



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
[EEC 2101] CONTROLE AVANÇADO

2ª Lista de Exercícios

**Anna Giselle Câmara Dantas Ribeiro
Cristiano Gurgel de Castro
Diogo Leite Rebouças
Thiago Medeiros Barros**

**Natal – RN
2010.1**

Questão 1

Considere o sistema:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{0.5}{(s+0.5)(s+1)}$$

Pede-se:

- Discretize o sistema com um período de amostragem de 0.2 s
- Obtenha o modelo FSR (resposta ao degrau Truncada) do sistema considerando $N = 10$
- Calcule a resposta do sistema a uma entrada qualquer utilizando a equação obtida no item (b) e compare com o seu valor exato
- Obtenha a expressão do sinal de controle do controlador DMC, considerando $NY = NU = 5$ e $\lambda = 0.8$
- Simule no Matlab[®] (ou Scilab[®]) o sistema com o controlador projetado no item (d)

Resolução:

Item A

Uma maneira simples de realizar a discretização de funções de transferência é utilizar o método da invariância ao degrau, no qual a idéia é a de se utilizar um Segurador de Ordem Zero (Zero Order Holder – ZOH) para aproximar a função de transferência $G(s)$ de $G(z)$. Matematicamente, a discretização de uma função de transferência analógica $G(s)$ por tal método é dada por:

$$G_{\text{ZOH}}(z) = G(z) = (1 - z^{-1}) \mathcal{Z} \left\{ \mathcal{L}^{-1} \left[\frac{G(s)}{s} \right] \Big|_{t=nT} \right\} \quad (1)$$

em que \mathcal{Z} corresponde à Transformada Z, \mathcal{L}^{-1} à Transformada Inversa de Laplace e, para $t = nT$, T corresponde ao período de amostragem e $z = e^{-sT}$.

Então, para a função de transferência $G(s)$ dada no enunciado:

$$\frac{G(s)}{s} = \frac{0.5}{s(s+0.5)(s+1)} \quad (2)$$

Expandindo a Eq. 2 em frações parciais, tem-se que:

$$\begin{aligned}
\frac{G(s)}{s} &= \left[\frac{0.5}{s(s+0.5)(s+1)} \right] \\
&= \frac{A}{s} + \frac{B}{s+0.5} + \frac{C}{s+1}
\end{aligned} \tag{3}$$

Mas,

$$\begin{aligned}
A &= \left. \frac{0.5}{(s+0.5)(s+1)} \right|_{s=0} = \frac{0.5}{0.5} = 1 \\
B &= \left. \frac{0.5}{s(s+1)} \right|_{s=-0.5} = \frac{0.5}{-0.25} = -2 \\
C &= \left. \frac{0.5}{s(s+0.5)} \right|_{s=-1} = \frac{0.5}{0.5} = 1
\end{aligned}$$

Então, aplicando então a transformada inversa de Laplace, tem-se que:

$$\begin{aligned}
\mathcal{L}^{-1} \left[\frac{G(s)}{s} \right] &= \mathcal{L}^{-1} \left[\frac{1}{s} - 2 \frac{1}{s+0.5} + \frac{1}{s+1} \right] \\
&= \mathcal{L}^{-1} \left[\frac{1}{s} \right] - 2 \mathcal{L}^{-1} \left[\frac{1}{s+0.5} \right] + \mathcal{L}^{-1} \left[\frac{1}{s+1} \right] \\
&= 1 - 2e^{-0.5t} + e^{-t}
\end{aligned} \tag{4}$$

Substituindo a Eq. 4 na Eq. 1 quando $t = nT$, tem-se:

$$\begin{aligned}
G(z) &= (1 - z^{-1}) \mathcal{Z} \left\{ \left. \mathcal{L}^{-1} \left[\frac{G(s)}{s} \right] \right|_{t=nT} \right\} \\
&= \left(\frac{z-1}{z} \right) \mathcal{Z} \left[1 - 2e^{-0.5nT} + e^{-nT} \right] \\
&= \frac{z-1}{z} \left\{ \mathcal{Z}[1] - 2\mathcal{Z}[e^{-0.5nT}] + \mathcal{Z}[e^{-nT}] \right\} \\
&= \frac{z-1}{z} \left(\frac{z}{z-1} - 2 \frac{z}{z - e^{-0.5nT}} + \frac{z}{z - e^{-nT}} \right) \\
&= \frac{z-1}{z-1} - 2 \frac{z-1}{z - e^{-0.5nT}} + \frac{z-1}{z - e^{-nT}}
\end{aligned} \tag{5}$$

Desconsiderando o termo n , referente a n -ésima amostra em nT e substituindo $T = 0.2$, tem-se:

$$\begin{aligned}
G(z) &= \frac{z-1}{z-1} - 2\frac{z-1}{z-e^{-0.5 \cdot 0.2}} + \frac{z-1}{z-e^{-0.2}} \\
&= 1 - 2\frac{z-1}{z-\underbrace{0.9048}_{\alpha}} + \frac{z-1}{z-\underbrace{0.8187}_{\beta}} \\
&= \frac{1(z-\alpha)(z-\beta) - 2(z-1)(z-\beta) + (z-1)(z-\alpha)}{(z-\alpha)(z-\beta)} \\
&= \frac{(-2\alpha + \beta + 1)z + (\alpha\beta - 2\beta + \alpha)}{z^2 - (\alpha + \beta)z + \alpha\beta} \\
&= \frac{0.0091z + 0.0082}{z^2 - 1.7235z + 0.7408} \tag{6}
\end{aligned}$$

Sabendo que a função de transferência de um sistema é definida como sendo a *Transformada de Laplace* da resposta ao impulso com condições iniciais nulas, e que seu equivalente em \mathcal{Z} possui as mesmas características, então:

$$\begin{aligned}
G(z) = \frac{Y(z)}{U(z)} &= \frac{0.0091z + 0.0082}{z^2 - 1.7235z + 0.7408} \\
Y(z)(z^2 - 1.7235z + 0.7408) &= U(z)(0.0091z + 0.0082) \\
z^2Y(z) - 1.7235zY(z) + 0.7408Y(z) &= 0.0091zU(z) + 0.0082U(z) \tag{7}
\end{aligned}$$

Aplicando a transformada \mathcal{Z} inversa na Eq. 7, tem-se:

$$\begin{aligned}
z^2Y(z) - 1.7235zY(z) + 0.7408Y(z) &= 0.0091zU(z) + 0.0082U(z) \xrightarrow{\mathcal{Z}^{-1}} \\
\stackrel{\mathcal{Z}^{-1}}{\implies} y(k+2) - 1.7235y(k+1) + 0.7408y(k) &= 0.0091u(k+1) + 0.0082u(k) \\
y(k) - 1.7235y(k-1) + 0.7408y(k-2) &= 0.0091u(k-1) + 0.0082u(k-2) \tag{8}
\end{aligned}$$

O resultado obtido na Eq. 7 pode ser confirmado fazendo uso da função `c2d` do Matlab[®], conforme *script* desenvolvido, mostrado no Apêndice A.1.

Item B

Para Santos (2007), a modelagem de um processo dinâmico consiste em obter um modelo matemático capaz de representar adequadamente suas características de interesse. Tal recurso se torna muitas vezes fundamental quando o processo real apresenta restrições de ordem operacional, econômico-financeira ou de segurança.

O modelo de um sistema pode ser obtido, basicamente, de duas formas: a partir das equações que descrevem sua dinâmica (*modelagem fenomenológica*) ou a partir da medição de dados de entrada e saída. Dentre os modelos que levam em consideração essa relação entrada/saída do sistema, existem os modelos **paramétricos** e os modelos **não paramétricos**.

Os modelos paramétricos correspondem àqueles que apresentam parâmetros característicos. Tais parâmetros são os coeficientes de uma equação a diferenças ou de uma função de transferência discreta do sistema. Já os modelos não paramétricos representam a dinâmica do

processo através dos coeficientes da resposta ao impulso (*modelos FIR – Finite Impulse Response*) ou da resposta ao degrau (*modelos ISR – Infinite Step Response e FSR – Finite Step Response*). [Santos 2007]

O modelo matemático baseado na resposta ao degrau representa o processo com um número infinito de termos que correspondem aos coeficientes da resposta ao degrau do sistema (ISR). Para sistemas estáveis, esses coeficientes tendem, assintoticamente, para um valor constante h_s , conforme Fig. 1.

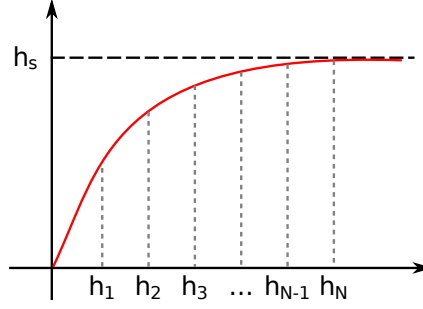


Figura 1: Exemplo de saída do sistema para o modelo ISR.

O modelo FSR corresponde ao modelo de resposta ao degrau limitando o número de coeficientes a um valor N . Tal modelo, também conhecido como modelo de resposta ao degrau truncada, tem fundamentação convolutiva para relacionar a entrada com a saída do sistema. Vale observar que nesse modelo o ruído não é levado em consideração. A saída do processo no modelo FSR é dada pela Eq. 9

$$y(k) = \sum_{i=1}^N h_i \Delta u(k-i) \quad (9)$$

Para o valor de $N = 10$ dado no enunciado da questão, tem-se

$$\begin{aligned} y(10) = & h_1 \Delta u(9) + h_2 \Delta u(8) + h_3 \Delta u(7) + h_4 \Delta u(6) + h_5 \Delta u(5) + \\ & + h_6 \Delta u(4) + h_7 \Delta u(3) + h_8 \Delta u(2) + h_9 \Delta u(1) + h_{10} \Delta u(0) \end{aligned}$$

Considerando $\hat{y}(k)$ como a saída dada pela Eq. 8, então:

$$\begin{aligned} y(10) = & \hat{y}(1)[u(9) - u(8)] + \hat{y}(2)[u(8) - u(7)] + \hat{y}(3)[u(7) - u(6)] + \\ & + \hat{y}(4)[u(6) - u(5)] + \hat{y}(5)[u(5) - u(4)] + \hat{y}(6)[u(4) - u(3)] + \\ & + \hat{y}(7)[u(3) - u(2)] + \hat{y}(8)[u(2) - u(1)] + \hat{y}(9)[u(1) - u(0)] + \\ & + \hat{y}(10)[u(0) - u(-1)] \end{aligned}$$

Para uma entrada degrau, $u(k) - u(k-1) = 0$, portanto:

$$y(10) = \hat{y}(10)u(0) = [1.7235y(9) - 0.7408y(8) + 0.0091u(9) + 0.0082u(8)]u(0) \quad (10)$$

Item C

Para efeitos comparativos foi utilizada uma entrada degrau, cujos resultados podem ser observados nas Figs. 2 a 5 para valores de $N = 10, 25, 50$ e 100 .

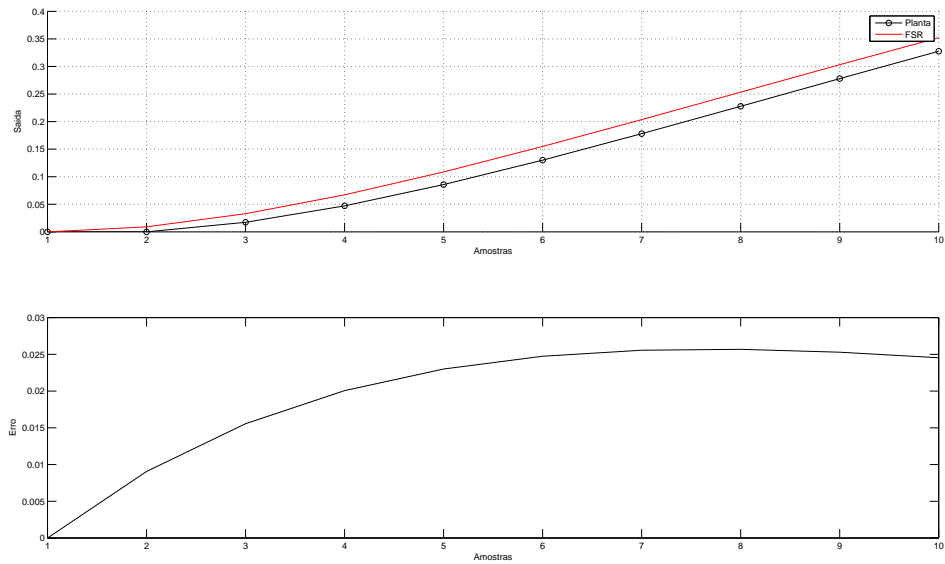


Figura 2: Comparação realizada com $N = 10$.

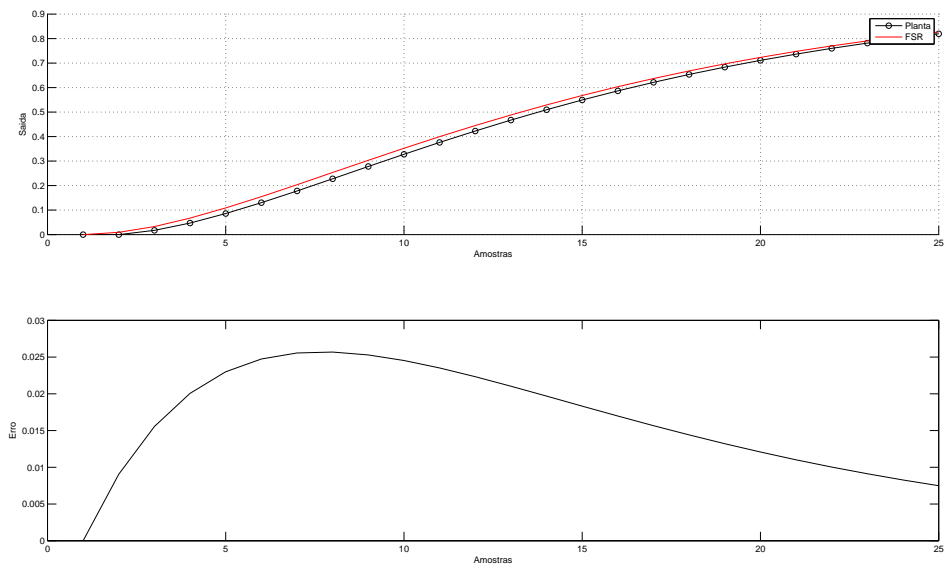


Figura 3: Comparação realizada com $N = 25$.

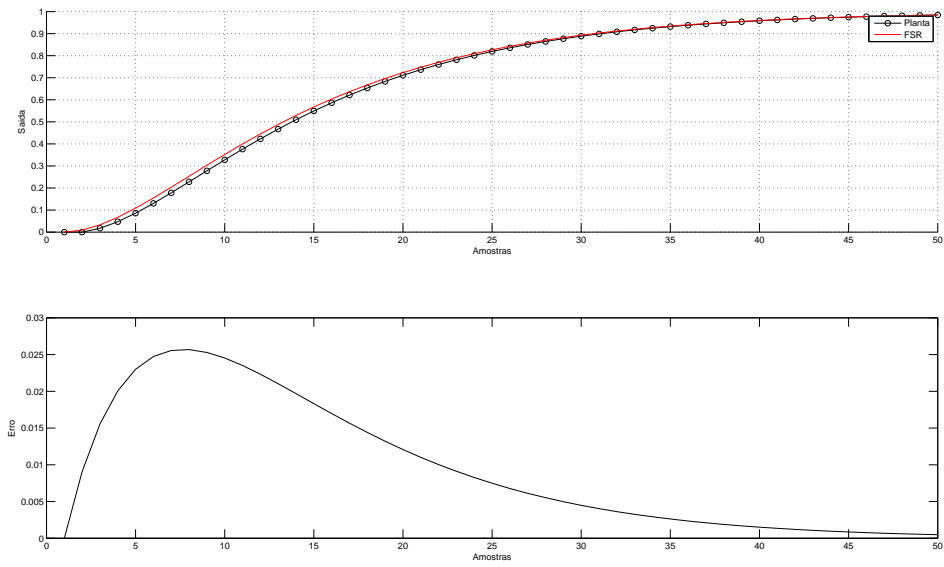


Figura 4: Comparação realizada com $N = 50$.

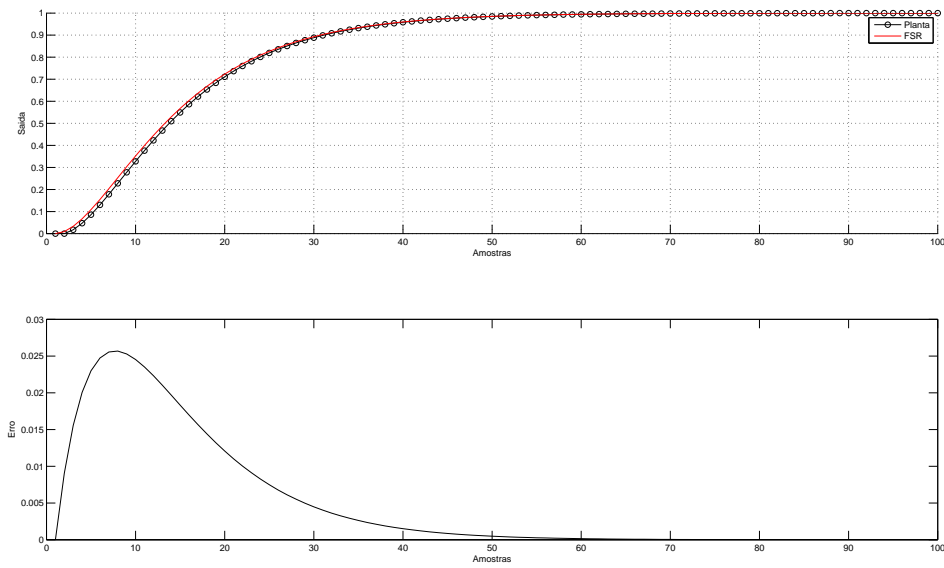


Figura 5: Comparação realizada com $N = 100$.

Observando-se as curvas de erro absoluto, percebe-se que para valores mais elevados de N , ou seja, para um número maior de coeficientes, o sistema tende a convergir mais rapidamente. O *script* do Matlab[®] desenvolvido para a resolução dessa questão pode ser encontrado no Apêndice A.2.

Item D

O controlador DMC (*Dynamic Matrix Controller*) foi desenvolvido no final da década de 70 por dois engenheiros da *Shell Oil Corporation*. O algoritmo, desenvolvido em 1979 de maneira não adaptativa e sem restrições, foi modificado em 1980 de maneira que permitisse a manipulação de restrições.

Anos depois foram propostas diversas modificações para esse algoritmo ao longo dos anos de maneira a melhorar suas características. Dentre elas, pode-se citar as implementações com tratamento de restrições e através de espaço de estados. O sucesso do algoritmo hoje se deve também aos produtos comerciais desenvolvidos que possuem funcionalidades como a identificação de modelos e a otimização global de plantas.

Esse controlador utiliza o modelo de *resposta ao degrau truncada* (FSR), fazendo uso de diversos parâmetros com o intuito de garantir uma boa qualidade de predição. O erro de predição do controlador é considerado como sendo um distúrbio atuante na saída.

Para o caso monovariável sem restrições, considera-se uma planta, estável em malha aberta, cuja resposta ao degrau pode ser descrita através de infinitos coeficientes:

$$y(k) = \sum_{i=1}^{\infty} h_i \Delta u(k-i) \quad (11)$$

Na prática, utiliza-se um modelo de resposta truncada, conforme Eq. 9, introduzindo-se a variável erro (usualmente inferior a 5%):

$$y(k) = \sum_{i=1}^N h_i \Delta u(k-i) + e(k) \quad (12)$$

Baseado no modelo descrito pela Eq. 12, calcula-se um valor aproximado para $y(k)$ que será diferente da saída real devido ao erro de modelagem e a um possível ruído a ser inserido no processo de medição:

$$\tilde{y}(k) \approx \sum_{i=1}^N h_i \Delta u(k-i) \quad (13)$$

O erro inerente ao truncamento e ao ruído de medição pode ser calculado da seguinte forma:

$$e(k) = y_m(k) - \tilde{y}(k) = y_m(k) - \sum_{i=1}^N h_i \Delta u(k-i) \quad (14)$$

em que $y_m(k)$ corresponde ao valor que deveria ser medido na saída no instante k . Assim sendo, para o instante $k+j$, tem-se:

$$\begin{aligned} \tilde{y}(k+j) &= \sum_{i=1}^N h_i \Delta u(k+j-i) + \tilde{e}(k+j) \\ \tilde{y}(k+j) &= \sum_{i=1}^j h_i \Delta u(k+j-i) + \sum_{i=j+1}^N h_i \Delta u(k+j-i) + \tilde{e}(k+j) \end{aligned}$$

Uma vez que não é possível estimar o valor futuro do erro de predição a partir do modelo, intrinsecamente determinístico, o algoritmo supõe que os erros das j -ésimas predições à frente serão iguais ao erro calculado no instante atual:

$$\tilde{y}(k+j) = \sum_{i=1}^j h_i \Delta u(k+j-i) + \sum_{i=j+1}^N h_i \Delta u(k+j-i) + e(k) \quad (15)$$

Substituindo a Eq. 14 na Eq. 15, tem-se:

$$\tilde{y}(k+j) = \underbrace{\sum_{i=1}^j h_i \Delta u(k+j-i)}_{\text{parcela forçada}} + \underbrace{\sum_{i=j+1}^N h_i \Delta u(k+j-i) + y_m(k) - \sum_{i=1}^N h_i \Delta u(k-i)}_{\text{parcela livre}} \quad (16)$$

O primeiro termo, que corresponde a *parcela forçada* (\tilde{y}_f), pode ser obtido considerando-se condições iniciais nulas (em k) e a sequência das ações de controle futuras. Os dois outros termos, correspondentes a *parcela livre* (\tilde{y}_l), são obtidos a partir da resposta natural do sistema, com as condições atuais, considerando a sequência de ações de controle futuras iguais a zero.

Para a resposta livre, tem-se que:

$$\tilde{y}_l(k+j) = \sum_{i=j+1}^N h_i \Delta u(k+j-i) + y_m(k) - \sum_{i=1}^N h_i \Delta u(k-i)$$

Rearranjando os termos quando $N \rightarrow \infty$, tem-se:

$$\tilde{y}_l(k+j) = y_m(k) + \sum_{i=1}^{\infty} (h_{j+i} - h_i) \Delta u(k-i) \quad (17)$$

Se o sistema for assintoticamente estável, após N períodos de amostragem, tem-se:

$$h_{j+i} - h_i \approx 0$$

Expandindo então a Eq. 16, tem-se:

$$\begin{aligned} \tilde{y}(k+1) &= h_1 \Delta u(k) + h_2 \Delta u(k-1) + \dots + h_N \Delta u(k-N+1) + e(k) \\ \tilde{y}(k+2) &= h_1 \Delta u(k+1) + h_2 \Delta u(k) + \dots + h_N \Delta u(k-N+2) + e(k) \\ &\vdots \\ \tilde{y}(k+NY) &= h_1 \Delta u(k+NY-1) + h_2 \Delta u(k+NY-2) + \dots + h_N \Delta u(k-N+NY) + e(k) \end{aligned}$$

Na qual a resposta livre corresponde as parcelas:

$$\begin{aligned} \tilde{y}_l(k+1) &= h_2 \Delta u(k-1) + h_3 \Delta u(k-2) + \dots + h_N \Delta u(k-N+1) + e(k) \\ \tilde{y}_l(k+1) &= h_3 \Delta u(k-1) + h_4 \Delta u(k-2) + \dots + h_N \Delta u(k-N+2) + e(k) \\ &\vdots \\ \tilde{y}_l(k+NY) &= h_{NY+1} \Delta u(k-1) + h_{NY+2} \Delta u(k-2) + \dots + h_N \Delta u(k-N+NY) + e(k) \end{aligned}$$

Assim, matricialmente, tem-se:

$$\tilde{\mathbf{Y}}_l = \begin{bmatrix} h_N & h_{N-1} & \dots & h_2 \\ 0 & h_N & \dots & h_3 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & h_N \end{bmatrix} \begin{bmatrix} \Delta u(k-N+1) \\ \vdots \\ \Delta u(k-2) \\ \Delta u(k-1) \end{bmatrix} + \begin{bmatrix} e(k) \\ e(k) \\ \vdots \\ e(k) \end{bmatrix} \quad (18)$$

Para a resposta forçada, observa-se que, por definição, $\Delta u(k+i) = 0$, para $i \geq NU$ ou $i < 0$, sendo NU o horizonte de controle, ou seja, ao número de incrementos futuros a serem fornecidos à ação de controle atual. Assim, verifica-se que:

$$\begin{aligned}\tilde{y}_f(k+1) &= h_1 \Delta u(k) \\ \tilde{y}_f(k+2) &= h_1 \Delta u(k+1) + h_2 \Delta u(k) \\ &\vdots \\ \tilde{y}_f(k+NU) &= h_1 \Delta u(k+NU-1) + \dots + h_{NU-1} \Delta u(k) \\ &\vdots \\ \tilde{y}_f(k+NY) &= h_{NY-NU} \Delta u(k+NU) + \dots + h_{NY} \Delta u(k)\end{aligned}$$

Matricialmente, tem-se:

$$\tilde{\mathbf{Y}}_f = \begin{bmatrix} h_1 & 0 & \dots & 0 \\ h_2 & h_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h_{NY} & h_{NY-1} & \dots & h_1 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+NU) \end{bmatrix} \Rightarrow \tilde{\mathbf{Y}}_f = \mathbf{H} \Delta \mathbf{U} \quad (19)$$

A matriz \mathbf{H} é conhecida como matriz dinâmica, a qual dá origem ao nome do algoritmo. O valor pretendido é então dado por:

$$\tilde{\mathbf{Y}} = \tilde{\mathbf{Y}}_l + \tilde{\mathbf{Y}}_f \tilde{\mathbf{Y}} = \tilde{\mathbf{Y}}_l + \mathbf{H} \Delta \mathbf{U} \quad (20)$$

Vale observar que a matriz \mathbf{H} é constituída de NU colunas obtidas a partir da resposta ao degrau do sistema, apropriadamente escolhidas em ordem decrescente.

Os próximos incrementos da ação de controle são obtidos através do seguinte critério de minimização:

$$J = \sum_{j=1}^{NY} [\hat{y}(k+j) - r(k-j)]^2 + \sum_{j=1}^{NU} [\lambda \Delta u(k+j)]^2$$

Uma das vantagens da utilização desse critério é que a variação de controle é considerada, evitando J crescer indefinidamente. Fazendo uso da notação matricial, tem-se:

$$\begin{aligned}J &= (\tilde{\mathbf{Y}} - \mathbf{R})^T (\tilde{\mathbf{Y}} - \mathbf{R}) + \lambda \Delta \mathbf{U}^T \Delta \mathbf{U} \\ &= (\tilde{\mathbf{Y}}_l + \mathbf{H} \Delta \mathbf{U} - \mathbf{R})^T (\tilde{\mathbf{Y}}_l + \mathbf{H} \Delta \mathbf{U} - \mathbf{R}) + \lambda \Delta \mathbf{U}^T \Delta \mathbf{U}\end{aligned}$$

Fazendo $\frac{\partial J}{\partial \Delta \mathbf{U}} = 0$, tem-se:

$$\frac{\partial J}{\partial \Delta \mathbf{U}} = 2\mathbf{H}^T \mathbf{H} \Delta \mathbf{U} + 2\mathbf{H}^T (\tilde{\mathbf{Y}}_l - \mathbf{R}) + 2\lambda \Delta \mathbf{U} = 0$$

O que leva a:

$$\Delta \mathbf{U} = (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{I})^{-1} \mathbf{H}^T (\mathbf{R} - \tilde{\mathbf{Y}}_l) \quad (21)$$

Item E

Questão 2

Implemente o método dos mínimos quadrados offline e teste o programa com o processo da questão 1, simulando variações lentas em seus parâmetros ($\pm 10\%$) e avaliando o comportamento das estimativas.

Apresente gráficos mostrando as estimativas, os valores reais dos parâmetros, o sinal de controle, a saída real e a estimada com cada modelo utilizado.

Resolução:

O sistema da questão 1 consiste na função de transferência dada pela Eq. 22:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{0.5}{(s+0.5)(s+1)} \quad (22)$$

A discretização obtida nessa mesma questão é repetida pela Eq. 23,

$$y(k) - 1.7235y(k-1) + 0.7408y(k-2) = 0.0091u(k-1) + 0.0082u(k-2) \quad (23)$$

Generalizando o modelo obtido na Eq. 23 e isolando o termo $y(k)$, tem-se que:

$$\begin{aligned} y(k) + a_1y(k-1) + a_2y(k-2) &= b_1u(k-1) + b_2u(k-2) \\ y(k) &= -a_1y(k-1) - a_2y(k-2) + b_1u(k-1) + b_2u(k-2) \end{aligned} \quad (24)$$

O estimador de mínimos quadrados não-recursivo considera um sistema representado por uma equação a diferenças semelhante a Eq. 24. Segundo Coelho e Coelho (2004), haverá $n_a + n_b + 1$ parâmetros a serem estimados pelo algoritmo e para determinar os valores de a_i , deve-se utilizar as medidas de entrada e saída do processo.

Assim sendo, de posse da equação generalizada dos parâmetros do sistema, utilizar o método dos mínimos quadrados *offline* consiste em excitar a planta com um determinado sinal de entrada e armazenar os valores de saída obtidos para executar o algoritmo não-recursivo. Assim sendo, pode-se dizer que o processo de identificação é realizado de uma só vez, ou em batelada.

Supondo que exista um sistema discreto que possa ser descrito conforme *modelo de regressão linear*:

$$\mathbf{Y} = \mathbf{X}\theta + \mathbf{e} \quad (25)$$

tal que \mathbf{Y} corresponde à saída do sistema, \mathbf{X} é um vetor determinístico conhecido, θ é o vetor de parâmetros a serem estimados e \mathbf{e} corresponde ao erro do modelo. Deseja-se então, estimar o vetor θ a partir de N experimentos, de tal forma que:

$$\begin{aligned}
\mathbf{Y}_1 &= \mathbf{X}_1\theta_1 + \mathbf{e}_1 \\
\mathbf{Y}_2 &= \mathbf{X}_2\theta_2 + \mathbf{e}_2 \\
&\vdots \\
\mathbf{Y}_N &= \mathbf{X}_N\theta_N + \mathbf{e}_N
\end{aligned}$$

considerando:

$$\theta = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \\ b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} -Y(1) & -Y(0) & u(1) & u(0) \\ -Y(2) & -Y(1) & u(2) & u(1) \\ \vdots & \vdots & \vdots & \vdots \\ -Y(N-1) & -Y(N-n) & u(N-1) & u(N-m) \end{bmatrix}$$

$$\mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \quad \mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}$$

O método dos mínimos quadrados tem por objetivo realizar a estimativa de θ de modo a minimizar a função de erro J , tal que:

$$J = \sum_{K=1}^N e^2(K) = \mathbf{e}^T \mathbf{e} \quad (26)$$

Pela Eq. 25:

$$\mathbf{Y} = \mathbf{X}\theta + \mathbf{e} \implies \mathbf{e} = \mathbf{Y} - \mathbf{X}\theta \quad (27)$$

Assim, a função de erro J , pode ser reescrita como:

$$J = (\mathbf{Y} - \mathbf{X}\theta)^T (\mathbf{Y} - \mathbf{X}\theta) \quad (28)$$

que possui mínimo quando:

$$\left. \frac{\partial J}{\partial \theta} \right|_{\theta=\hat{\theta}} = 0$$

Considerando que:

$$\frac{\partial(\mathbf{A}\mathbf{x} + \mathbf{b})^T \mathbf{C}(\mathbf{D}\mathbf{x} + \mathbf{e})}{\partial \mathbf{x}} = (\mathbf{D}\mathbf{x} + \mathbf{e})^T \mathbf{C}^T \mathbf{A} + (\mathbf{A}\mathbf{x} + \mathbf{b})^T \mathbf{C} \mathbf{D}$$

e fazendo:

$$\begin{aligned}
\mathbf{A} &= -\mathbf{X} & \mathbf{x} &= \theta & \mathbf{b} &= \mathbf{Y} \\
\mathbf{C} &= \mathbf{I} & \mathbf{D} &= -\mathbf{X} & \mathbf{e} &= \mathbf{Y}
\end{aligned}$$

na Eq. 28, tem-se que:

$$\begin{aligned}
\frac{\partial J}{\partial \theta} &= (-\mathbf{X}\theta + \mathbf{Y})^T(-\mathbf{X}) + (-\mathbf{X}\theta + \mathbf{Y})^T(-\mathbf{X}) \\
&= -2[(-\mathbf{X}\theta + \mathbf{Y})]^T \mathbf{X} \\
&= -2[(-\mathbf{X}\theta)^T + \mathbf{Y}^T] \mathbf{X} \\
&= -2[-\theta^T \mathbf{X}^T + \mathbf{Y}^T] \mathbf{X} \\
&= 2\theta^T \mathbf{X}^T \mathbf{X} - 2\mathbf{Y}^T \mathbf{X}
\end{aligned} \tag{29}$$

Como $\frac{\partial J}{\partial \theta} = 0$ (escalar), então, pode-se transpor a Eq. 29, de tal forma que:

$$\begin{aligned}
\frac{\partial J}{\partial \theta} &= 2\mathbf{X}^T(\theta^T \mathbf{X}^T)^T - 2\mathbf{X}^T \mathbf{Y} \\
0 &= 2\mathbf{X}^T(\mathbf{X}\theta) - 2\mathbf{X}^T \mathbf{Y} \\
0 &= 2\mathbf{X}^T \mathbf{X}\theta - 2\mathbf{X}^T \mathbf{Y}
\end{aligned} \tag{30}$$

Substituindo θ por $\hat{\theta}$ e isolando $\hat{\theta}$, tem-se que:

$$\hat{\theta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} \tag{31}$$

Algumas observações podem ser feitas acerca da Eq. 31:

- A solução existirá quando $(\mathbf{X}^T \mathbf{X})^{-1}$ for não singular
- A sequência escolhida das entradas $u(k)$ deverá garantir essa não singularidade
- Quando não há a presença de ruídos $\hat{\theta}$ pode ser encontrado em $n + m$ passos
- A matriz \mathbf{X} cresce na medida em que N cresce

O algoritmo desenvolvido leva em consideração uma variação dos parâmetros aleatória para $\pm 10\%$ do valor do parâmetro a ser modificado. O número de variações também é aleatório, correspondendo a, no máximo, 5% do número de amostras. Ou seja, se forem consideradas 100 amostras, haverão, no máximo, $5 - 1 = 4$ variações de parâmetros equiespaçadas, uma vez que da primeira amostra até a primeira variação não deve ser contabilizada. A variação dos parâmetros é mantida até que um novo valor seja sorteado.

O sinal de entrada do sistema varia de maneira análoga à variação dos parâmetros da função, ou seja, a cada K iterações o sinal muda e se mantém naquele valor até o próximo instante de variação. Para compor esse sinal, o número máximo de variações correspondia a 50% do número de amostras. O valor da entrada $u(k)$ variava entre $\pm 50\% u(k-1)$ nos instantes de variação e era repetido para as demais amostras.

No primeiro teste, foram consideradas 100 amostras. O número de variações sorteado foi de 2 variações. Portanto, cada variação era mantida por 33 amostras. A saída do sistema com os parâmetros reais, bem como com os parâmetros variáveis e o sinal de controle pode ser observada na Fig. 6. A estimativa realizada é vista na Tab. 1.¹

¹Por questões de precisão, os valores utilizados em todos os testes foram aqueles obtidos através do *script* do MATLAB[®] mostrado no Apêndice A.1.

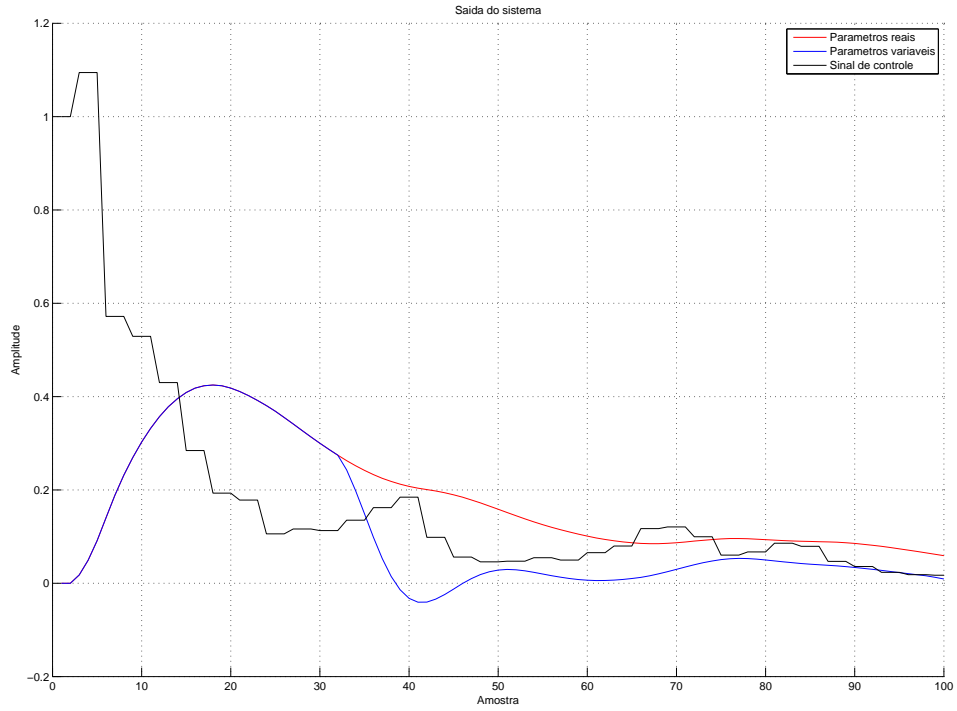


Figura 6: Saída do sistema para $N = 100$.

Tabela 1: Estimativa realizada para $N = 100$.

Parâmetros	θ_{ideal}	$\theta_{\text{sem variação}}$	$\theta_{\text{com variação}}$
a_1	-1.724	-1.7240	-1.8617
a_2	0.7408	0.7408	0.8807
b_1	0.009056	0.0096	0.0103
b_2	0.008194	0.0082	0.0032

Devido a aleatoriedade envolvida no algoritmo, percebeu-se que quando há um grande número de variações dos parâmetros ou quando a variação corresponde a um valor elevado, o sistema se torna instável e sua saída passa a divergir, o que leva a estimativas erradas dos parâmetros, como pode ser verificado na Fig. 7 e na Tab. 2. Esses resultados foram obtidos para 100 amostras e, coincidentemente, para 2 variações.

O algoritmo também foi executado para 1000 amostras. Entranto, devido a dificuldade de se encontrar valores de saída que não divergissem para o valor máximo de 5% do número de amostras de variações, ou seja, $50 - 1 = 49$ variações, essa porcentagem foi reduzida à 1%. Os resultados obtidos para a primeira execução podem ser observados na Fig. 8 e na Tab. 3. Para esse resultado, houve apenas 1 variação dos parâmetros.

Outra execução resultou em resultados semelhantes, como pode ser observado na Fig. 9 e na Tab. 4

Todos os valores aleatórios obtidos nas simulações seguiam uma distribuição uniforme. O *script* do Matlab[®] desenvolvido para a resolução dessa questão pode ser encontrado no Apêndice B.

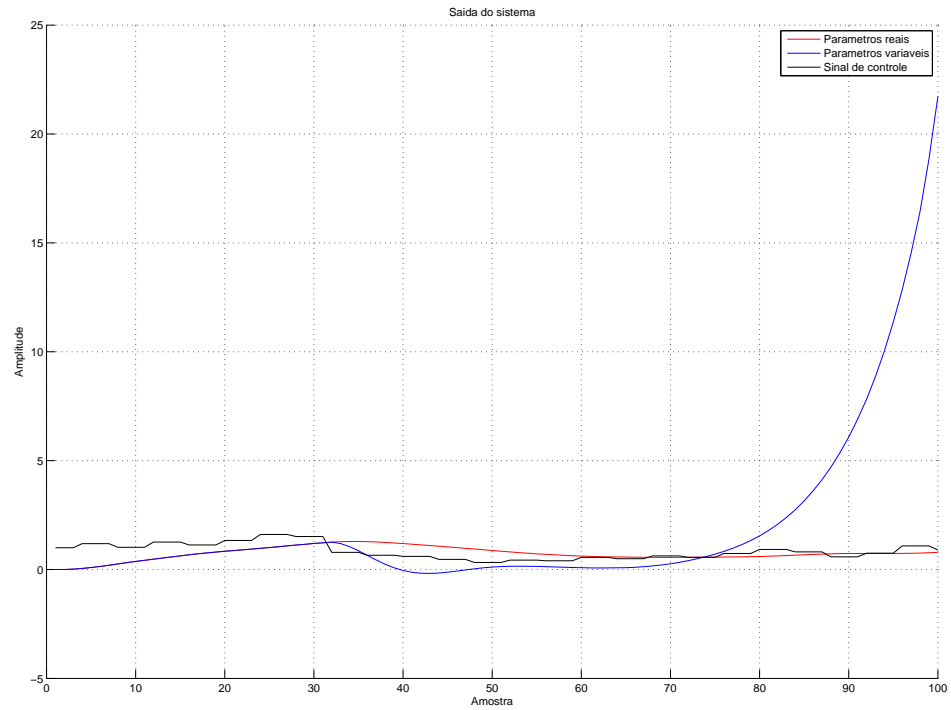


Figura 7: Saída do sistema para $N = 100$ – Divergente.

Tabela 2: Estimativa realizada para $N = 100$ – Divergente.

Parâmetros	θ_{ideal}	$\theta_{\text{sem variação}}$	$\theta_{\text{com variação}}$
a_1	-1.724	-1.7240	-2,1177
a_2	0.7408	0.7408	1.1105
b_1	0.009056	0.0096	0.0400
b_2	0.008194	0.0082	-0.0463

Tabela 3: Estimativa realizada para $N = 1000$.

Parâmetros	θ_{ideal}	$\theta_{\text{sem variação}}$	$\theta_{\text{com variação}}$
a_1	-1.724	-1.7240	-1.9613
a_2	0.7408	0.7408	0.9641
b_1	0.009056	0.0096	0.0100
b_2	0.008194	0.0082	-0.0073

Tabela 4: Estimativa realizada para $N = 1000$.

Parâmetros	θ_{ideal}	$\theta_{\text{sem variação}}$	$\theta_{\text{com variação}}$
a_1	-1.724	-1.7240	-1.9661
a_2	0.7408	0.7408	0.9680
b_1	0.009056	0.0096	0.0093
b_2	0.008194	0.0082	-0.0076

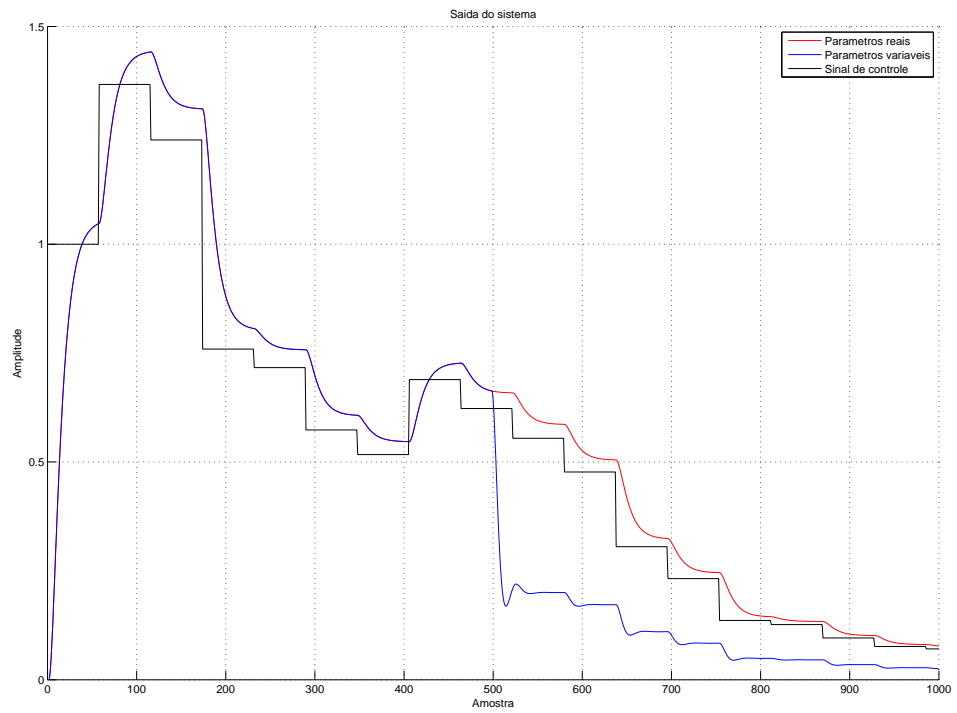


Figura 8: Saída do sistema para $N = 1000$

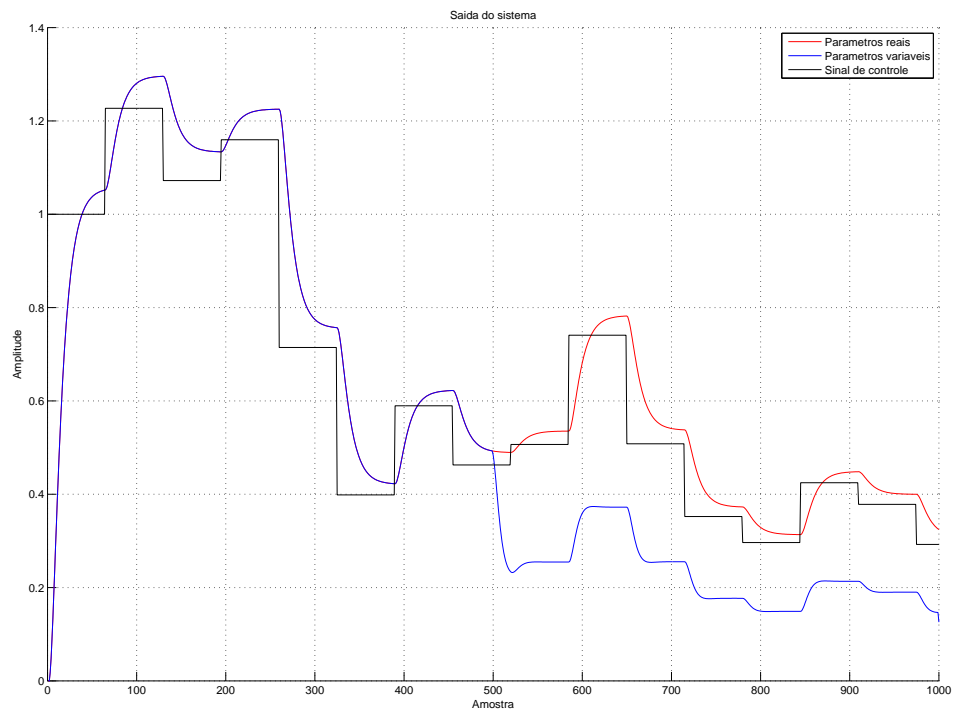


Figura 9: Saída do sistema para $N = 1000$

Apêndice A

Código da Questão 1

A.1 Item A

```
1 % Universidade Federal do Rio Grande do Norte
  % Programa de Pos-Graduacao em Engenharia Eletrica e de Computacao
3 % Lista 2 - Questao 1 - A
  % Autores: Anna Giselle Camara Dantas Ribeiro
5 %           Cristiano Gurgel de Castro
  %           Diogo Leite Reboucas
7 %           Thiago Medeiros Barros

9 clear ;
  clc ;
11
  % Funcao de transferencia do sistema em Laplace
13 Gs = tf( [0.5], [1 1.5 0.5] );

15 % Periodo de amostragem
  T = 0.2;
17
  % Metodo de aproximacao
19 metodo = 'zoh';

21 % Funcao de transferencia do sistema em Zadeh
  Gz = c2d( Gs, T, metodo )
```

A.2 Item C

```
  % Universidade Federal do Rio Grande do Norte
2 % Programa de Pos-Graduacao em Engenharia Eletrica e de Computacao
  % Lista 2 - Questao 1 - C
4 % Autores: Anna Giselle Camara Dantas Ribeiro
  %           Cristiano Gurgel de Castro
6 %           Diogo Leite Reboucas
  %           Thiago Medeiros Barros
8
  clear all;
10 close all;
  clc ;
12
  % Funcao de transferencia do sistema em Laplace
14 Gs = tf( [0.5], [1 1.5 0.5] );

16 % Periodo de amostragem
  T = 0.2;
18
  % Metodo de aproximacao
20 metodo = 'zoh';

22 % Funcao de transferencia do sistema em Zadeh
  Gz = c2d( Gs, T, metodo );
24
```

```

% Numero de coeficientes
26 N = 25;

28 % Saida do sistema para a resposta ao degrau
saida = step( Gz, N );
30
% Coeficientes (Valores dos H's)
32 h = saida( 2:N );

34 % Os valores de delta_u existiraõ somente para a primeiro coeficiente 'h',
% pois a diferenca  $u(k) - u(k-1) = 0$  para os demais coeficientes, conforme
36 % explicado no texto
delta_u = [ 1 ; zeros( N - 1, 1 ) ];
38
y = zeros( N, 1 );
40
for k = 1 : N
42     soma = 0;

44     for i = 1 : N
46         if ( k - i ) > 0
48             soma = soma + h(i) * delta_u( k - i );
50         else
52             break;
54         end
56     end

58     y(k) = soma;
60 end

% Entrada degrau unitario
u = ones( N, 1 );

62 % Determinando a saida de acordo com:
%  $y_{\text{chapeu}}(k) = 1.7235*y_{\text{chapeu}}(k-1) - 0.7408*y_{\text{chapeu}}(k-2) +$ 
%  $0.0091*u(k-1) + 0.0082*u(k-2)$ 
y_chapeu = zeros( N, 1 );
62 y_chapeu( 1 ) = 0;
y_chapeu( 2 ) = 0;
64
for k = 3 : N
66     y_chapeu(k) = 1.7235*y_chapeu(k-1) - 0.7408*y_chapeu(k-2) + ...
68         0.0091*u(k-1) + 0.0082*u(k-2);
70 end

figure;

72 subplot( 2, 1, 1 );
grid on;
74 hold on;
plot( y_chapeu, 'ko-' );
76 plot( y, 'r' );
legend( 'Planta', 'FSR' );
78 ylabel( 'Saida' );
xlabel( 'Amostras' );
80
subplot( 2, 1, 2 );
82 plot( abs( y_chapeu - y ), 'k' );
ylabel( 'Erro' );
84 xlabel( 'Amostras' );

```

Apêndice B

Código da Questão 2

```
% Universidade Federal do Rio Grande do Norte
2 % Programa de Pós-Graduação em Engenharia Elétrica e de Computação
% Lista 2 – Questão 2
4 % Autores: Anna Giselle Camara Dantas Ribeiro
%           Cristiano Gurgel de Castro
6 %           Diogo Leite Rebouças
%           Thiago Medeiros Barros
8
9 clear ;
10 clc ;

12 % Numero de amostras
N = 1000;

14 % Definindo os vetores de entrada/saída
16 y = zeros( N, 1 );
   y_est = zeros( N, 1 );
18 u = zeros( N, 1 );

20 % Condições iniciais
   y(1) = 0;
22 y(2) = 0;

24 y_est(1) = 0;
   y_est(2) = 0;
26
28 u(1) = 1;
   u(2) = 1;

30 % Parametros
   a1 = -1.724;
32 a2 = 0.7408;
   n = 2;
34
36 b1 = 0.00956;
   b2 = 0.008194;
   m = 2;
38

39 % Parametros com variação
40 a1_tmp = a1;
   a2_tmp = a2;
42 b1_tmp = b1;
   b2_tmp = b2;
44

45 % Numero de vezes em que o sinal de controle irá variar (Até 50% do numero
46 % de amostras)
   num_var_sinal_cont = fix( random( 1, 0.5*N ) );
48 amostras_por_var_sinal_cont = fix( random( 1, 0.1*N ) );

50 % Valores de entrada
for k = 3 : N
52     % Novos valores de entrada
       if mod( k, amostras_por_var_sinal_cont ) == 0
54         u(k) = u(k-1) * random( 0.5, 1.5 );
```

```

56         else
            u(k) = u(k-1);
58     end

60     % Numero de vezes em que os parametros irao variar (Ate 5% do numero de
62     % amostras)
    num_variacoes = fix( random( 1, N * 0.01 ) );
64     amostras_por_variacao = fix( N / num_variacoes );

66     % Valores de saida
    for k = 3 : N
68         % Saida real
            y(k) = -a1*y(k-1) - a2*y(k-2) + b1*u(k-1) + b2*u(k-2);

70         % Saida com variacao dos parametros
72         if mod( k, amostras_por_variacao ) == 0
            a1_tmp = a1 * random( 0.9, 1.1 );
74             a2_tmp = a2 * random( 0.9, 1.1 );
            b1_tmp = b1 * random( 0.9, 1.1 );
76             b2_tmp = b2 * random( 0.9, 1.1 );
        end

78         y_est(k) = -a1_tmp*y_est(k-1) - a2_tmp*y_est(k-2) + ...
80             b1_tmp*u(k-1) + b2_tmp*u(k-2);
    end

82     % Matriz X
84     X = [-y((2:N-1), 1) -y((1:N-n), 1) u((2:N-1), 1) u((1:N-m), 1)];
    X_est = [ -y_est((2:N-1), 1) -y_est((1:N-n), 1) ...
86         u((2:N-1), 1) u((1:N-m), 1)];

88     % Estimacao de theta
    theta_est_real = inv( X'*X ) * X' * y((3:N), 1);
90     theta_est_variacao = inv( X_est'*X_est ) * X_est' * y_est((3:N), 1);

92     % Exibindo as saidas
    figure;
94     grid on;
    hold on;
96     plot( y, 'r' );
    plot( y_est, 'b' );
98     plot( u, 'k' );
    ylabel( 'Amplitude' );
100    xlabel( 'Amostra' );
    title( 'Saida do sistema' );
102    legend( 'Parametros reais', 'Parametros variaveis', 'Sinal de controle' );

104    disp( 'Numero de variacoes:' )
    disp( num_variacoes - 1 ); % Da primeira amostra ate a primeira variacao
106    % nao deve ser contabilizado
    disp( 'Numero de amostras por variacao:' )
108    disp( amostras_por_variacao );
    disp( 'Theta estimado com parametros reais' );
110    disp( theta_est_real );
    disp( 'Theta estimado com parametros reais' );
112    disp( theta_est_variacao );

```

Referências Bibliográficas

Coelho, Antônio Augusto Rodrigues e Leandro Dos Santos Coelho (2004), *Identificação de Sistemas Dinâmicos Lineares*, UFSC – 1ª Edição.

Santos, José Eli Santos Dos (2007), Controle preditivo não-linear para sistemas de hammerstein, Tese de doutorado, Universidade Federal de Santa Catarina - UFSC.