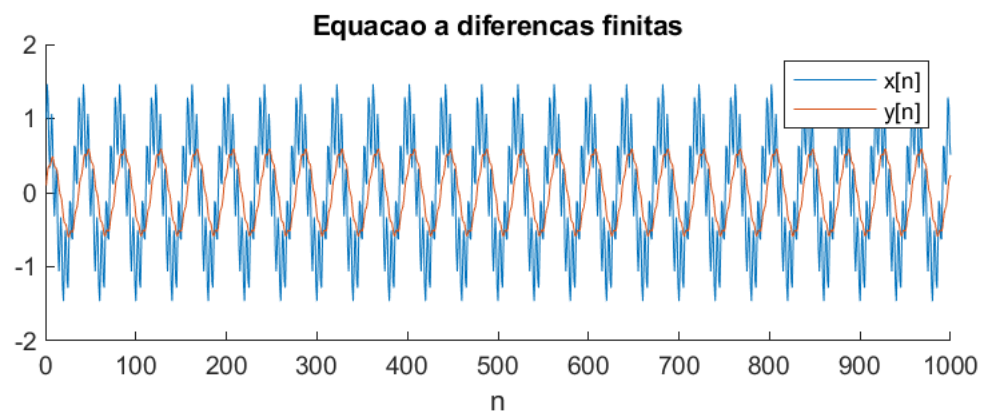


Figura 14: Gráfico gerado

## XII. TUTORIAL 3 - LETRA L

Usando  $f_2 = 1/500$

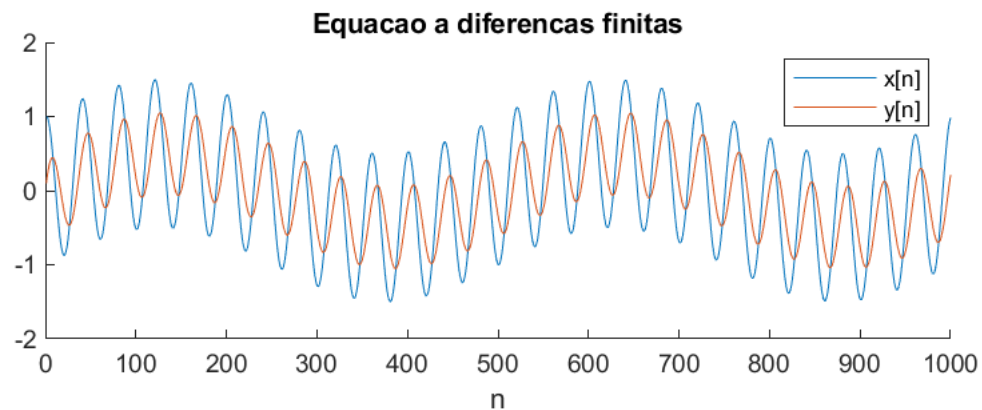


Figura 15: Gráfico gerado

Usando  $A_2 = 0.75$

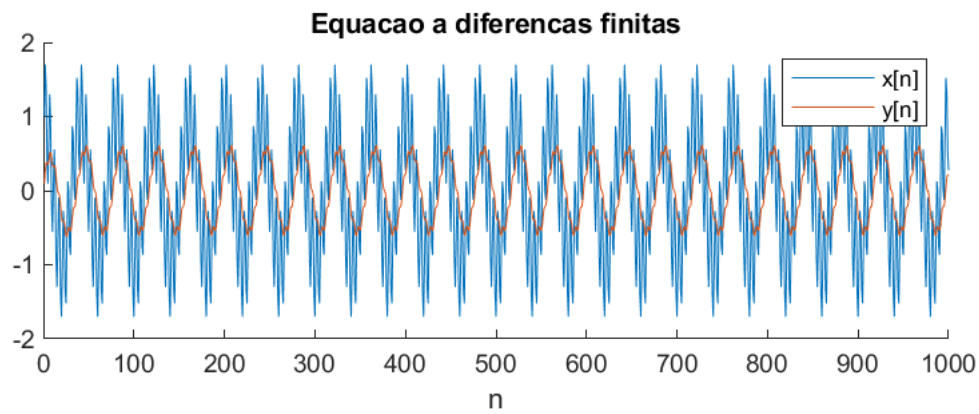


Figura 16: Gráfico gerado

Usando  $\alpha = 1$

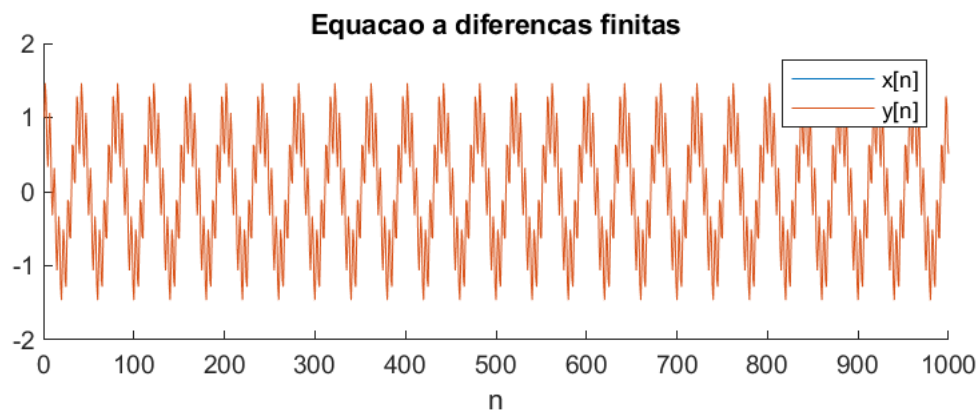


Figura 17: Gráfico gerado

Se  $\alpha > 0$ , o sinal é exponencial crescente; Se  $\alpha < 0$ , o sinal é exponencial decrescente;

### XIII. TUTORIAL 3 - LETRA M

```

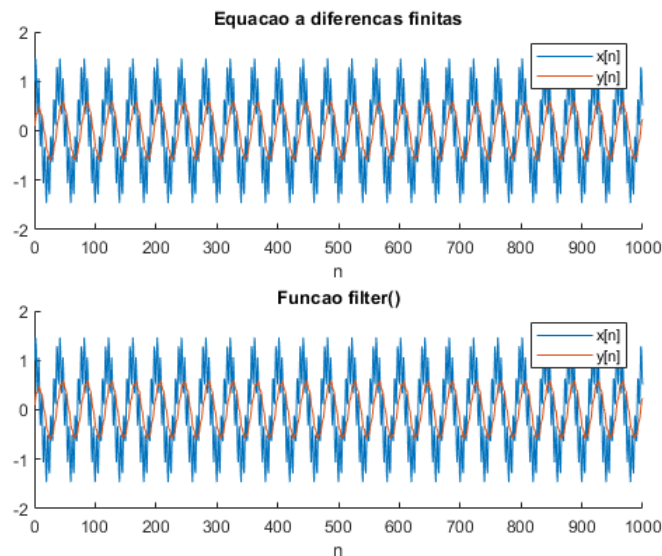
1 % Sistema de 1a. ordem em tempo discreto
2 close all
3 clear
4 clc
5
6 alfa = 0.1;
7 y_ant = 0; % memoriza saida anterior
8 N = 1000; % horizonte de simulacao
9 f1 = 1/40; % freq de interesse
10 A1 = 1; % amplitude do sinal de interesse
11 f2 = 1/5; % freq do ruido a ser filtrado
12 A2 = 0.5; % amplitude do ruido a ser filtrado
13 fs = 10e3; % freq de amostragem (para reproducao)
14 t = 0:N-1;
15
16 x = A1*cos(2*pi*t*f1) + A2*sin(2*pi*t*f2); % entrada
17 y = linspace(1,N); % saida

```

```

18
19 % implementacao equacao a diferencas
20 for i=1:N
21     y(i)= alfa*x(i)+(1-alfa)*y_ant;
22     y_ant = y(i);
23 end
24
25 figure(1);
26 subplot(2,1,1);
27 hold on
28 plot(x); plot(y);
29 xlabel('n');
30 legend('x[n]', 'y[n]'); title('Equacao a diferencas finitas');
31 hold off
32
33 % usa funco filter; consulte o help
34 a = 1;
35 b = [alfa (1-alfa)];
36 y2 = filter(b,a,x);
37 subplot(2,1,2);
38
39 hold on
40 plot(x); plot(y);
41 xlabel('n');
42 legend('x[n]', 'y[n]'); title('Funcao filter()');
43 hold off

```



**Figura 18:** Gráfico gerado

#### XIV. TUTORIAL 3 - LETRA N

```

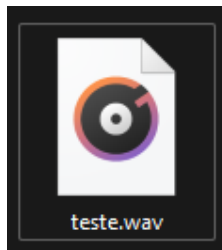
1 % Sistema de 1a. ordem em tempo discreto
2 close all
3 clear
4 clc
5

```

```

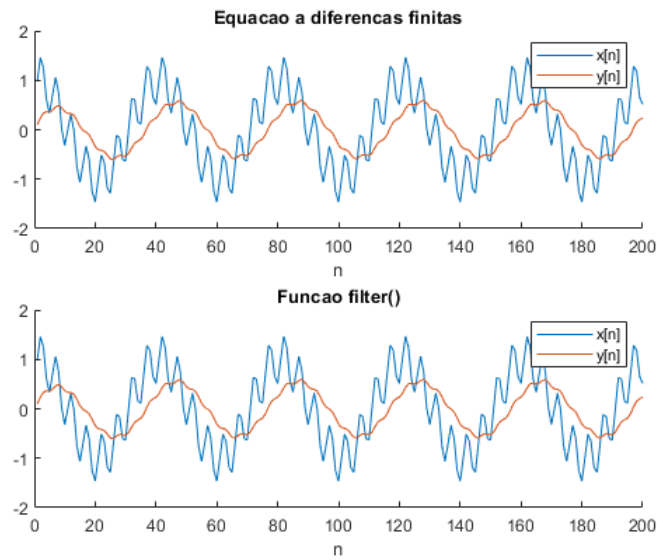
6  alfa = 0.1;
7  y_ant = 0;      % memoriza saida anterior
8  N = 200;        % horizonte de simulacao
9  f1 = 1/40;      % freq de interesse
10 A1 = 1;         % amplitude do sinal de interesse
11 f2 = 1/5;       % freq do ruido a ser filtrado
12 A2 = 0.5;       % amplitude do ruido a ser filtrado
13 fs = 10e3;      % freq de amostragem (para reproducao)
14 t = 0:N-1;
15
16 x = A1*cos(2*pi*t*f1) + A2*sin(2*pi*t*f2); % entrada
17 y = linspace(1,N); % saida
18
19 % implementacao equacao a diferencas
20 for i=1:N
21     y(i)= alfa*x(i)+(1-alfa)*y_ant;
22     y_ant = y(i);
23 end
24
25 figure(1);
26 subplot(2,1,1);
27 hold on
28 plot(x); plot(y);
29 xlabel('n');
30 legend('x[n]', 'y[n]'); title('Equacao a diferencas finitas');
31 hold off
32
33 % usa funco filter; consulte o help
34 a = 1;
35 b = [alfa (1-alfa)];
36 y2 = filter(b,a,x);
37 subplot(2,1,2);
38
39 hold on
40 plot(x); plot(y);
41 xlabel('n');
42 legend('x[n]', 'y[n]'); title('Funcao filter()');
43 hold off
44
45 sound(x,fs);      % reproduz entrada x na frq amostragem fs
46 pause;           % aguarda qq tecla na linha de comando
47 sound(y,fs);      % reproduz saida filtrada y
48
49 % gera um arquivo de saida; nao substitui o arquivo existente
50 audiowrite('teste.wav', y, fs, 'BitsPerSample',16);
51 info = audiointo('teste.wav');

```



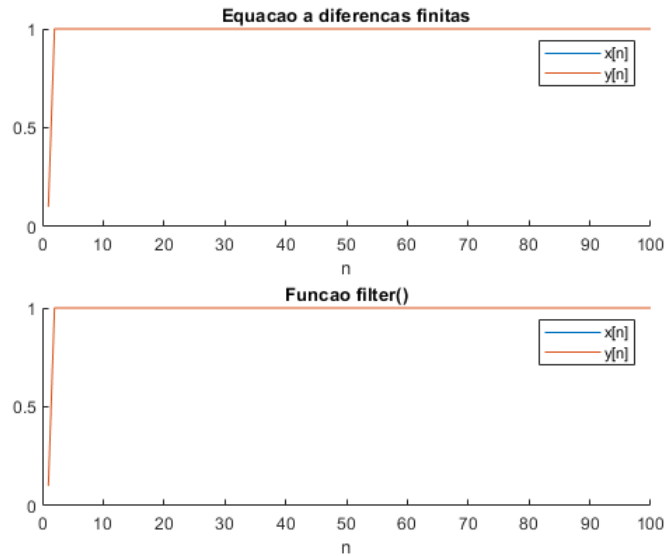
**Figura 19:** Arquivo teste.wav no PC

Utilizando  $N = 200$



**Figura 20:** *Gráfico gerado*

Utilizando  $N = 1$



**Figura 21:** *Gráfico gerado*

## XV. BIBLIOGRAFIA

[https://edisciplinas.usp.br/pluginfile.php/3560104/mod\\_resource/content/0/sistemas\\_primeira\\_ordem.pdf](https://edisciplinas.usp.br/pluginfile.php/3560104/mod_resource/content/0/sistemas_primeira_ordem.pdf)
[http://www.univasf.edu.br/~www.professores.uff.br/ninoska/wp-content/uploads/sites/57/2017/08/Control\\_Aula20\\_RespostaFrequencia.pdf](http://www.univasf.edu.br/~www.professores.uff.br/ninoska/wp-content/uploads/sites/57/2017/08/Control_Aula20_RespostaFrequencia.pdf)
<https://statics-shoptime.b2w.io/sherlock/books/firstChapter/112168262.pdf>