INSTITUTO FEDERAL DO ESPIRITO SANTO

Curso de Engenharia Elétrica

ANÁLISE DE SINAIS E SISTEMAS – 2023/1 – G1

PRÁTICA 1 – RESPOSTA DE UM SISTEMA E CONVOLUÇÃO

Etapa 1 – Uso do Comando <INLINE>.

a) Criando uma Função

```
f = inline(' exp(-0.5 * t). * sin(2 * pi * t)','t')
```

%b) Calculando o valor pontual f(2)

```
%Forma 1
t = 1.75;
f(t)
% Forma 2
f(1.75)
```

c) Acessando a faixa de valores [-1.75:1.75];

```
t = (-1.75 : 1.75);
f(t)
```

d) Plotando uma função <INLINE> do vetor de tempo

d.1) Vetor de valores inteiros com baixa resolução

```
t = (-1.75 : 1.75);
figure(1)
plot(t, f(t), 'b-');
xlabel(' t');
ylabel(' f(t)');
grid on
```

Observe que o gráfico, de valores inteiros, está com baixa resolução então, não expressa o real formato da função.

d.2) Vetor de valores inteiros com alta resolução

```
t = (-1.75 : 0.01 : 1.75);
hold on
plot(t, f(t), 'r-');
```

e) Criando a função degrau, e Plotando em um intervalo.

```
u = inline('(t >= 0)','t')
% Baixa Resolução
t = (-3 : 3);
figure(2)
plot (t,u(t),'b-');
xlabel(' t');
ylabel(' u(t)');
% Resolução alta
t = (-3 : 0.01 : 3);
hold on
plot(t, u(t), 'r-');
```

f) Criando um Pulso (diferença de degraus)

```
p = inline('(t >= -1) & (t < 1)','t');
t = (-2 : 0.01 : 2);
figure(3)
plot(t, p(t), 'b-');
xlabel(' t');
ylabel(' p(t) = u(t + 1) - u(t - 1)');
axis ([-2 2 - .1 1.1]);
```

g) Criando funções limitadas por degrau e plotando-as em um intervalo

```
g = inline(' exp(-0.3 * t). * cos(2 * pi * t). * (t >= 0)','t')

t = (-1 : 0.01 : 2);
g1) Plotando sem mudança de escala
figure(4)
subplot(3, 1,1)
plot(t, g(t));
xlabel(' t');
ylabel(' g(t)')';
grid on

g2) Plotando com mudança de escala e deslocamen to temporal
subplot(3, 1,2)
plot(t, g(2 * t + 1));
xlabel(' t');
ylabel(' g(2t + 1)')';
grid on

g3) Plotando com reversão de escala e deslocamen to temporal
subplot(3, 1,3)
plot(t, g(-t + 1));
xlabel(' t');
ylabel(' g(-t + 1)')';
grid on
```

Etapa 2 – Dado o sistema $(D^2+4D+k)y(t)=(D+1)x(t)$, obter as raízes do polinômio característico para k=5, k=7, k=45, utilizando o comando <ROOTS>.

```
>> r = roots([1 4 3]);

>> disp(['case (k=3): roots = [',num2str(r.'),']']);
```

Etapa 3 – Use o help do Matlab para entender o comando <EZPLOT> e utilize-o para plotar, em um único gráfico, cada equação mostrada abaixo (use os comandos hold on e grid on). Pesquise o comando <HEAVISIDE> para usar como DEGRAU do <EZPLOT>.

a) $y_0(t) = 5e^{-t} + e^{-2t}$,

b), $y_0(t) = (2 + 3t)e^{-3t} \qquad t \geq 0$

c) $h(t) = (-e^{-3t} + 2e^{-2t})u(t)$

Etapa 4 - Dado o sistema $(D^2 + 4D + k)y(t) = (3D + 1)x(t)$ com condições iniciais $y'(0) = 3$ e $y(0) = -7$, determine a componente de **Entrada Nula** da resposta para k=7, k=11, k=70, utilizando o comando <DSOLVE>.

Exemplo: Utilize os comandos no formato abaixo:

```
>> y_0 = dsolve('D2y+4*Dy+3*y=0','y(0)=-7','Dy(0)=3','t');

>> disp(['(a) k= 3 ; y_0 = ',char(y_0)])
```

Etapa 5 – Determine a **resposta ao impulso** (h(t)) para o sistema $(D^2 + 3D + 2)y(t) = (D + 2)x(t)$.

Este é um sistema de 2ª ordem com bo=0, inicialmente deve-se determinar a resposta para os modos naturais (yn(t) com as condições iniciais PADRÕES PARA FUNÇÃO IMPULSO: $y'(0) = 1$ e $y(0) = 0$.

LEMBRE-SE: Como *P(D)=D*, a resposta obtida deve ser diferenciada (derivada) durante a montagem final de h(t).

```
>> y_n = dsolve (...) ; % <<insira sua equação aqui

>> Dy_n = diff(y_n); << Assim se derivam as funções

>> disp(['h(t) = (',char(YYYY),')u(t)']); << Use o commando disp
para mostrar o resultado
```

Etapa 6 – O ANEXO A possui um script (parte 1), que realiza a convolução da função de entrada: $x(t) = (5t + 3)u(t)$ com a função h(t) resultante da tarefa 3.

Depois, na parte 2, o ANEXO 3 plota (no intervalo 0 a 3,75) e compara o resultado do algoritmo com o resultado matemático desta convolução, que é:

$$y_k(t) = (0.25e^{-2t} + 2e^{-t} + 2{,}5t - 2{,}25)u(t)$$

6.1 Execute o script do ANEXO A, analise-o e tente compreender seu funcionamento.

6.2 Como Verificação do aprendizado, faça a Convolução da função $h(t) = e^{2t}u(t)$ com $x(t) = 10e^{3t}u(t)$, compare, "plote" e verifique seu resultado com o algoritmo do Anexo A.

Obs.: Mantenha o intervalo de tempo existente no script.

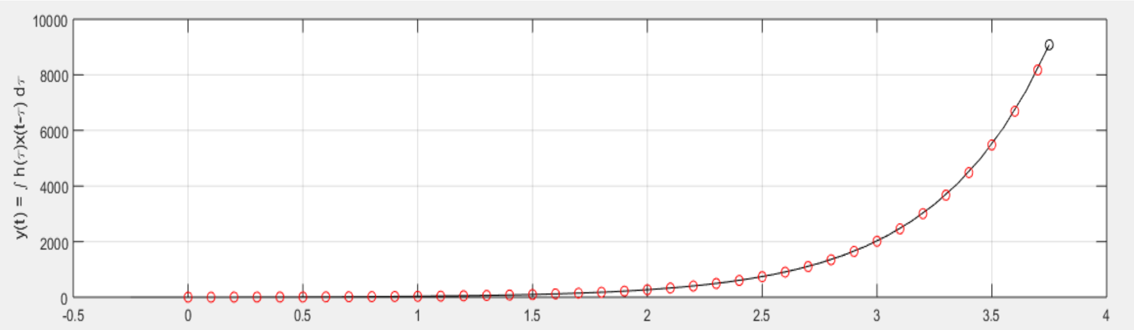
Para facilitar esta tarefa, evitando que você precise usar a CONVOLUÇÃO GRÁFICA, utilize o recorte abaixo da tabela de convolução para fazer esse exercício.

4	$e^{\lambda_1 t}u(t)$	$e^{\lambda_2 t}u(t)$	$\dfrac{e^{\lambda_1 t} - e^{\lambda_2 t}}{\lambda_1 - \lambda_2}\, u(t)$	$\lambda_1 \neq \lambda_2$

INSTITUTO FEDERAL DO ESPIRITO SANTO

Curso de Engenharia Elétrica

Gráfico para Conferência – item 4.2



NOTA:

1) Gráfico Contínuo PRETO (‘k-‘) e com Pontos PRETOS (‘ok‘): Resultado do algoritmo referente à Parte 1 do ANEXO A

2) Gráfico Pontilhado VERMELHO (‘r.‘): Resultado do algoritmo referente à Parte 2 do ANEXO A

ANEXO A

```
clc
clear all
close all
%============================================================
% Parte 1 - Realizando a convolução gráfica (LATHI)
%============================================================
figure(1)
h = inline('(1*exp(-t)-1*exp(-2*t)).*(t>=0)','t');
x = inline('(5*t+3).*(t>=0)','t');
dtau = 0.005;
tau = -1:dtau:4;
ti = 0;
tvec = -0.25:0.1:3.75;
y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
for t = tvec,
    ti = ti + 1; % Time index
    xh = x(t-tau).*h(tau);
    lxh = length(xh);
    y(ti) = sum(xh.*dtau); % Trapezoidal approximation of integral
    subplot(2,1,1)
    plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok')
    %axis([tau(1) tau(end) -2.0 2.5])
    patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
        [zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
        [0.8 0.8 0.8],'edgecolor','none')
    xlabel('\tau')
    legend('h(\tau)','x(t-\tau)','t','h(\tau)x(t-\tau)')
    c = get(gca,'children');
    set(gca,'children',[c(2);c(3);c(4);c(1)]);
    subplot(2,1,2)
    plot(tvec,y,'k',tvec(ti),y(ti),'ok')
    xlabel('t')
    ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau')
    %axis([tau(1) tau(end) -1.0 2.0])
    grid
    drawnow
end
%============================================================
% Parte 2 - Verificação com função calculada (Prof. Arnaldo)
%============================================================
subplot(2,1,2)
hold on
x=inline('(0.25*exp(-2*t)+2*exp(-t)+2.5*t-2.25)','t')
t=0:0.1:3.75;
plot(t,x(t),'r.')
```