

Contents

- [Atividade 06 - Discretização e Simulação de Sistemas Dinâmicos](#)
- [Parâmetros Iniciais](#)
- [Inicialização dos Vetores](#)
- [Ganhos do controlador](#)
- [Planta e Coeficientes da Equação de Diferenças](#)
- [Condicionamento de Sinais e Coeficientes da Equação de Diferenças](#)
- [Simulação](#)
- [Gráficos das variáveis](#)

Atividade 06 - Discretização e Simulação de Sistemas Dinâmicos

```
%Welliton Leal - a1543857
```

Parâmetros Iniciais

```
clear;
clc;
close all;

NAC = 2*50;      %Número de amostras por ciclo
NC = 2;          %Número de ciclos
N = NC*NAC;      %Número de pontos / N = 200
T = 0.1e-3;      %Tempo da amostragem / Chaveamento

s = tf('s');
z = tf('z',T);

Vmax = 10;       %tensão máxima da planta
Vmin = -10;      %tensão mínima da planta
```

Inicialização dos Vetores

```
t = zeros(1,N); %variavel de tempo
x = zeros(1,N); %referencia
e = zeros(1,N); %erro
u = zeros(1,N); %saida controlador
y = zeros(1,N); %saida planta
ym = zeros(1,N); %saida sensor
yr = zeros(1,N); %planta + ruído
kq = 0;
```

Ganhos do controlador

```
K1 = 0.10;
K2 = 0.20;
```

Planta e Coeficientes da Equação de Diferenças

```

%Planta Contínua P(s) / Função de Transferência em Tempo Contínuo (CTTF)
Pc = 9e6 / (s^2 + 3e3*s + 9e6);
%Planta Discreta P(z) / Função de Transferência em Tempo Discreto (DTTF)
Pd = c2d(Pc,T);

%Coeficientes ED
[npd,dpd] = tfdata(Pd,'v')
a1 = dpd(2);
a0 = dpd(3);
b1 = npd(2);
b0 = npd(3);

```

```

npd =

    0    0.0405    0.0367

```

```

dpd =

    1.0000   -1.6636    0.7408

```

Condicionamento de Sinais e Coeficientes da Equação de Diferenças

```

%Sensor Contínuo P(s)
Sc = 36e6 / (s^2 + 3e3*s + 36e6);
%Sensor Discreto
Sd = c2d(Sc,T);

%Coeficientes ED
[nsd,dsd] = tfdata(Sd,'v')
as1 = dsd(2);
as0 = dsd(3);
bs1 = nsd(2);
bs0 = nsd(3);

```

```

nsd =

    0    0.1585    0.1433

```

```

dsd =

    1.0000   -1.4390    0.7408

```

Simulação

```

for k = 2:N
    % Vetor de tempo disc. para graficos
    t(k) = k*T;
    kq = kq + 1;

    % Geração do sinal de referência

```

```

if kq <= NAC/2
    x(k) = 5;
else
    x(k) = 0;
end

if kq == NAC
    kq = 0;
end

%erro = referencia + saida sensor
e(k) = x(k) - ym(k);

%saida do controlador
u(k) = u(k-1) + K1*(e(k) - K2*e(k-1));

%ceifador
if u(k) > Vmax
    u(k) = Vmax;
end
if u(k) < Vmin
    u(k) = Vmin;
end

%saida da planta
y(k+1) = -a1*y(k) - a0*y(k-1) + b1*u(k) + b0*u(k-1);

%adicionando ruido de medida ao sistema
yr(k+1) = y(k+1) + 0.01*3*randn;

%saida do sensor / transdutor
ym(k+1) = -as1*ym(k) - as0*ym(k-1) + bs1*yr(k) + bs0*yr(k-1);
end

```

Gráficos das variáveis

```

y(k+1) = [];
yr(k+1) = [];
ym(k+1) = [];

subplot(411)
plot(t, yr)
hold on
plot(t, x, '--')
set(findall(gcf, 'Type', 'line'), 'LineWidth', 2)
grid on
legend('yr - saida da planta', 'x - sinal de entrada')

subplot(412)
plot(t, e, '--', 'Color', [0.3010 0.7450 0.9330])
hold on
plot(t, u, ':')
set(findall(gcf, 'Type', 'line'), 'LineWidth', 2)
grid on
legend('e - sinal de erro', 'u - sinal de controle')

subplot(413)
plot(t, yr)
hold on
plot(t, u, ':')

```

```

set(findall(gcf,'Type','line'),'LineWidth',2)
grid on
legend('yr - saída da planta','u - sinal de controle')

subplot(414)
plot(t,yr)
hold on
plot(t,x,'--')
hold on
plot(t,ym)
set(findall(gcf,'Type','line'),'LineWidth',2)
grid on
legend('yr - saída da planta','x - sinal de entrada','ym - saída do sensor')

```

