

**UNIVERSIDADE FEDERAL DE ALAGOAS
INSTITUTO DE COMPUTAÇÃO
CURSO DE ENGENHARIA DE COMPUTAÇÃO**

**RELATÓRIO DE EXPERIÊNCIA
Modelagem de Métodos de Identificação de Sistemas**

Israel L. C. Vasconcelos: 15110761
Ricardo A. F. O. Lopes: 12111638

Maceió - AL
2018

Israel L. C. Vasconcelos
Ricardo A. F. O. Lopes

MÉTODOS DETERMINÍSTICOS DE IDENTIFICAÇÃO DE SISTEMAS

Primeiro Relatório Parcial apresentado à disciplina de Introdução à identificação de sistemas dinâmicos, correspondente à avaliação do semestre 2017.2 do 10º período do curso de Engenharia de Computação da Universidade Federal de Alagoas, sob orientação do **Prof. Ícaro Bezerra Queiroz de Araújo.**

Maceió - AL
2018

RESUMO

Este trabalho apresenta um estudo contendo fundamentação, seguido de implementação e validação das técnicas tradicionais empregadas na identificação de curvas características de sistemas dinâmicos. Utilizamos os métodos clássicos para estimação de parâmetros dos sistemas propostos e a consequente identificação dos mesmos. São reportados os desafios na experimentação, bem como os resultados obtidos através de gráficos e análise numérica, inerentes aos problemas escopo estudado.

Palavras-chave: Identificação de sistemas, Processamento de sinais, Sistemas de controle.

Lista de Figuras

1	Comportamento do Sistema 1.	13
2	Comportamento do Sistema 2.	14
3	Comportamento do Sistema 3.	14
4	Comportamento do Sistema 4.	15
5	Avaliação utilizando Erro Quadrático Médio.	15
6	Avaliação utilizando Erro Quadrático Médio.	16
7	Estimativa utilizando Yuwana Seborg.	17

Sumário

1	INTRODUÇÃO	6
2	FUNDAMENTAÇÃO TEÓRICA	7
2.1	Ziegler & Nichols	7
2.2	Hagglund	7
2.3	Smith (1ª e 2ª Ordem)	7
2.4	Sundaresan & Krishnaswamy	8
2.5	Mollenkamp	8
2.6	Método Yuwana-Seborg	9
3	METODOLOGIA	11
3.1	Software e Ferramentas	11
3.2	Recursos Implementados	11
4	RESULTADOS E DISCUSSÕES	13
4.1	Modelagem de Métodos de Identificação de Sistemas	13
4.1.1	Dataset 1	13
4.1.2	Datasets 2 e 3	14
4.1.3	Dataset 4	15
4.2	Avaliação quantitativa: Erro Quadrático Médio	15
4.3	Métodos determinísticos para sistemas em malha fechada	16
4.3.1	Comportamento dos Sistemas	17
4.3.2	Informações Complementares	18
5	CONSIDERAÇÕES FINAIS	22
	REFERÊNCIAS	23

1 INTRODUÇÃO

Este relatório apresenta um estudo contendo fundamentação, seguido de implementação e validação das técnicas tradicionais empregadas na identificação de curvas características de sistemas dinâmicos.

São discutidos seis métodos para identificação de sistemas: Ziegler-Nichols, Hagglund, Smith, Sundaresan-Krishnaswamy, Mollenkamp e Yuwana Seborg. A validação desses métodos contempla quatro conjuntos de teste diferentes onde é avaliado o comportamento dos algoritmos. Etapas preliminares de calibração manual e eliminação de ruído foram praticadas para refinamento dos resultados.

Foi utilizada a linguagem R na versão 3.4.4 [R Development Core Team, 2010], sistema operacional Ubuntu 18.04 e a *toolbox* **CONTROL** para modelagem dos sistemas e implementação das soluções, além da biblioteca **METRICS** para análise numérica dos resultados, utilizando como métrica o Erro Quadrático Médio.

A seção 2 apresenta detalhamentos acerca da fundamentação teórica das técnicas empregadas. A seção 3 discute os passos seguidos para geração dos resultados e informações técnicas relevantes. A seção 4 traz os gráficos contendo os resultados obtidos nos experimentos. A seção 5 comenta as conclusões, considerações finais e perspectivas futuras deste trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Ziegler & Nichols

O método de Ziegler-Nichols [Ziegler and Nichols, 1942] é uma ferramenta heurística para para estimação de parâmetros e calibração de sistemas dinâmicos submetidos à atuação de controladores.

Consiste em detectar as derivadas de primeira e segunda ordem no comportamento da resposta do sistema sob investigação, buscando em seguida os respectivos instantes de tempo correspondentes aos zeros da função.

Para realizar a estimação, fazemos $\theta = t_1$ e $\tau = t_3 - t_1$ para alimentar o modelo descrito na equação 1.

$$G_p(s) = e^{-s\theta} \frac{K}{s\tau + 1} \quad (1)$$

2.2 Hagglund

O método de Hagglund [Aguirre, 2007] utiliza o mesmo modelo anteriormente visto na equação 1, entretanto, a forma como são definidos os instantes de tempo utiliza um princípio heurístico diferente.

Dessa forma, o valor tomado na definição de $\tau = t_2 - t_1$, onde o instante de tempo no qual t_2 corresponde a $A_1 = A_{max} \times 0.632$ sendo A_1 a amplitude relativa ao valor máximo, A_{max} .

2.3 Smith (1ª e 2ª Ordem)

O método de Smith [Aguirre, 2007] consiste na escolha de dois instantes de tempo na função que define a resposta do sistema, para modelos de primeira ordem, temos

$Y(t_1) = y_i + 0.283y_f$ e $Y(t_2) = y_i + 0.632y_f$. Em seguida, calculamos: $K = \frac{\Delta_y}{\Delta_u}$ para $\Delta_y = y_f - y_i$ e $\Delta_u = u_f - u_i$.

Dessa forma, podemos obter em função destes, os parâmetros τ e θ através das equações 2 e 3 abaixo:

$$\tau = \frac{3}{2}(t_2 - t_1) \quad (2)$$

$$\theta = t_2 - \tau \quad (3)$$

Por outro lado, a heurística empregada no método de Smith para sistemas de 2ª ordem define os intervalos de tempo onde $A_1 = A_{max} \times 0.2$ e $A_2 = A_{max} \times 0.6$, sendo A_1 e A_2 amplitudes relativas ao valor máximo, A_{max} , obtido no sistema modelado para segunda ordem.

$$G_p(s) = \frac{Ke^{-s\theta}}{\tau^2 s^2 + 2s\zeta\tau + 1} \quad (4)$$

2.4 Sundaresan & Krishnaswamy

De maneira análoga às técnicas supracitadas, o método de Sundaresan & Krishnaswamy consiste na definição de instantes de tempo específicos, neste caso, t_1 corresponde ao ponto relativo cuja amplitude A_2 obedece à relação $A_1 = A_{max} \times 0.353$, enquanto t_2 corresponde ao instante onde é atingida a amplitude amplitude $A_2 = A_{max} \times 0.853$.

Em seguida, utilizamos os parâmetros t_1 e t_2 para estimar θ e τ através das equações 5 e 5 definidas abaixo:

$$\theta = 1.3t_1 - 0.29t_2 \quad (5)$$

$$\tau = 0.67(t_2 - t_1) \quad (6)$$

De forma empírica, é possível observar uma tendência de redução nos erros de inferência através da utilização dos parâmetros definidos nas equações acima. Este método é aconselhável para estimação de parâmetros sob circunstâncias onde as respostas obtidas apresentam características ruidosas.

2.5 Mollenkamp

O método de Mollenkamp utiliza heurística definida em função da definição de três instante de tempo, sendo t_1 correspondente ao instante onde $A_1 = A_{max} \times 0.15$, t_2 correspondente ao instante onde $A_2 = A_{max} \times 0.45$ e t_3 correspondente ao instante onde $A_3 = A_{max} \times 0.75$.

Ao definir na função estes parâmetros, alimentamos as equações a seguir.

$$x = \frac{t_1 - t_2}{t_3 - t_1} \quad (7)$$

$$\zeta = \frac{0.0805 - 5.547(0.475 - x)^2}{x - 0.356} \quad (8)$$

$$f_2(\zeta) = \begin{cases} (0.708)(2.811)^\zeta, & \text{se } \zeta < 1 \\ 2.6\zeta - 0.6, & \text{caso contrário} \end{cases} \quad (9)$$

$$f_3(\zeta) = (0.922)(1.66)^\zeta \quad (10)$$

$$\omega_n = \frac{f_2(\zeta)}{t_3 - t_1} \quad (11)$$

$$\theta = t_2 - \frac{f_3(\zeta)}{\omega_n} \quad (12)$$

$$\tau_1, \tau_2 = \left\{ \frac{\zeta \pm \sqrt{\zeta^2 - 1}}{\omega_n}, \quad \text{se } \zeta \geq 1 \right. \quad (13)$$

$$G_p(s) = \frac{Ke^{-s\theta}}{(\tau_1 s + 1)(\tau_2 s + 1)} \quad (14)$$

Levando em consideração suas características definidas nas equações acima, sua utilização é aconselhável para sistemas de 2ª ordem.

2.6 Método Yuwana-Seborg

O método de Yuwana-Seborg se difere do método de Ziegler-Nichols por não precisar de um tempo crítico ou o sistema ser marginalmente estável. Como opera em malha fechada esse método exige um teste experimental para estimar os parâmetros para um modelo de primeira ordem e com um tempo de atraso, com realimentação unitária negativa.

Para utilizar o método de Yuwana-Seborg devemos seguir as seguintes equações para chegar em regime estacionário

$$\widehat{y_\infty} = \frac{y_p 2 y_p 1 - (y_m 1)^2}{y_p 1 + y_p 2 - 2 y_m 1} \quad (15)$$

onde y_∞ é a saída em estado estacionários e $\widehat{y_\infty}$ é sua estimação dentro da equação. Tendo isso partimos para as próximas equações para estimar ζ (coeficiente de amortecimento) do sistema

$$\widehat{\zeta} = \frac{-\ln\left(\frac{y_\infty - y_m 1}{y_p 1 - y_\infty}\right)}{\sqrt{\pi^2 + \left(\ln\left(\frac{y_\infty - y_m 1}{y_p 1 - y_\infty}\right)\right)^2}} \quad (16)$$

$$\widehat{\zeta} = \frac{-\ln\left(\frac{y_p 2 - y_\infty}{y_p 1 - y_\infty}\right)}{\sqrt{4\pi^2 + \left(\ln\left(\frac{y_p 2 - y_\infty}{y_p 1 - y_\infty}\right)\right)^2}} \quad (17)$$

Vamos encontrar também K_c , ganho no controlador e um A para ser um setpoint podemos achar o ganho do sistema

$$K_m = \frac{y_\infty}{K_c(A - y_\infty)} \quad (18)$$

e também o K_f que é esse ganho com o controlador

$$K_f = K_c K_m \quad (19)$$

Com isso temos como encontrar os parâmetros que vão formar nossa função de transferência τ_m e θ_m

$$\tau_m = \frac{\Delta_t}{\pi} [\widehat{\zeta} \sqrt{K_f + 1} + \sqrt{\widehat{\zeta}^2 (K_f + 1) + K_f}] X \sqrt{(1 - \widehat{\zeta}^2)(K_f + 1)} \quad (20)$$

$$\theta_m = \frac{2\Delta_t \sqrt{(1 - \hat{\zeta}^2)(K_f + 1)}}{\pi[\hat{\zeta}^2(K_f + 1) + K_f]} \quad (21)$$

onde Δ_t é a variação do tempo do sobressinal.

Podemos então gerar uma função de transferência para nosso sistema de primeira ordem

$$G_m(s) = \frac{K_m e^{-\theta_m s}}{\tau_m s + 1} \quad (22)$$

Agora colocando o sistema em malha fechada obtemos a equação

$$T(s) = \frac{K_f e^{-\theta_m s}}{1 + \tau_m s + K_f e^{-\theta_m s}} \quad (23)$$

Para avançarmos no método de Yuwana precisamos utilizar a aproximação de Padé que gera:

$$T(s) = \frac{\bar{K}(1 - \theta_m s)}{\bar{\tau}^2 s^2 + 2\zeta \bar{\tau} s + 1} \quad (24)$$

$$\bar{K} = \frac{K_f}{K_f + 1} \quad (25)$$

Sendo assim o sistema fica com um zero no semi-plano direito, um sistema de fase não-mínima.

$$\bar{\tau} = \sqrt{\left(\frac{\theta_m \tau_m}{2(K_f + 1)}\right)} \quad (26)$$

$$\zeta = \frac{\tau_m + 0.5\theta_m(1 - K_f)}{\sqrt{(2\theta_m \tau_m(K_f + 1))}} \quad (27)$$

3 METODOLOGIA

3.1 Software e Ferramentas

Para verificar os métodos foi utilizada a linguagem estatística R [Team et al., 2013], por intermédio do uso de bibliotecas contendo funções de controle. Inserido no escopo específico da disciplina, encontramos disponível a *toolbox* **CONTROL** [Ubah, 2017], desenvolvida através do programa Google Summer of Code [GSoC, 2017] que trata da modelagem de Sistemas Lineares e Unvariantes no tempo, além de modelos contínuos e discretos, bem como as devidas transformações aos domínios de frequência e acoplamento de diagramas de blocos.

3.2 Recursos Implementados

A estrutura definida pela *toolbox* para funções de transferência exige a atribuição explícita de seus respectivos numeradores e denominadores em sua forma expandida. Este fator é particularmente limitante no momento de inserir as modelagens dos atrasadores, ainda sem suporte no pacote devido ao seu recente desenvolvimento (Julho, 2017). Diante disso, foi implementada uma correção para esta limitação através de aproximações polinomiais utilizando séries de potências.

Em seguida, realizamos a realimentação do sistema e conseguimos através de manipulação matricial a localização dos máximos e mínimos locais das funções. Para concluir, realizamos a estimação dos parâmetros e construímos o modelo aproximado.

Considerando a recente publicação da biblioteca em uso citada acima, alguns ajustes foram necessários para tornar os experimentos praticáveis: Os blocos genéricos para funções de transferência de controladores e atrasadores são ilustrados a seguir.

```

1  library(control)
2
3  #####
4  # Delayer block using 1st Order Pade approximation
5  DelayerBlock <- function(theta) {
6      numDelayer <- c((-theta/2), 1)
7      denDelayer <- c((theta/2), 1)
8
9      return(tf(numDelayer, denDelayer))
10 }
11
12 #####
13 # Generic blocks: 1st Order System
14 ApproximationBlock1stOrder <- function(nP, nPs, dP, dPs) {
15     numPB <- c(nPs, nP)
16     denPB <- c(nPs, nP)
17
18     return(tf(numPB, denPB))
19 }
20
```

```

21 #####
22 # Generic blocks: 2nd Order System
23 ApproximationBlock2ndOrder <- function(nP, nPs, nPs2, dP, dPs, dPs2) {
24     numPlant <- c(nPs2, nPs, nP)
25     denPlant <- c(dPs2, dPs, dP)
26
27     return(tf(numPlant, denPlant))
28 }

```

Ainda no contexto do tratamento de dados, foram observados alguns conjuntos ruidosos, de modo que é possível observar na seção 4 algumas curvas filtradas para melhorar a legibilidade dos dados para calibração e extração dos parâmetros para realizar as previsões.

Os filtros em frequência baseados em **Transformada de Fourier** foram implementados manualmente e seu procedimento é reportado a seguir:

```

1 # TRANSFORMADA DE FOURIER DIRETA
2 getFFTFreqs <- function(data, samplingFreq) {
3     Nyq.Freq=samplingFreq/2;
4
5     if ((length(data) %% 2) == 1) { # °N ímpar de amostras
6         FFTFreqs <- c(seq(0, Nyq.Freq, length.out=(length(data)+1)/2),
7             seq(-Nyq.Freq, 0, length.out=(length(data)-1)/2))
8     }
9
10    else { # °N par
11        FFTFreqs <- c(seq(0, Nyq.Freq, length.out=length(data)/2),
12            seq(-Nyq.Freq, 0, length.out=length(data)/2))
13    }
14    return(FFTFreqs)
15 }
16
17 # TRANSFORMADA INVERSA DE FOURIER NORMALIZADA
18 fftinv <- function(x) { fft(x, inverse=TRUE) / length(x) }
19
20 # FILTRO PASSA-BAIXA
21 lowpass <- function(fftData, samplingFreq, threshold, times=1) {
22     FFTFreqs <- Mod(getFFTFreqs(fftData, samplingFreq))
23     coefs <- 1/(1+(FFTFreqs/threshold)) # coeficientes multiplicativos para ganho
24
25     if(times>1)
26         for(i in 1:times)
27             coefs <- coefs*(1/(1+(threshold/FFTFreqs)))
28
29     data <- coefs*fftData
30     data <- Re(fftinverse(data))
31     return(data)
32 }

```

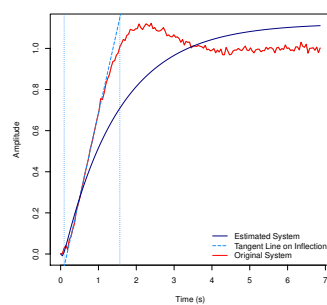
4 RESULTADOS E DISCUSSÕES

4.1 Modelagem de Métodos de Identificação de Sistemas

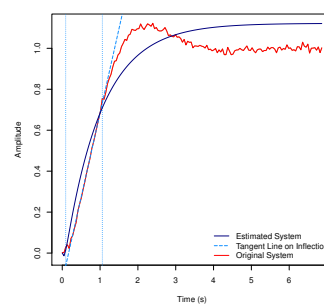
Neste seção serão apresentados os dados obtidos através dos experimentos envolvendo as técnicas clássicas de identificação de sistemas, onde foram testados os seis métodos introduzidos na seção 2.

As Figuras 1, 2, 3 e 4 apresentam os gráficos em função do tempo dos *datasets* avaliados, além do comportamento das respostas aproximadas.

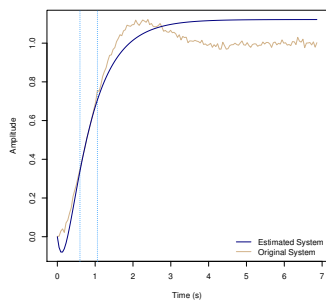
4.1.1 Dataset 1



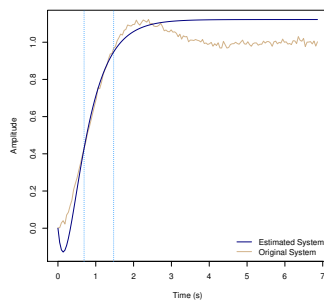
(a) Ziegler-Nichols.



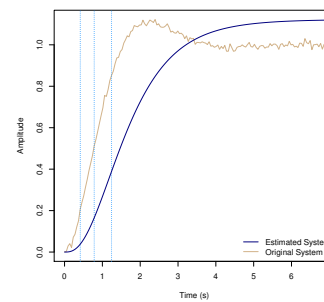
(b) Hagglund.



(c) Smith 1ª Ordem.



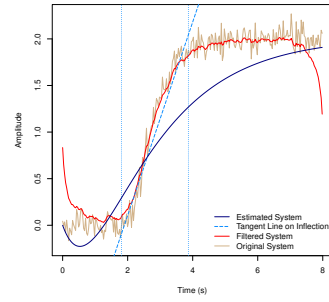
(d) Sundaresan-Krishnaswamy.



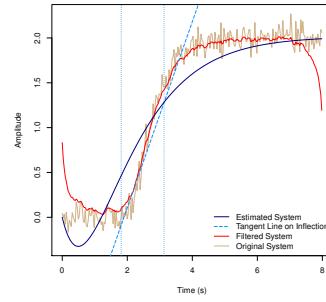
(e) Mollenkamp.

Figura 1: Comportamento do Sistema 1.

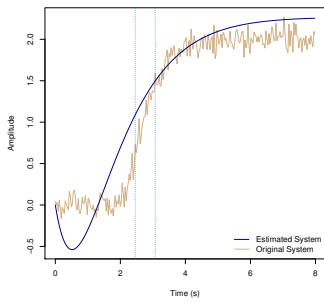
4.1.2 Datasets 2 e 3



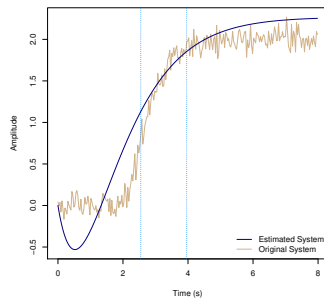
(a) Ziegler-Nichols.



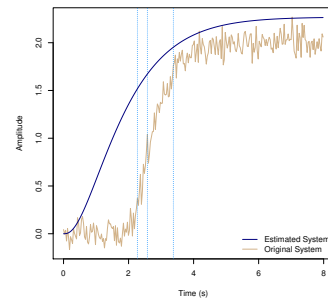
(b) Hagglund.



(c) Smith 1ª Ordem.

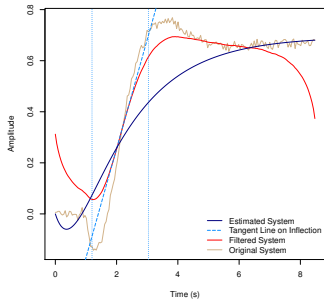


(d) Sundaresan-Krishnaswamy.

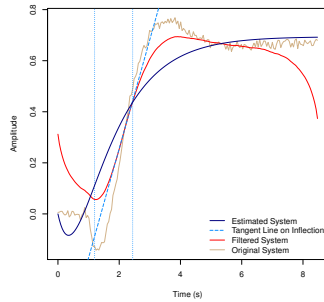


(e) Mollenkamp.

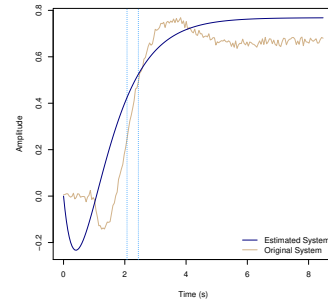
Figura 2: Comportamento do Sistema 2.



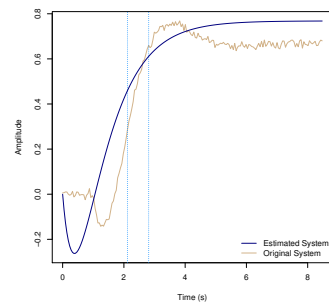
(a) Ziegler-Nichols.



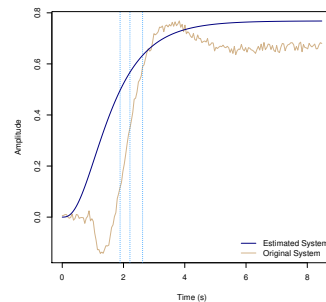
(b) Hagglund.



(c) Smith 1ª Ordem.



(d) Sundaresan-Krishnaswamy.



(e) Mollenkamp.

Figura 3: Comportamento do Sistema 3.

4.1.3 Dataset 4

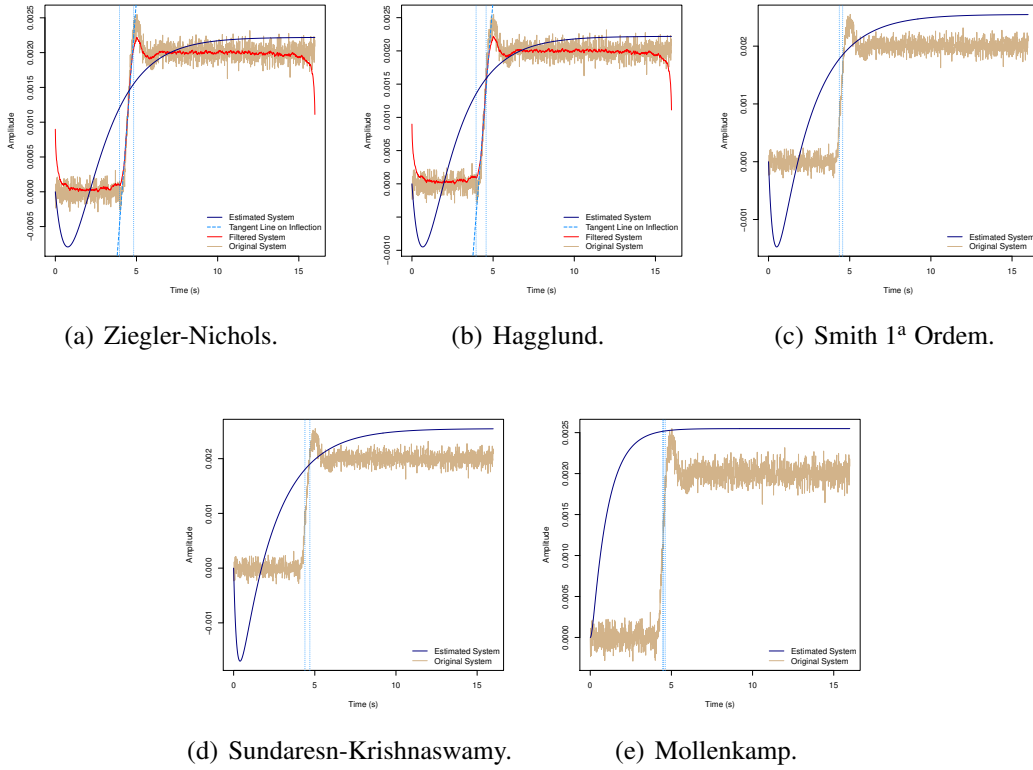


Figura 4: Comportamento do Sistema 4.

4.2 Avaliação quantitativa: Erro Quadrático Médio

As Figuras 5(a), 5(b), 6(a) e 6(b) apresentam a avaliação das técnicas utilizando o Erro Quadrático Médio.

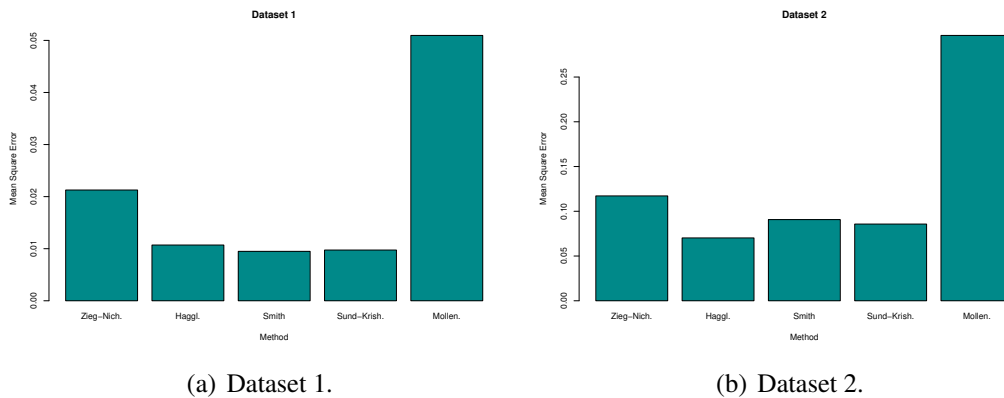


Figura 5: Avaliação utilizando Erro Quadrático Médio.

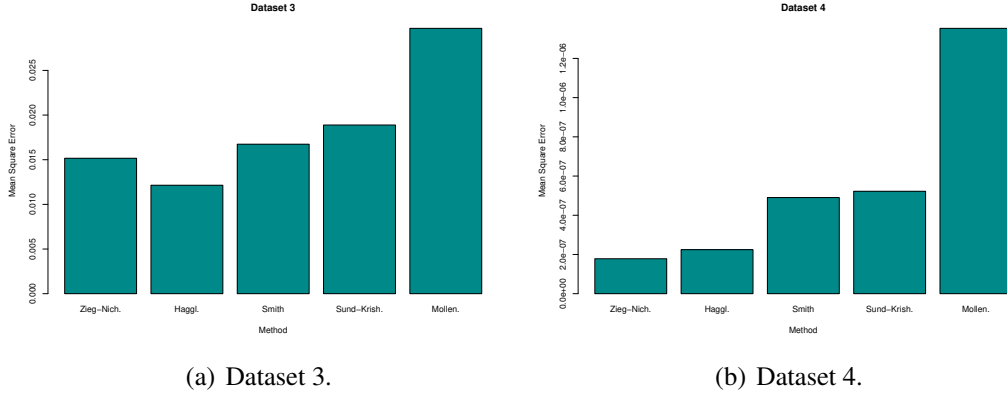


Figura 6: Avaliação utilizando Erro Quadrático Médio.

4.3 Métodos determinísticos para sistemas em malha fechada

A seguir, serão apresentados os dados obtidos através dos experimentos para sistemas em malha fechada utilizando o método de Yuwana Seborg, onde foi avaliado o comportamento do método com quatro entradas respectivas às funções de transferência das equações 28, 29, 30 e 31.

As Figuras 1, 2, 3 e 4 apresentam os gráficos em função do tempo dos conjuntos de dados avaliados, além do comportamento das respostas aproximadas. É possível observar que a técnica falha ao estimar o sistema da Figura 7(c) pela característica do sinal com ausência de pontos críticos.

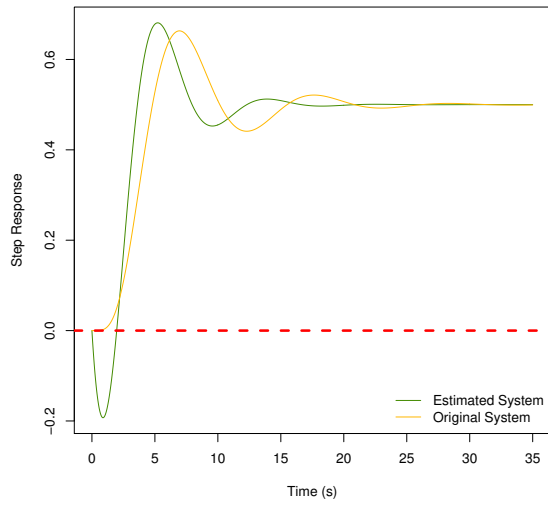
$$G(s) = \frac{1}{(s+5)^5} \quad (28)$$

$$G(s) = \frac{10}{s(s+1)(s+2)} \quad (29)$$

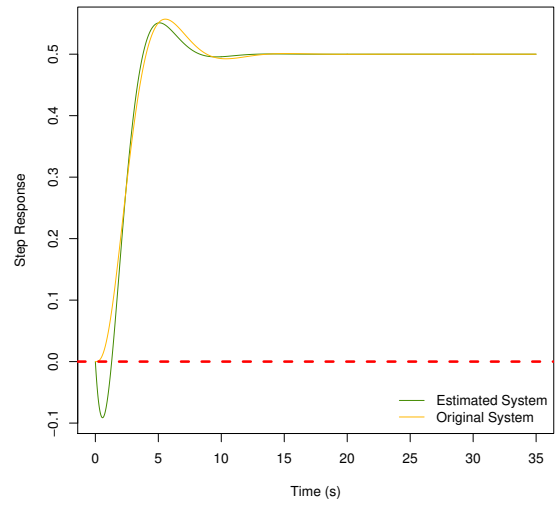
$$G(s) = \frac{3e^{-0.1s}}{2s+1} \quad (30)$$

$$G(s) = \frac{3e^{-0.5s}}{(s+1)^2} \quad (31)$$

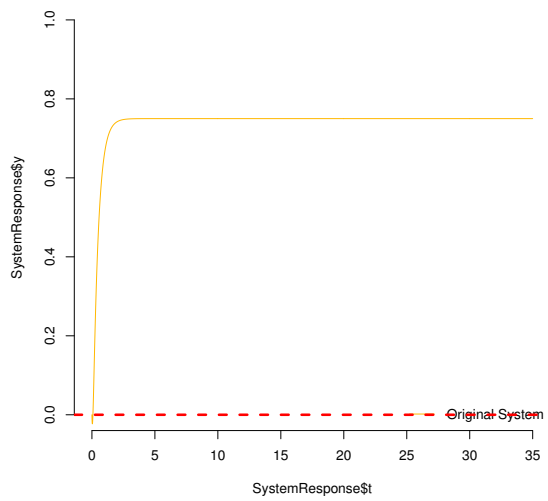
4.3.1 Comportamento dos Sistemas



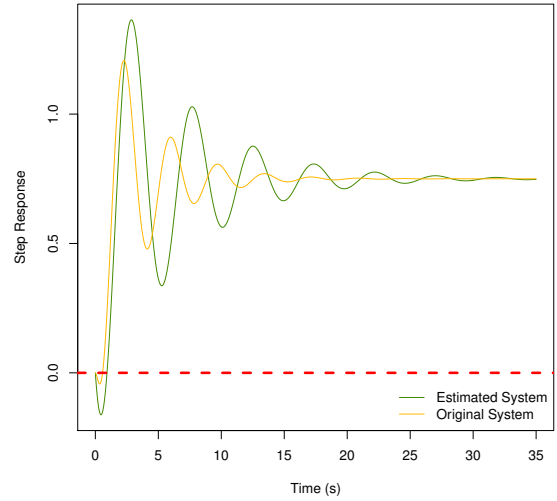
(a) Dataset 1.



(b) Dataset 2.



(c) Dataset 3.



(d) Dataset 4

Figura 7: Estimativa utilizando Yuwana Seborg.

4.3.2 Informações Complementares

A seguir, são apresentadas as estimativas dos parâmetros em cada um dos sistemas avaliados. Abaixo, referentes ao sistema: $G(s) = \frac{1}{(s+5)^5}$.

```
1      First Local Peak:  0.663258800926032
2      Second Local Peak: 0.521199298511143
3      First Local Minimum: 0.441165254344954
4      Mean Square Error: 0.00536270246287152
5      yInf: 0.499998162230433
6      Dt: 5.34
7      Km: 0.99999264894875
8      Zeta: 0.308984476643971
9      Tm: 3.49394837243725
10     Theta: 2.99189935586292
11     TauEst: 1.61659985055947
12     KEst: 0.499998162230433
13
14     [1] "Original System:"
15
16     yl:
17
18         1
19     - - - - -
20         s^5 + 5 s^4 + 10 s^3 + 10 s^2 + 5 s + 2
21
22     Transfer Function: Continuous time model
23
24     [1] "Estimated System Below:"
25
26     yl:
27         -0.7479721 s^1 + 0.4999982
28     - - - - -
29         1.6166 s^2 + 0.9990085 s + 1
30
31
32     Transfer Function: Continuous time model
```

Abaixo, referentes ao sistema: $G(s) = \frac{10}{s(s+1)(s+2)}$.

```
1 First Local Peak: 0.556975877609381
2 Second Local Peak: 0.50098610600011
3 First Local Minimum: 0.492503370510311
4 Mean Square Error: 0.000398383328219251
5 yInf: 0.499999791667699
6 Dt: 4.81
7 Km: 0.999999166671142
8 Zeta: 0.542385901367748
9 Tm: 3.68795473403699
10 Theta: 1.79455626257906
11 TauEst: 1.28629749879854
12 KEst: 0.499999791667699
13
14 [1] "Original System:"
15
16 yl:
17
18           1
19 - - - - -
20 2 s^3 + 5 s^2 + 4 s + 2
21
22 Transfer Function: Continuous time model
23
24 [1] "Estimated System Below:"
25
26 yl:
27 -0.4486389 s^1 + 0.4999998
28 - - - - -
29 1.286297 s^2 + 1.395339 s + 1
30
31
32 Transfer Function: Continuous time model
```

Abaixo, referentes ao sistema: $G(s) = \frac{3e^{-0.1s}}{2s+1}$.

```
1 First Local Peak: 0.749999999999998
2 Second Local Peak: 0.749999999999998
3 First Local Minimum: 0.749999999999998
4 yInf: NaN
5 Dt: 0
6 Km: NaN
7 Zeta: NaN
8 Tm: NaN
9 Theta: NaN
10 TauEst: NaN
11 KEst: NaN
12
13 [1] "Original System:"
14
15 yl:
16     -0.15 s^1 + 3
17     - - - - -
18     0.1 s^2 + 1.9 s + 4
```

Abaixo, referentes ao sistema: $G(s) = \frac{3e^{-0.5s}}{(s+1)^2}$.

```

1      First Local Peak:  1.2072919418178
2      Second Local Peak:  0.911000528428966
3      First Local Minimum:  0.478662773994411
4      Mean Square Error:  0.0270744974328487
5      yInf:  0.750000304972139
6      Dt:  1.85
7      Km:  3.00000487956017
8      Zeta:  0.163897615468344
9      Tm:  2.4288947504646
10     Theta:  1.11147518860157
11     TauEst:  0.580910164912166
12     KEst:  0.750000304972139
13
14     [1] "Original System:"
15
16     yl:
17           -0.75 s^1 + 3
18     - - - - -
19     0.25 s^3 + 1.5 s^2 + 1.5 s + 4
20
21
22     Transfer Function: Continuous time model
23
24     [1] "Estimated System Below:"
25
26     yl:
27           -0.4168034 s^1 + 0.7500003
28     - - - - -
29     0.5809102 s^2 + 0.1904196 s + 1
30
31
32     Transfer Function: Continuous time model

```

5 CONSIDERAÇÕES FINAIS

Este trabalho apresentou uma implementação prática para execução dos métodos clássico de identificação de sistemas em malha aberta e fechada, introduzindo uma portabilidade de funções tipicamente oferecidas em ferramentas pagas (Matlab, por exemplo) para código aberto e colaborativo na linguagem R, ponto que consideramos uma das principais contribuições deste trabalho.

Consideramos **positivos** os resultados obtidos nos experimentos, bem como consistente o aprendizado incorporado no decorrer da redação deste relatório. As dificuldades encontradas nas tentativas anteriores, tais como a utilização de blocos atrasadores e redução de ruído através de filtragem, foram solucionadas com sucesso: Fato adicional que confirma a sedimentação das ferramentas em estudo.

Finalmente, todo código utilizado neste trabalho encontra-se documentado e disponível em repositório aberto através do link <https://github.com/isrvasconcelos/SystemsIdentification-2017.2.git> para fins de reprodutibilidade.

Referências

- L.A. Aguirre. *Introdução à Identificação de Sistemas – Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais*. Editora UFMG, 2007. ISBN 9788570415844. URL <https://books.google.com.br/books?id=f9IwE7Ph0fYC>.
- GSoC. Google summer of code, 2017. URL <https://summerofcode.withgoogle.com/>.
- PM Meshram and Rohit G Kanojiya. Tuning of pid controller using ziegler-nichols method for speed control of dc motor. In *Advances in Engineering, Science and Management (ICAESM), 2012 International Conference on*, pages 117–122. IEEE, 2012.
- Emmanuel Paradis, Julien Claude, and Korbinian Strimmer. Ape: analyses of phylogenetics and evolution in r language. *Bioinformatics*, 20(2):289–290, 2004.
- Jan Erik Mont Gomery Pinto. Aplicação prática do método de sintonia de controladores pid utilizando o método do relé com histerese. Master’s thesis, Universidade Federal do Rio Grande do Norte, 2014.
- R Development Core Team. *The R Project for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2010. ISBN 3-900051-07-0.
- R Core Team et al. R: A language and environment for statistical computing. 2013.
- Ben Ubah. Control systems toolbox in r - a gsoc 2017 project, jul 2017. URL <https://rviews.rstudio.com/2017/07/06/control-systems-toolbox-in-r---a-gsoc-2017-project/>.
- Minta Yuwana and Dale E Seborg. A new method for on-line controller tuning. *AIChE Journal*, 28(3):434–440, 1982.
- J.G. Ziegler and N. B. Nichols. Optimum settings for automatic controllers. *Transactions of the ASME*, 1942.