Análise e Modelagem do Sistema Motor do Lego

Alunos: Victor Pinheiro Matias - 538104

Kelvin Leandro Martins - 540006

Pedro Leinos Falcão Cunha - 542114

Universidade Federal do Ceará

Engenharia de Computação

Introdução

Este relatório foca na estimação de parâmetros de um sistema de motor Lego, utilizando metodologia off-line e o modelo ARX (AutoRegressivo com Entrada Exógena). Através deste método, é possível obter um modelo matemático que aproxime o comportamento do sistema em resposta a entradas conhecidas, mesmo quando os processos físicos são complexos e não permitem uma modelagem direta.

Desenvolvimento das Atividades 1 e 2.

A atividade envolveu a estimação de parâmetros para modelos de primeira e segunda ordem de um sistema de motor Lego. O processo foi dividido nas seguintes etapas:

- Programação: Utilizando MATLAB, foram montadas as equações normais e estimados os parâmetros para os dois modelos propostos. O código MATLAB empregado realizou o carregamento dos dados, a estimação dos parâmetros via metodologia off-line e o modelo ARX, e a simulação das respostas do sistema.
- Geração de Gráficos: Foram produzidos gráficos comparando os valores medidos e estimados para ambos os modelos, permitindo a visualização da adequação dos modelos em relação aos dados experimentais.
- Validação dos Modelos: O critério de validação utilizou a norma do erro de estimação (MSE), comparando os valores medidos com os valores estimados pelos modelos.
- Função de Transferência G(z): As funções de transferência foram obtidas para cada modelo, e a transformada contínua G(s) foi adquirida através do comando d2c.
- Comparação das Respostas ao Degrau: As respostas ao degrau, tanto no tempo discreto quanto no contínuo, foram comparadas para o modelo de segunda ordem.

Atividade 1

Modelo de Primeira Ordem.

Parâmetros Estimados (theta1):

0.9658: Este é o coeficiente autoregressivo a1 que multiplica o valor da saída no instante anterior y(k−1). Um valor próximo a 1 indica uma relação diretamente proporcional e quase igual em magnitude à saída anterior, sugerindo uma baixa inércia ou atraso do sistema.

0.0376: Este é o coeficiente b1 que multiplica o valor da entrada no instante anterior (k−1). Um valor pequeno implica que a entrada tem um impacto modesto na saída do sistema em qualquer instante de tempo.

Equação à Diferenças e Equação no Domínio Z:

A equação fornecida descreve o comportamento do sistema em termos da saída y(k) como uma função da saída anterior y(k−1) e da entrada anterior u(k−1). Em termos de sistemas dinâmicos, indica um modelo que depende fortemente do seu estado anterior com uma influência relativamente pequena da entrada mais recente.

$$y(k) \;=\; 0.9658\, y(k-1) \;+\; 0.0376\, u(k-1)$$

Aplicando a transformada Z à equação à diferenças determinada:

$$Y(z) \;=\; 0.9658\, Y(z)\, z^{-1} \;+\; 0.0376\, U(z)\, z^{-1}$$

Erro de Estimação (J):

O erro de estimação 1.574489e+02 é uma medida da precisão do modelo em relação aos dados medidos. Conhecido como o erro quadrático médio (MSE), este valor é alto, indicando que o modelo de primeira ordem pode não ser o melhor em capturar a dinâmica do sistema.

Função de Transferência em Z (Gz1):

A função de transferência G(z) no domínio Z para o modelo de primeira ordem é representada como uma razão entre a saída e a entrada, transformadas pelo operador z^{-1} (que representa um atraso de um passo de tempo no domínio discreto). A forma da função de transferência sugere que a saída reage à entrada com um atraso significativo.

$$\mathrm{G(z)} = \frac{0.0376\, z^{-1}}{1 - 0.9658\, z^{-1}}$$

Modelo de Segunda Ordem

Parâmetros Estimados (theta2):

1.9217: Coeficiente a1 que multiplica y(k−1), sugerindo que a saída é fortemente afetada pelo valor que ela tinha um passo de tempo antes.

-0.9271: Coeficiente a0 que multiplica y(k−2), que representa a influência da saída dois passos de tempo antes.

0.0027: Coeficiente b1 que multiplica u(k−1), indicando uma influência direta, mas muito pequena, da entrada no passo de tempo anterior na saída atual.

0.0027: Coeficiente b0 que multiplica u(k−2), indicando uma influência direta, mas muito pequena, da entrada no passo de tempo anterior na saída atual.

Equação à Diferenças e Equação no Domínio Z:

A equação à diferenças para o modelo de segunda ordem incorpora mais termos, refletindo uma dinâmica mais complexa e rica que considera a saída nos dois últimos passos de tempo e a entrada no último passo de tempo.

$$y(k) = 1.9217\, y(k-1) - 0.9271\, y(k-2) + 0.0027\, u(k-1) + 0.0027\, u(k-2)$$

Aplicando a transformada Z à equação à diferenças determinada:

$$Y(z) = 1.9217\, Y(z)\, z^{-1} - 0.9271\, Y(z)\, z^{-2} + 0.0027\, U(z)\, z^{-1} + 0.0027\, U(z)\, z^{-2}$$

Erro de Estimação (Jest):

O valor de 4.291582e-01 representa o erro quadrático médio (MSE) para o modelo de segunda ordem. Este valor é significativamente menor do que o MSE para o modelo de primeira ordem, o que implica que o modelo de segunda ordem é mais preciso na estimativa dos valores de saída com base nos dados medidos. Um menor MSE indica que o modelo tem um ajuste melhor aos dados experimentais e, portanto, pode ser considerado um modelo mais confiável para representar o sistema físico.

Função de Transferência em Z (Gz2):

A função de transferência G(z) no domínio Z para o modelo de segunda ordem é mais complexa do que a de primeira ordem. Ela inclui termos que representam a saída e a entrada com um e dois atrasos de tempo (representados por z^{-1} e z^{-2}, respectivamente). A presença desses termos adicionais permite que o modelo de segunda ordem capture melhor a dinâmica do sistema, que pode incluir comportamentos como oscilações ou sobreposições mais complexas antes de atingir um estado estável.

$$G(z) = \frac{0.0027\, z^{-1} + 0.0027\, z^{-2}}{1 - 1.9217\, z^{-1} + 0.9271\, z^{-2}}$$

Resultados e Análises

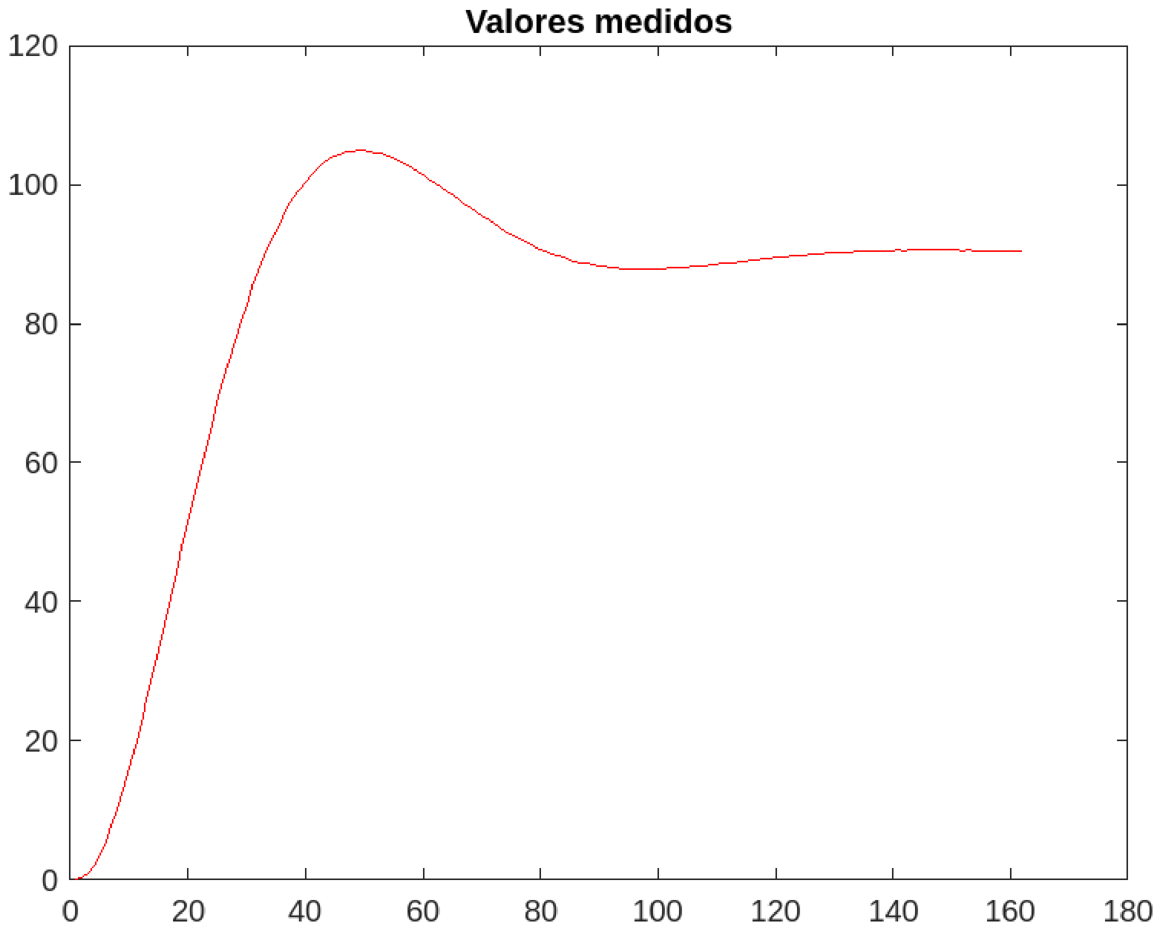

Gráfico 1: Valores Medidos - Este gráfico apresenta os valores medidos de y(k), que representam a resposta do sistema ao sinal de entrada degrau.

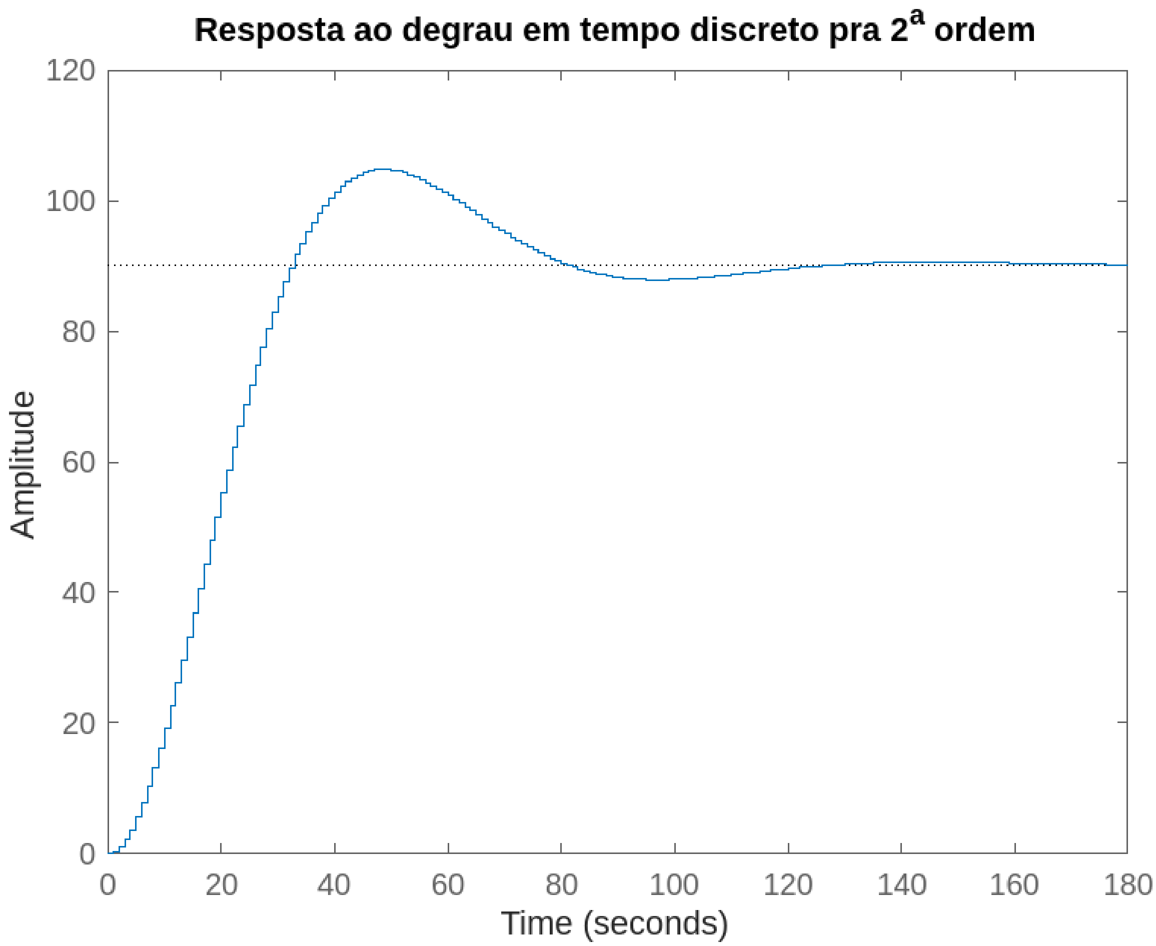

Gráfico 2: Resposta ao Degrau de Tempo Discreto - Mostra a resposta do modelo de segunda ordem ao degrau em tempo discreto.

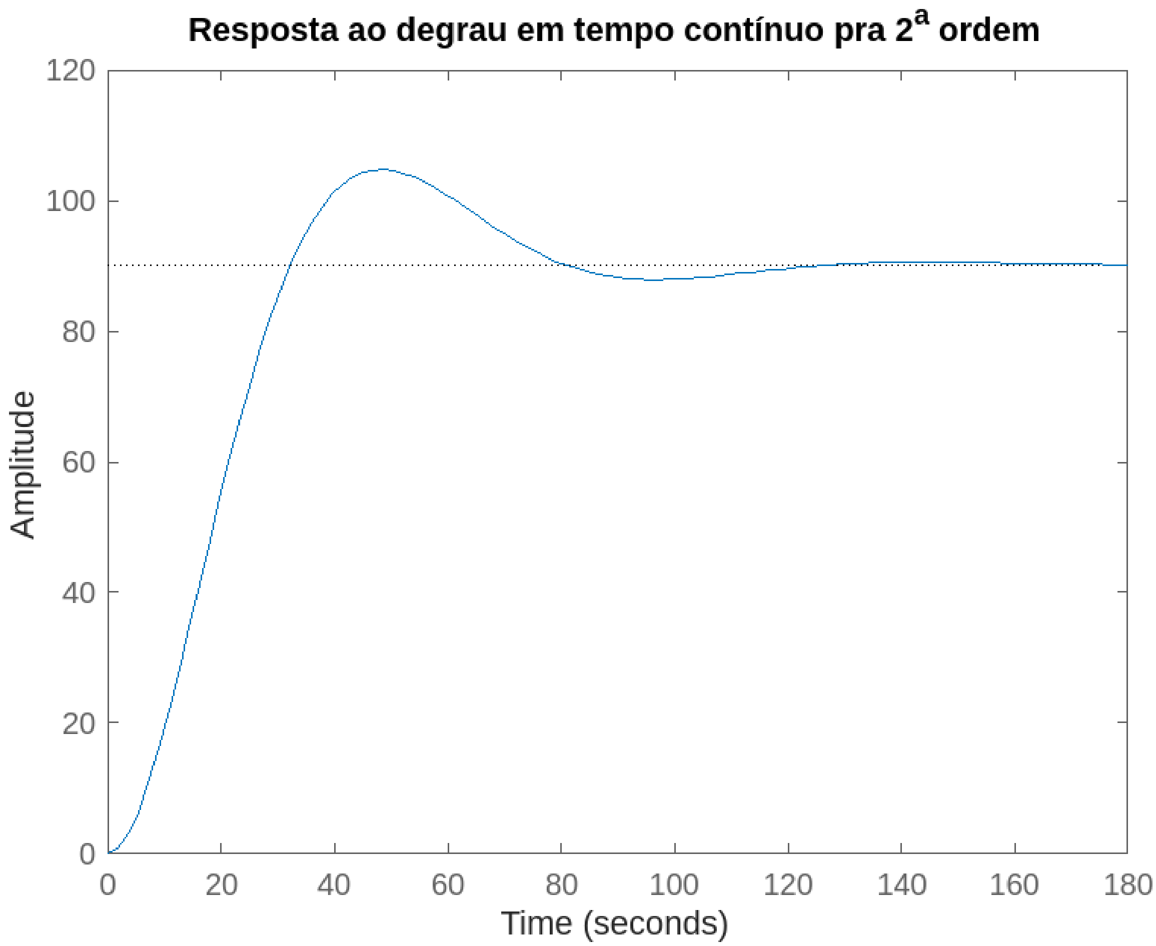

Gráfico 3: Resposta ao Degrau de Tempo Contínuo - Ilustra a resposta do modelo de segunda ordem ao degrau em tempo contínuo.

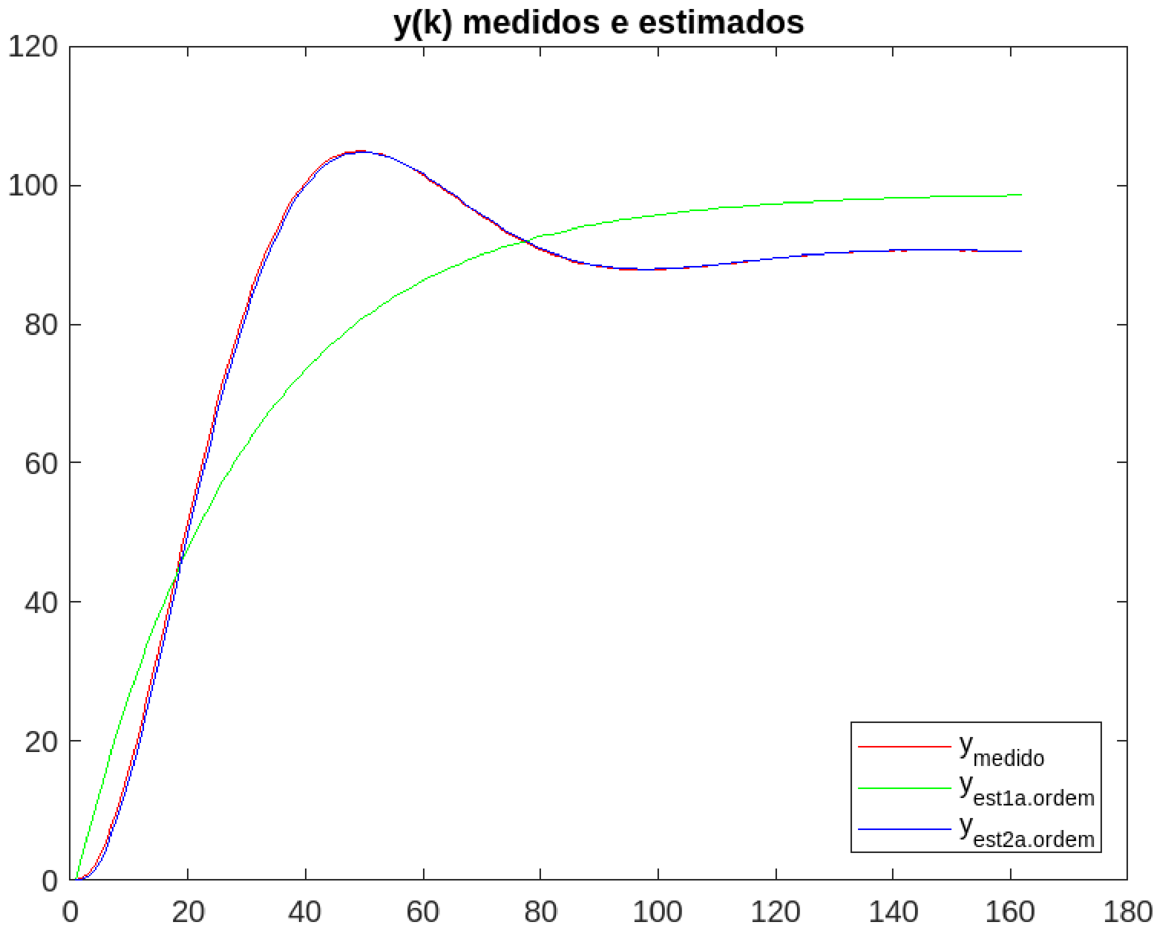

Gráfico 4: Comparação dos Valores Medidos e Estimados - Compara os valores medidos de y(k) com os valores estimados pelos modelos de primeira e de segunda ordem.

Atividade 2

A análise de sistemas discretos e o impacto do tempo de amostragem são essenciais no design de sistemas de controle. A atividade 2 visa estudar como as características de frequência de um sistema discreto são afetadas pelo tempo de amostragem. Através da aproximação bilinear (Tustin), é possível transformar a função de transferência de um sistema discreto em um equivalente de tempo contínuo e comparar o comportamento da resposta em frequência com diferentes tempos de amostragem.

Função de Transferência em Z (Gz2):

A função de transferência G(z) foi calculada a partir da equação a diferenças fornecida, resultando em uma expressão que relaciona a saída do sistema Y(z) à entrada X(z) no domínio do tempo discreto.

Equação à diferenças fornecida inicialmente:

$$y(k) = 1.238\,y(k-1) - 0.3016\,y(k) + 0.03175\,x(k-2) - 0.03175\,x(k)$$

Aplicando a transformada Z à equação à diferenças determinada:

$$Y(z) = 1.238\,Y(z)\,z^{-1} - 0.3016\,Y(z) + 0.03175\,X(z)\,z^{-2} - 0.03175\,X(z)$$

Por fim, determinando a função de transferência G(z):

$$\mathrm{G(z)} = \frac{-0.03175 + 0.03175\,z^{-2}}{1.3016 - 1.238\,z^{-1}}$$

Transformação de G(z) para G(s):

Utilizando a aproximação bilinear de Tustin,

$$z = \frac{2 + T.s}{2 - T.s}$$

a função de transferência G(z) foi convertida para G(s), um equivalente de tempo contínuo. Esta transformação permite analisar o sistema usando técnicas de sistemas contínuos e entender o efeito do tempo de amostragem na resposta em frequência.

Funções de Transferência para Diferentes Tempos de Amostragem:

Foram obtidas as funções de transferência G1(s) e G2(s) para tempos de amostragem T=0.1s e T=1.0s, respectivamente.

$$G1(s) = \frac{-0.0254\, s}{0.0254\, s^2 + 0.5206\, s + 0.2544}$$

$$G2(s) = \frac{-0.254\, s}{2.54\, s^2 + 5.206\, s + 0.2544}$$

Resultados e Análises

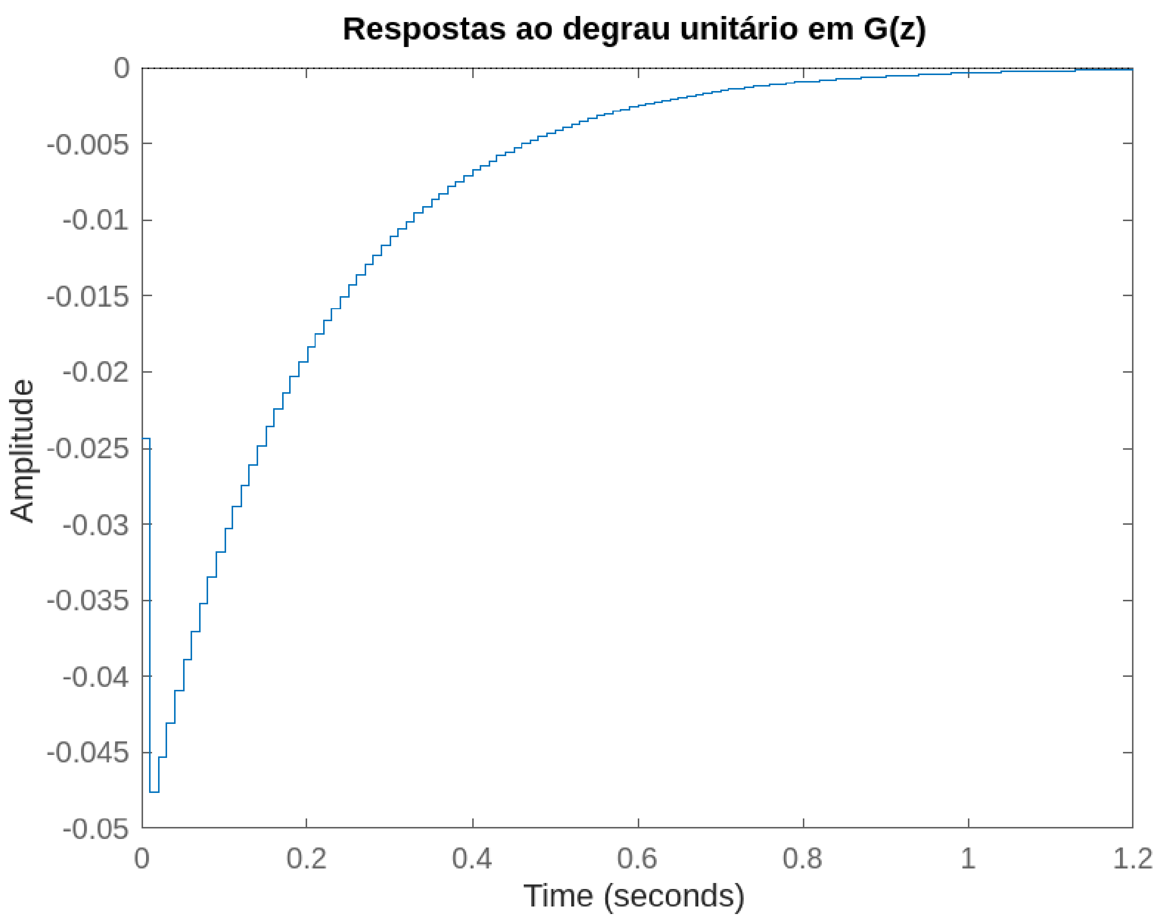

Figura 1: Resposta ao Degrau em G(z). Resposta ao degrau unitário do sistema G(z) no tempo discreto. A resposta mostra uma escalada escalonada, típica de sistemas discretos, e uma estabilização gradual.

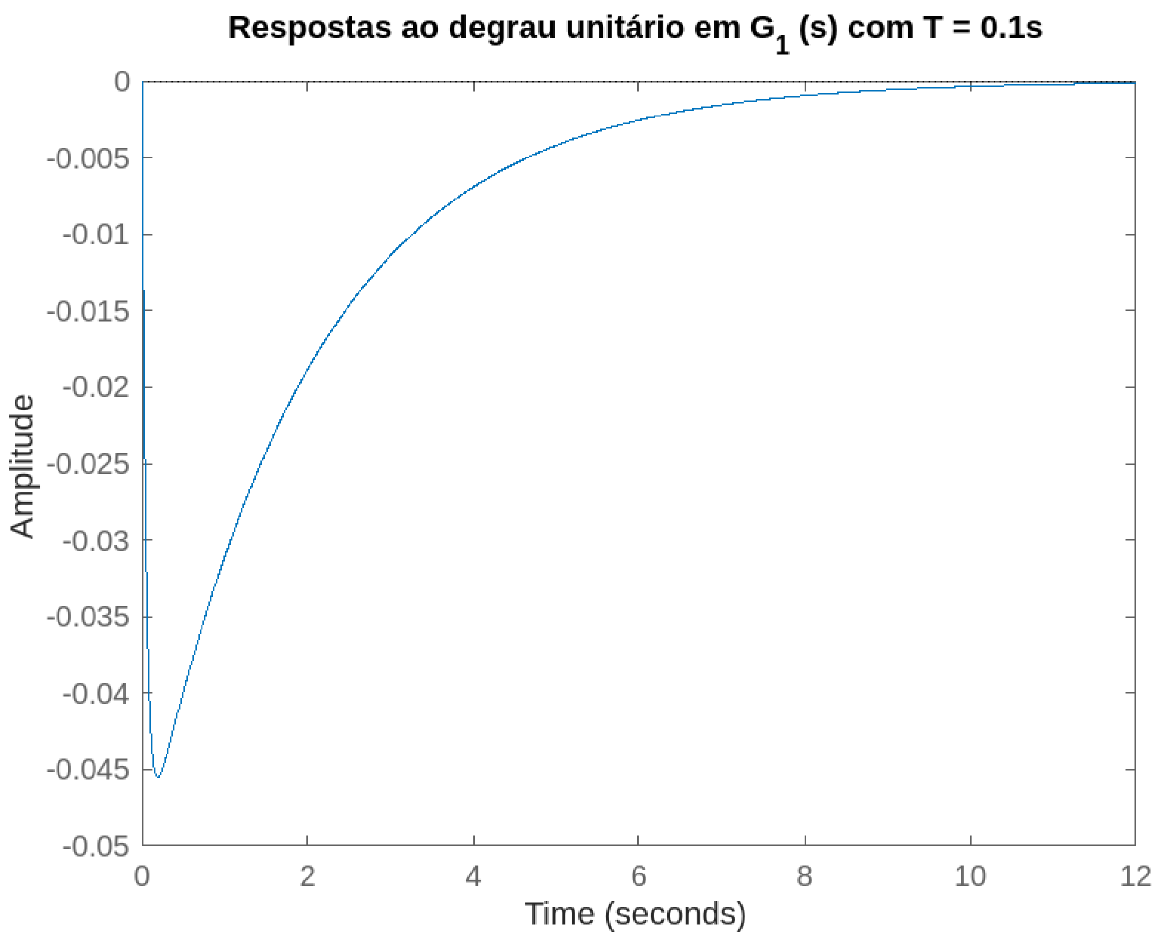

Figura 2: Resposta ao degrau unitário do sistema G1(s) com um tempo de amostragem de 0.1s. A resposta rápida sugere que um tempo de amostragem curto pode resultar em uma resposta mais ágil do sistema.

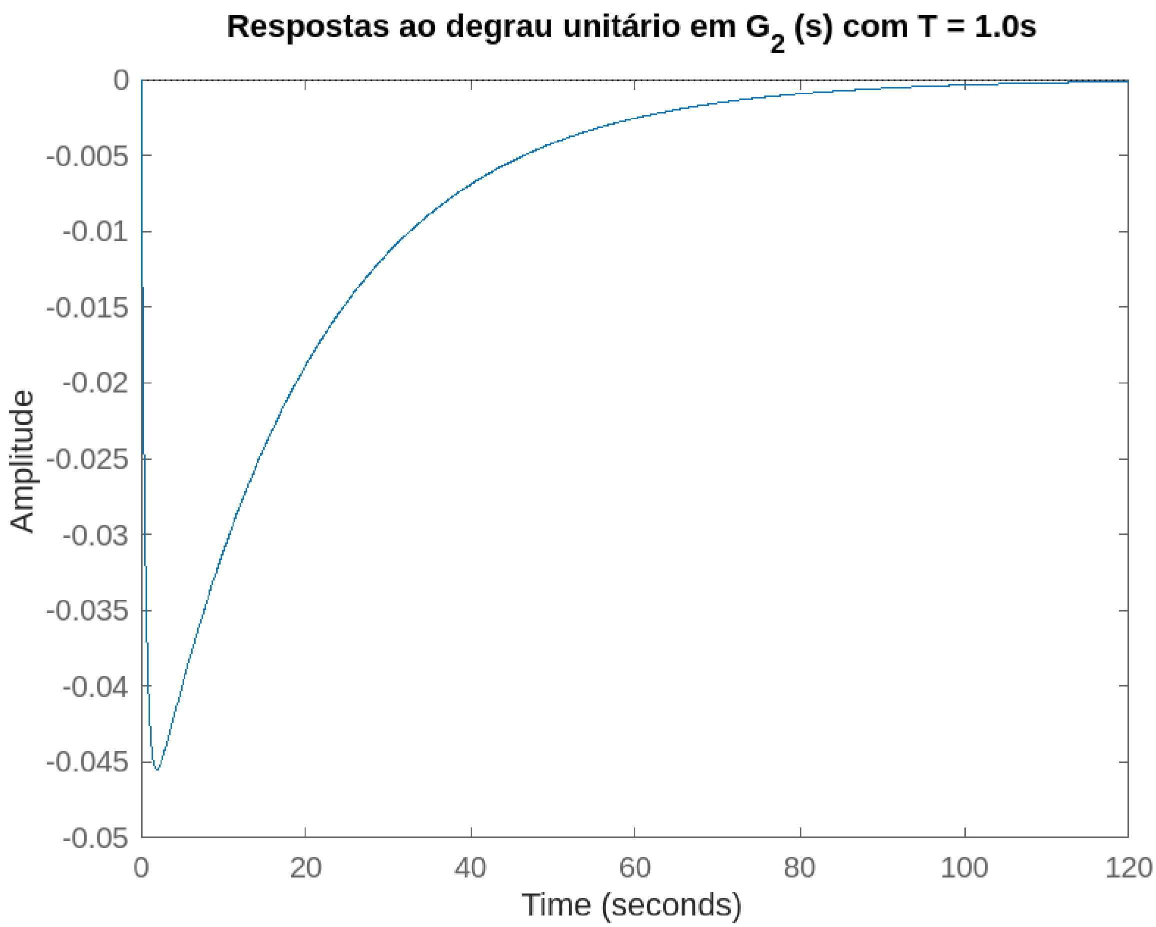

Figura 3: Resposta ao Degrau em G2(s) com T = 1s. Resposta ao degrau unitário do sistema G2(s) com um tempo de amostragem de 1s. A resposta mais lenta indica que um tempo de amostragem maior suaviza a resposta do sistema, o que pode ser benéfico para sistemas que precisam de estabilidade.

Conclusão sobre a Análise de Sistemas de Controle Discreto

Os resultados obtidos demonstram que o tempo de amostragem é um fator crítico na determinação da resposta de um sistema discreto. Um tempo de amostragem mais curto, como mostrado na resposta de G1(s), leva a uma resposta mais rápida e pode ser apropriado para aplicações que requerem acompanhamento rápido das mudanças de entrada ou onde a atenuação de ruído de alta frequência é desejável. Em contraste, um tempo de amostragem mais longo, como observado na resposta de G2(s), resulta em uma resposta mais lenta e pode ser vantajoso em aplicações onde a estabilidade e a rejeição de ruídos de alta frequência são críticas.

Código MATLAB Atividade 1

```
close all

clear all

clc

load GrupoRobo_10.mat; % baixar os dados do grupo A

l = length(z1); % comprimento do vetor z1

y = z1(:, 1); % valores medidos de y

u = z1(:, 2); % entrada de sinal

figure (1)

plot(y, 'r');

title("Valores medidos");

%$$$$$$$$$$$$$$$$$$$$$$$ Início Modelo de 1a. ordem
$$$$$$$$$$$$$$$$$$$$$$$

X1 = [y(1:end-1), u(1:end-1)];% Modelo Discreto

theta1 = pinv(X1'*X1)*X1'*y(2:end); % Obtenção dos parâmetros a1,b1

y1_est = sim_1st(u, theta1); % Simulando resposta pro modelo de 1a. ordem

erro1 = y - y1_est; % Erro de estimação

Jest1 = norm(erro1)^2 / l;  % MSE

fprintf('Os parâmetos estimados foram:\n'), theta1

fprintf('-------------------------------------------------------------
--\n');

fprintf('A equação à diferenças é:\ny(k) = %.4f y(k-1) + %.4f
u(k-1)\n\n', theta1(1), theta1(2));

fprintf('-------------------------------------------------------------
--\n');

fprintf('A aproximação J é: \n%d\n\n', Jest1);

fprintf('-------------------------------------------------------------
--\n');
```

```
fprintf('A função em Z é:\nY(z) = %.4f Y(z) z^-1 + %.4f U(z) z^-1
\n\n',theta1(1), theta1(2));

fprintf('------------------------------------------------------------
--\n');

fprintf('A função de transferência em z, G(z) é:\n\n');

fprintf('Y(z)/U(z) = (%.4f z^-1 )/(1 - %.4f z^-1
)\n\n',theta1(2),abs(theta1(1)));

%$$$$$$$$$$$$$$$$$$$$$$$$ Final Modelo de 1. Ordem
$$$$$$$$$$$$$$$$$$$$$$$$

%$$$$$$$$$$$$$$$$$$$$$$$ Início Modelo de 2a. ordem
$$$$$$$$$$$$$$$$$$$$$$$

X = [y(2:end-1), y(1:end-2), u(2:end-1), u(1:end-2)]; % Modelo Discreto

theta2 = pinv(X'*X)*X'*y(3:end); % Obtenção dos parâmetros a1, a0, b1, b0

y2_est = sim_2nd(u, theta2); % Simulando resposta pro modelo de 2a. ordem

erro2 = y - y2_est; % erro de estimação

Jest2 = norm(erro2)^2 / l;  % MSE

Gz2 = tf_z_2nd(theta2); % Função de transferência no domínio Z pra 2a.
ordem

Gs2 = d2c(Gz2); % Função de transferência no domínio S pra 2a. ordem

fprintf('Os parâmetos estimados foram:'), theta2

fprintf('------------------------------------------------------------
--\n');

fprintf('A equação a diferenças é:\ny(k) = %.4f y(k-1) + %.4f y(k-2) +
%.4f u(k-1) + %.4f u(k-2)\n\n', theta2(1), theta2(2), theta2(3),
theta2(4));

fprintf('------------------------------------------------------------
--\n');

fprintf('A aproximação Jest é: \n%d\n\n', Jest2);

fprintf('------------------------------------------------------------
--\n');

fprintf('A função em Z é:\nY(z) = %.4f Y(z) z^-1 + %.4f Y(z) z^-2 +%.4f
U(z) z^-1 + %.4f U(z) z^-2\n\n', theta2(1), theta2(2),
theta2(3),theta2(4));
```

```
fprintf('---------------------------------------------------------------
--\n');

fprintf('A função de transferência em z, G(z) é:\n\n')

fprintf('Y(z)/U(z) = (%.4f z^-1 + %.4f z^-2)/(1 - %.4f z^-1 + %.4f
z^-2)', theta2(3), theta2(4), theta2(1), abs(theta2(2)));

figure (2)

step(90*Gz2, 180) % Resposta ao degrau de tempo discreto para o modelo de
2a. ordem.

title("Resposta ao degrau em tempo discreto pra 2^a ordem")

figure (3)

step(90*Gs2, 180) % Resposta ao degrau de tempo contínuo para o modelo de
2a. ordem.

title("Resposta ao degrau em tempo contínuo pra 2^a ordem")

%$$$$$$$$$$$$$$$$$$$$$$$$$ Final Modelo de 2. Ordem
$$$$$$$$$$$$$$$$$$$$$$$$

figure (4)

plot(y,'r')

hold on

plot(y1_est,'g');

hold on

plot(y2_est,'b');

hold on

title("y(k) medidos e estimados")

legend({'y_m_e_d_i_d_o','y_e_s_t_1_a_._o_r_d_e_m','y_e_s_t_2_a_._o_r_d_e_
m'},'Location','southeast')

%$$$$$$$$$$$$$$$$$$$$$$$$$$$$ Funções Utilizadas
$$$$$$$$$$$$$$$$$$$$$$$$$$$

function y_est1 = sim_1st(u, theta)

   % Inicialização do vetor de saída estimado

   y_est1 = zeros(size(u));
```

```
    % Condição inicial
    y_est1(1) = 0; % Supondo condição inicial zero

    % Simulação do modelo de primeira ordem
    for k = 2:length(u)
        y_est1(k) = theta(1) * y_est1(k-1) + theta(2) * u(k-1);
    end
end
function y_est2 = sim_2nd(u, theta)
    % Inicialização do vetor de saída estimado
    y_est2 = zeros(size(u));

    % Condições iniciais
    y_est2(1) = 0; % Supondo condição inicial y(k=0) = 0
    y_est2(2) = 0; % Supondo condição inicial y(k=1) = 0

    % Simulação do modelo de segunda ordem
    for k = 3:length(u)
        y_est2(k) = theta(1) * y_est2(k-1) + theta(2) * y_est2(k-2) +
theta(3) * u(k-1) + theta(4) * u(k-2);
    end
end
function Gz2 = tf_z_2nd(theta)
    Gz2 = tf([0, theta(4), theta(3)], [1, -theta(1), -theta(2)], 1,
'Variable','z^-1');
end
```

Código MATLAB Atividade 2

```
close all

clear all

clc

% $$$$$$$$$$$$$$$$$$$$$$$$$$ Modelo de 2a. ordem
$$$$$$$$$$$$$$$$$$$$$$$$$$

% y(k) = a1y(k-1) + a0y(k-2) + b1u(k-1) + b0u(k-2)

% y(k) = 1.238 y(k-1) - 0.3016 y(k) + 0.03175 x(k-2) – 0.03175 x(k)

% y(k) - 1.238 y(k-1) + 0.3016 y(k) = 0.03175 x(k-2) – 0.03175 x(k)

% Y(z) - 1.238 Y(z) z^-1 + 0.3016 Y(z) = 0.03175 X(z) z^-2 - 0.03175 X(z)

% Y(z) [1.3016 - 1.238 z^-1] = X(z) [-0.03175 + 0.03175 z^-2]

% G(z) = [-0.03175 + 0.03175 z^-2]

%        -------------------------

%          [1.3016 - 1.238 z^-1]

numerator = [-0.03175, 0, 0.03175];

denominator = [1.3016, -1.238, 0];

Ts = 0.01;

Ts1 = 0.1;

Ts2 = 1.0;

% Criando a função de transferência no domínio Z

Gz = tf(numerator, denominator, Ts, 'Variable','z^-1');

% Substituindo z por (2+Ts1*s)/(2-Ts1*s) em Gz para obter G1_s

G1_s = tf(tustin_num(numerator, Ts1), tustin_den(denominator, Ts1)); %
Convertendo a função de transferência discreta para contínua

% Substituindo z por (2+Ts2*s)/(2-Ts2*s) em Gz para obter G2_s
```

```
G2_s = tf(tustin_num(numerator, Ts2), tustin_den(denominator, Ts2)); %
Convertendo a função de transferência discreta para contínua

Gz

G1_s

G2_s

figure (1)

step(Gz) % Resposta ao degrau de tempo discreto para o modelo de 2a.
ordem.

title("Respostas ao degrau unitário em G(z)")

figure (2)

step(G1_s) % Resposta ao degrau de tempo discreto para o modelo de 2a.
ordem.

title("Respostas ao degrau unitário em G_1 (s) com T = 0.1s")

figure (3)

step(G2_s) % Resposta ao degrau de tempo contínuo para o modelo de 2a.
ordem.

title("Respostas ao degrau unitário em G_2 (s) com T = 1.0s")

function numerator_s = tustin_num(numerator, Ts)

   numerator_s = [Ts^2 * (numerator(1) + numerator(3)), 4 * Ts *
(numerator(1) - numerator(3)), 4 * (numerator(1) + numerator(3))];

end

function denominator_s = tustin_den(denominator, Ts)

   denominator_s = [Ts^2 * (denominator(1) - denominator(2)), 4 * Ts *
denominator(1), 4 * (denominator(1) + denominator(2))];

end
```