UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

SEL0367 - Controle Adaptativo

PROFº Marco Henrique Terra

Prova 1

FELIPE ANDRADE GARCIA TOMMASELLI Nº USP: 11800910

São Carlos, 2023

1. Implementação

1.1. Exercício 1

Sistema DARMA:

$$y(k) = a_1 y(k-1) + a_2 y(k-2) + b_0 u(k-3)$$

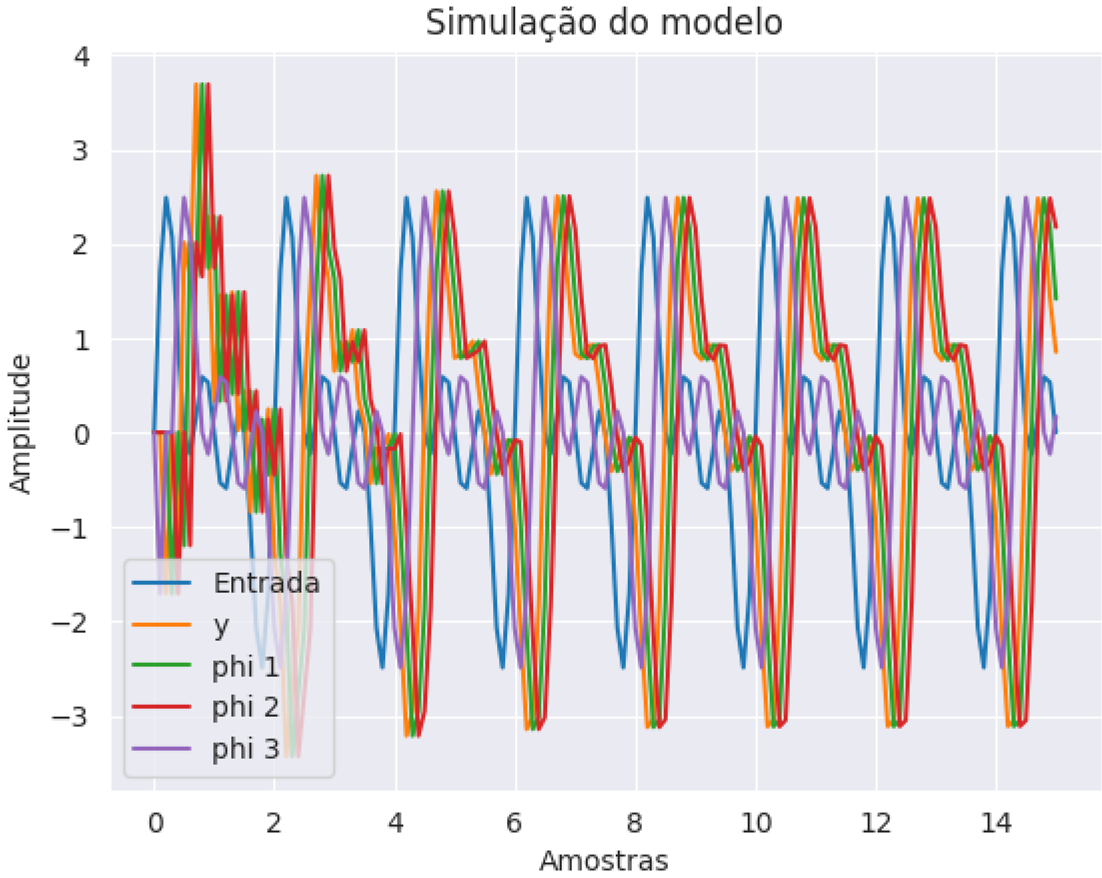
Para a estimação, será utilizado o modelo paramétrico $y = z = \theta^* \phi$, com um fator de normalização do tipo: $m^2 = 1 + 0.1\,\phi\,\phi^T$.

De acordo com o modelo paramétrico **(pergunta a))**:

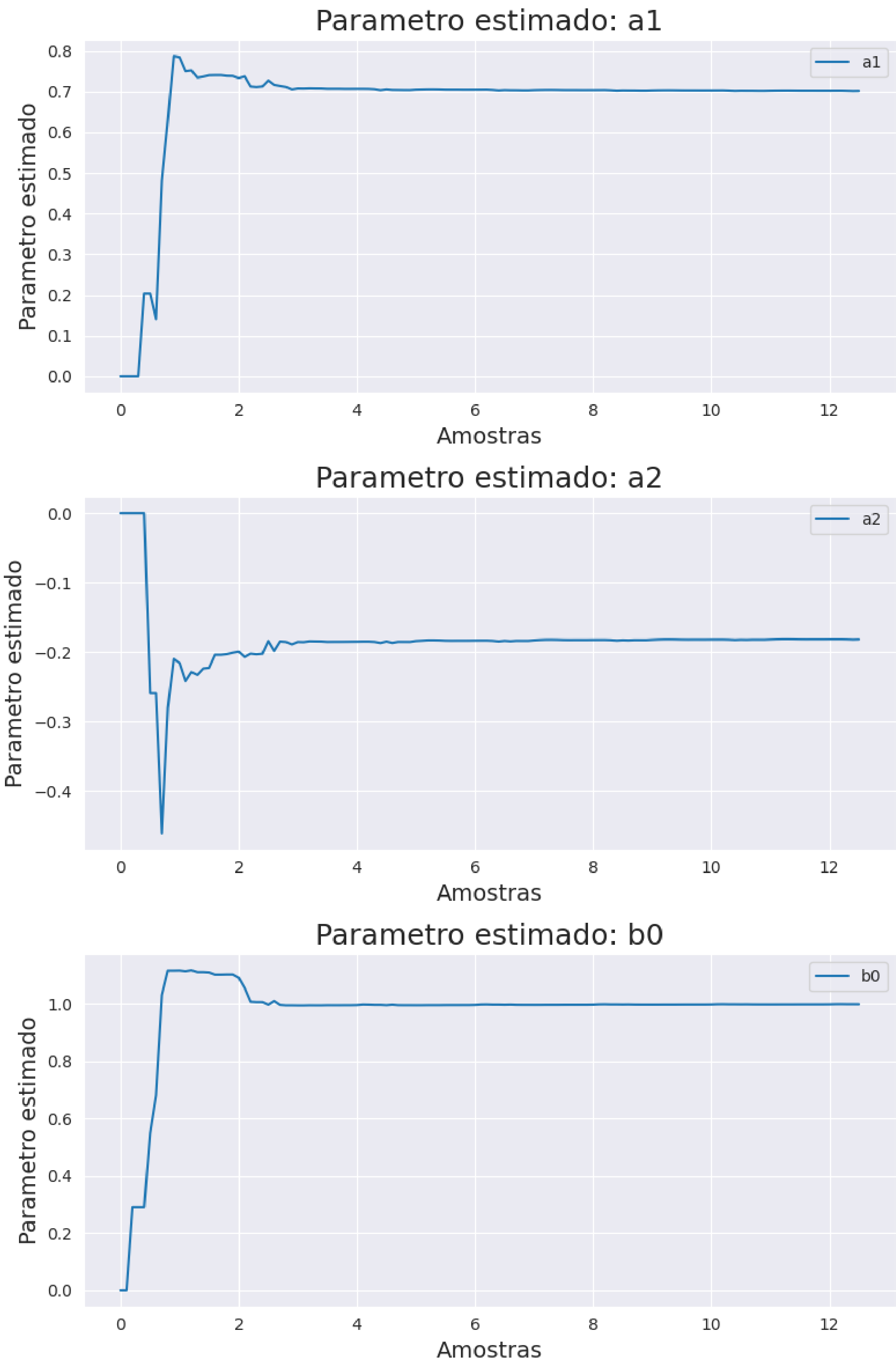
$$\begin{cases} z = y_{[k+1]}) \\ \theta = [a_1, a_2, b_0] \\ \phi = [y_{[k-1]}, y_{[k-2]}, u_{[k-3]}] \end{cases}$$

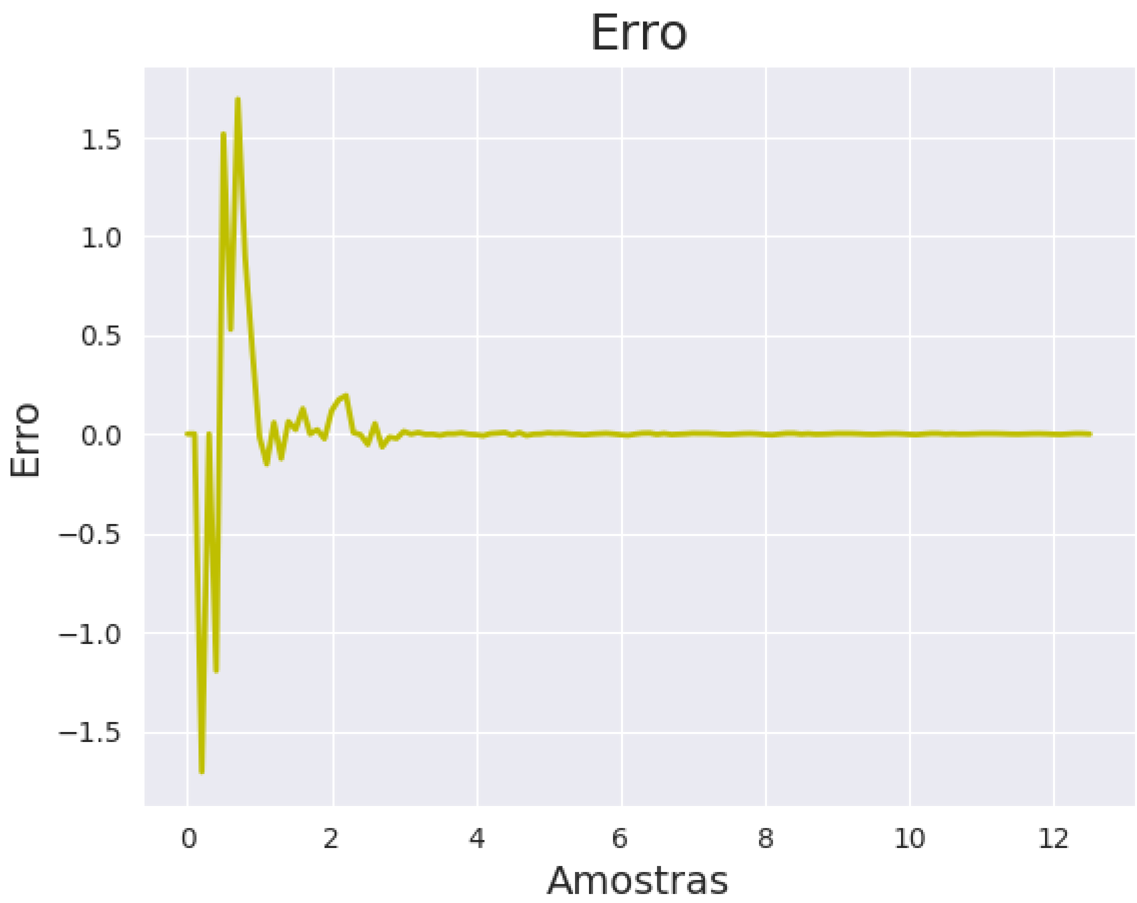
Além disso, tomaremos a simulação dos parâmetros com a1 = 0.7, a2 = -0.18 e b0 = 1. A princípio a entrada será suficientemente rica do tipo: u = sin(pi * T) + sin(2 * pi * T) + sin(3 * pi * T).

O sistema pode ser caracterizada visualmente (apenas para efeito ilustrativo) como:



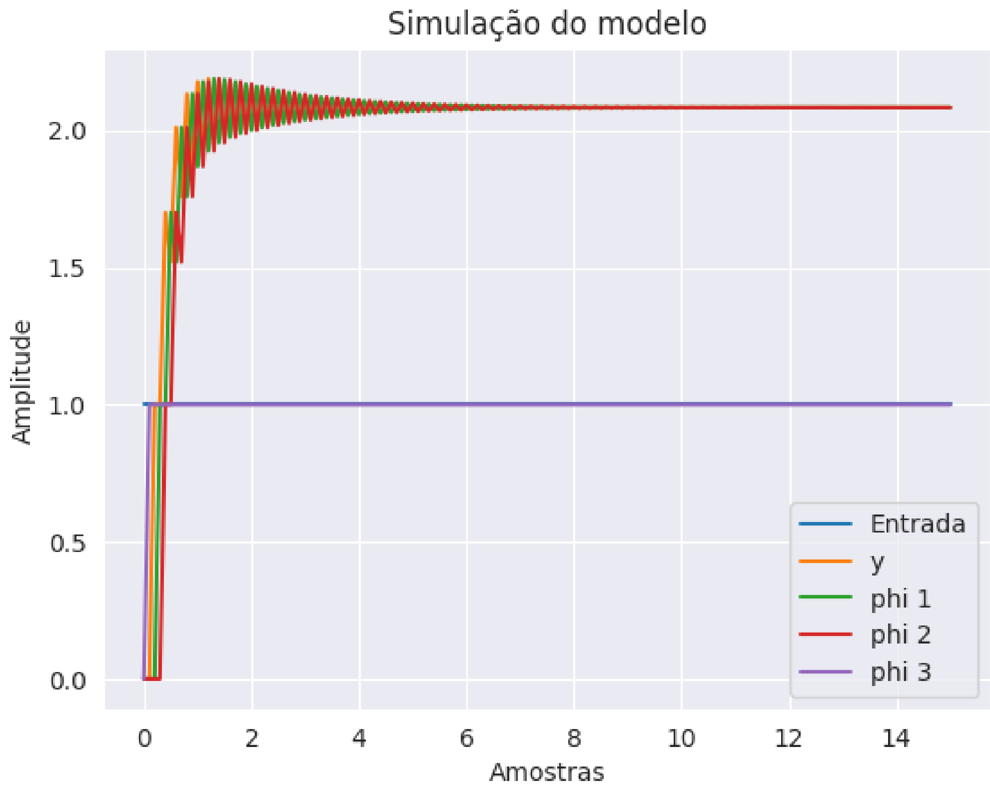
Para esse caso, a resposta da parametrização pode ser vista abaixo **(pergunta b)**:

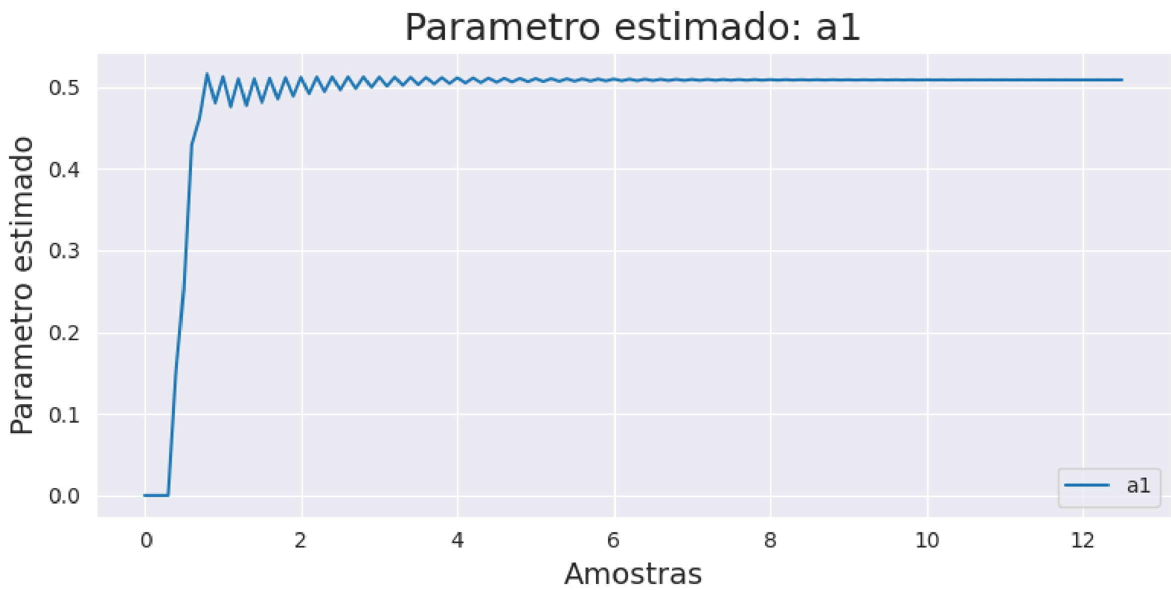

Parametro estimado: a1
Parametro estimado
Amostras
a1
Parametro estimado: a2
Parametro estimado
Amostras
a2
Parametro estimado: b0
Parametro estimado
Amostras
b0

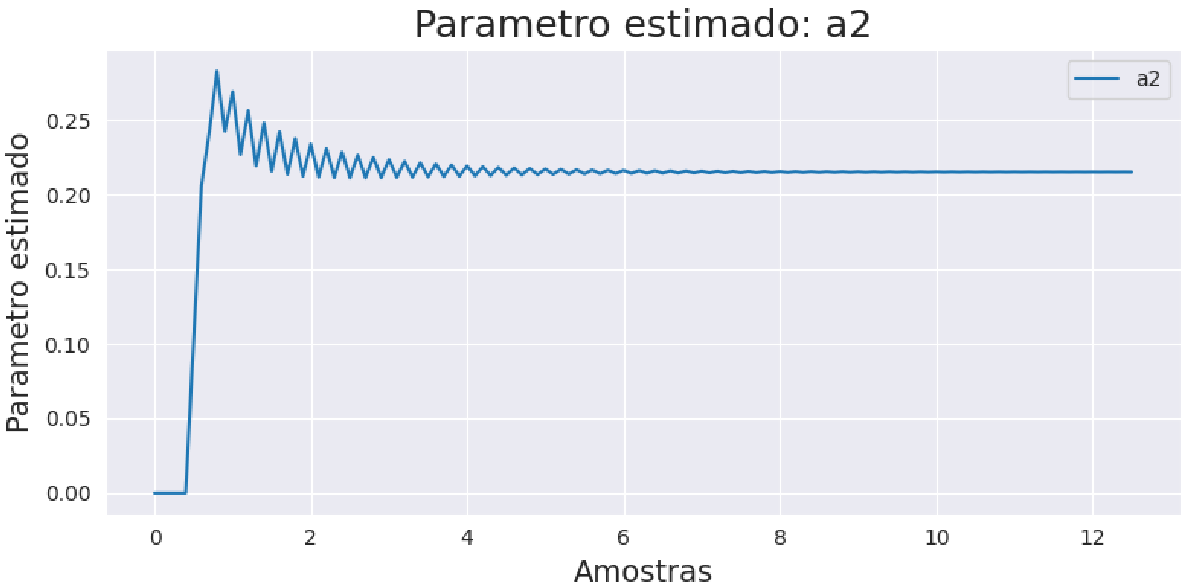


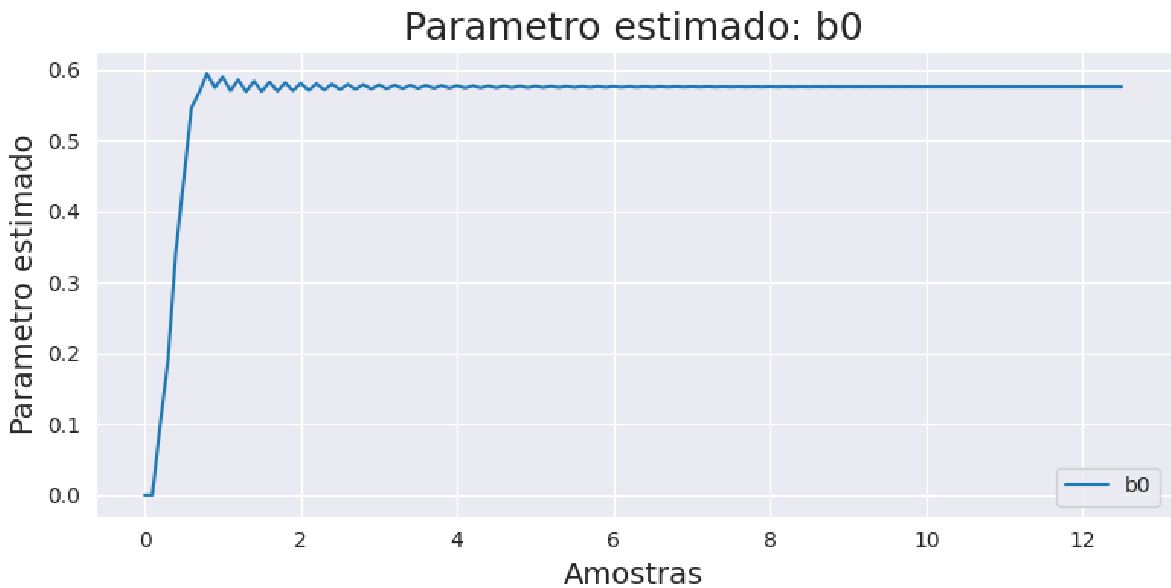
Perceba que os resultados foram bem satisfatórios, foi possível encontrar todos os parâmetros a1, a2 e b0 em poucas iterações, isto é, em pouco mais de 3 amostras, mostrando um resultado muito coerente.

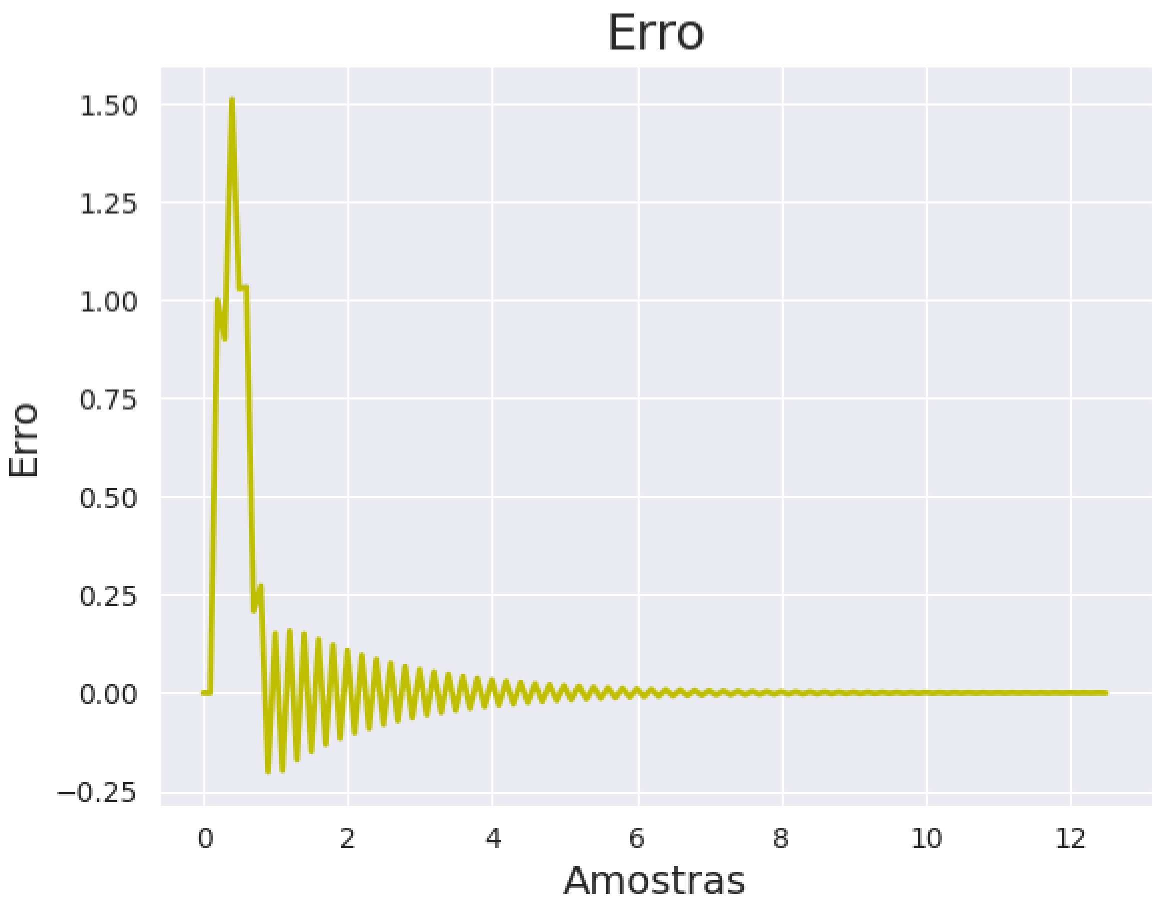
Por fim, colocando um entrada não suficientemente rica, como uma entrada degrau que possui menos de 3 frequências, nós podemos observar a não convergência do método **(pergunta c)**.


Simulação do modelo
Amplitude
Amostras
Entrada
y
phi 1
phi 2
phi 3


Parametro estimado: a1
Parametro estimado
Amostras
a1
0.0
0.1
0.2
0.3
0.4
0.5
0
2
4
6
8
10
12


Parametro estimado: a2
Parametro estimado
Amostras
a2
0.00
0.05
0.10
0.15
0.20
0.25
0
2
4
6
8
10
12


Parametro estimado: b0
Parametro estimado
Amostras
b0
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0
2
4
6
8
10
12



Com isso, fica evidente que a entrada não suficientemente rica não há convergência para os valores queridos, uma vez que todos os parâmetros estão estimados errados.

1.2. Exercício 2

Considerando a planta fixa e realizando o procedimento visto em aula para 2 parâmetros (b1 e b0):

$$y = \frac{b_1 s + b_0}{s^2 + 2s + 1} u$$

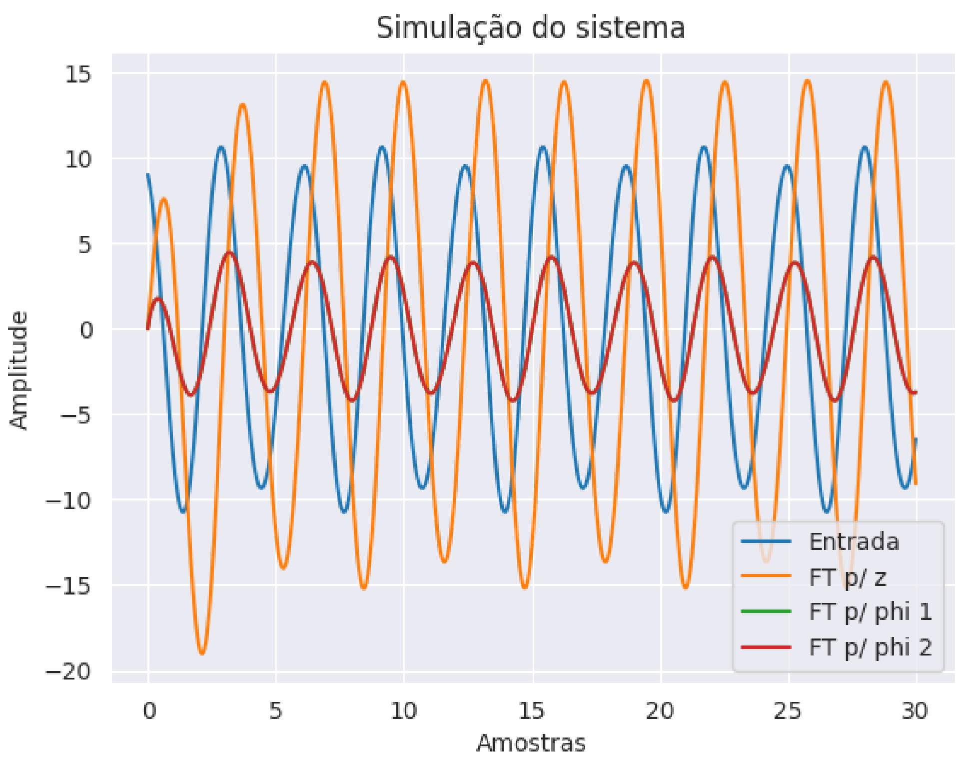
Com entrada:

$$u = \sin(2t + \frac{\pi}{7}) + 0.9\cos(3t)$$

Refazendo os mesmos passos do item anterior e reescrevendo o sistema no modo SPM:

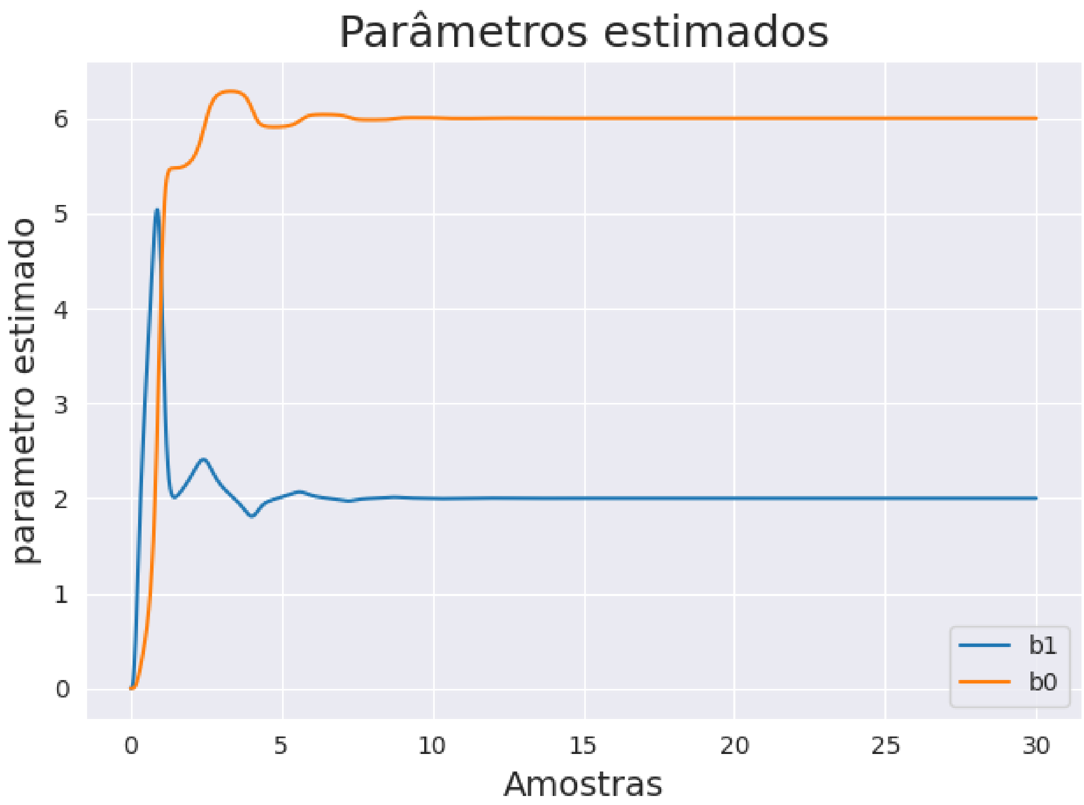
$$\begin{cases} z = [b_1 \cdot s \cdot U(s) + b_0 \cdot U(s)] \cdot \frac{1}{\lambda} \\ \theta^* = [b_1, b0] \\ \phi = [1, s] \cdot \frac{1}{\lambda} \end{cases}$$

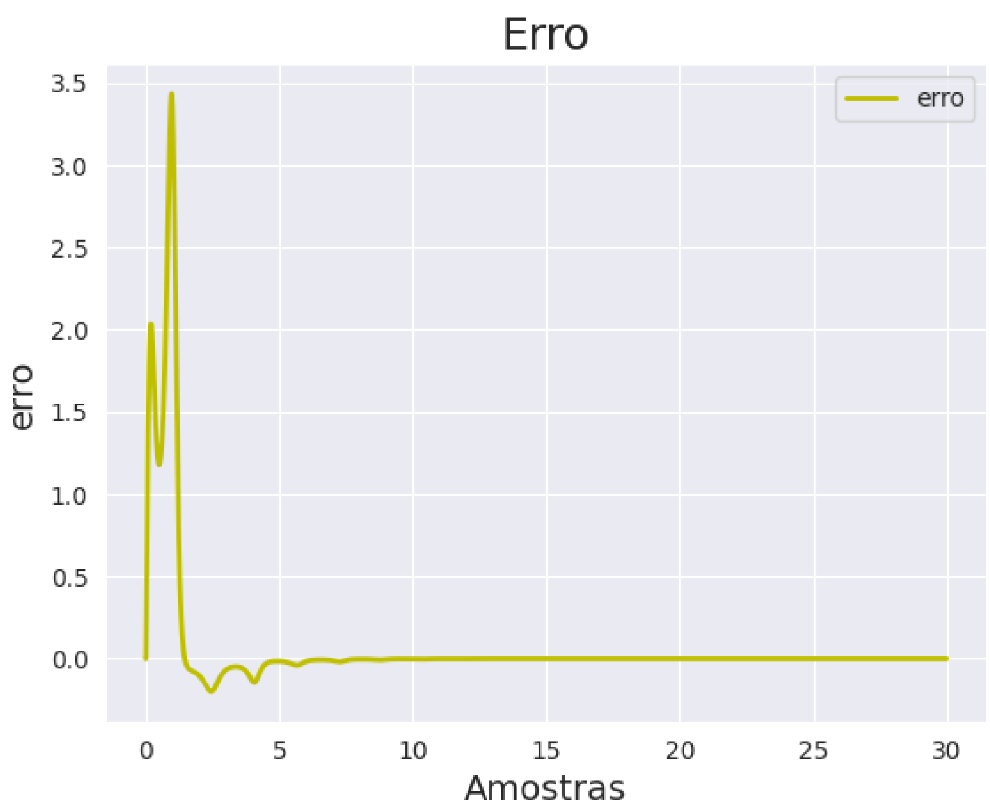
A definição do erro e do fator de normalização são dadas no enunciado. Com isso, a relação dos parâmetros conhecidos (para ilustração) é mostrado abaixo:



Os resultados obtidos em termos de estimativa de parâmetros está abaixo:

- gama = [[4, 0], [0, 4]];


Parâmetros estimados
parametro estimado
Amostras
b1
b0
0
1
2
3
4
5
6
0
5
10
15
20
25
30


Erro
erro
Amostras
erro
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
0
5
10
15
20
25
30

Os resultados para a implementação proposta demonstram uma boa convergência do sistema, convergindo para pouco mais de 5 amostras. Com isso, os valores de b0 e b1 foram encontrados.

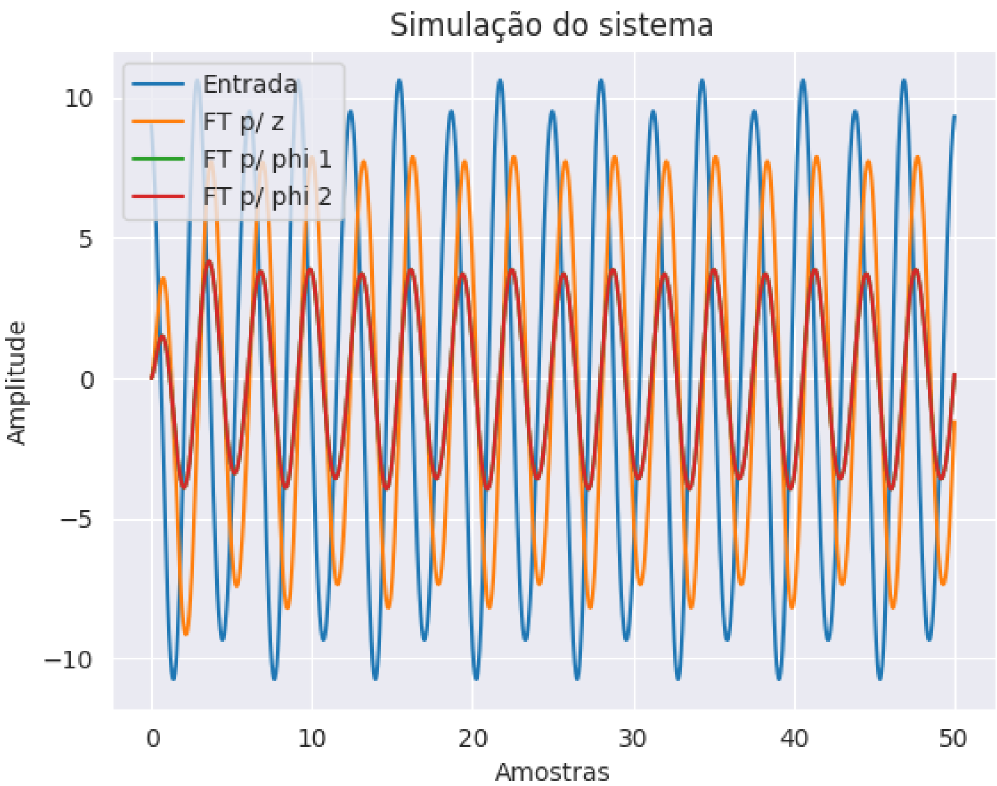
Exercício 3: FT fixa

Considerando a planta:

$$y = \frac{3s + 2}{s^2 + a_1 s + a_0} u$$

Com entrada:

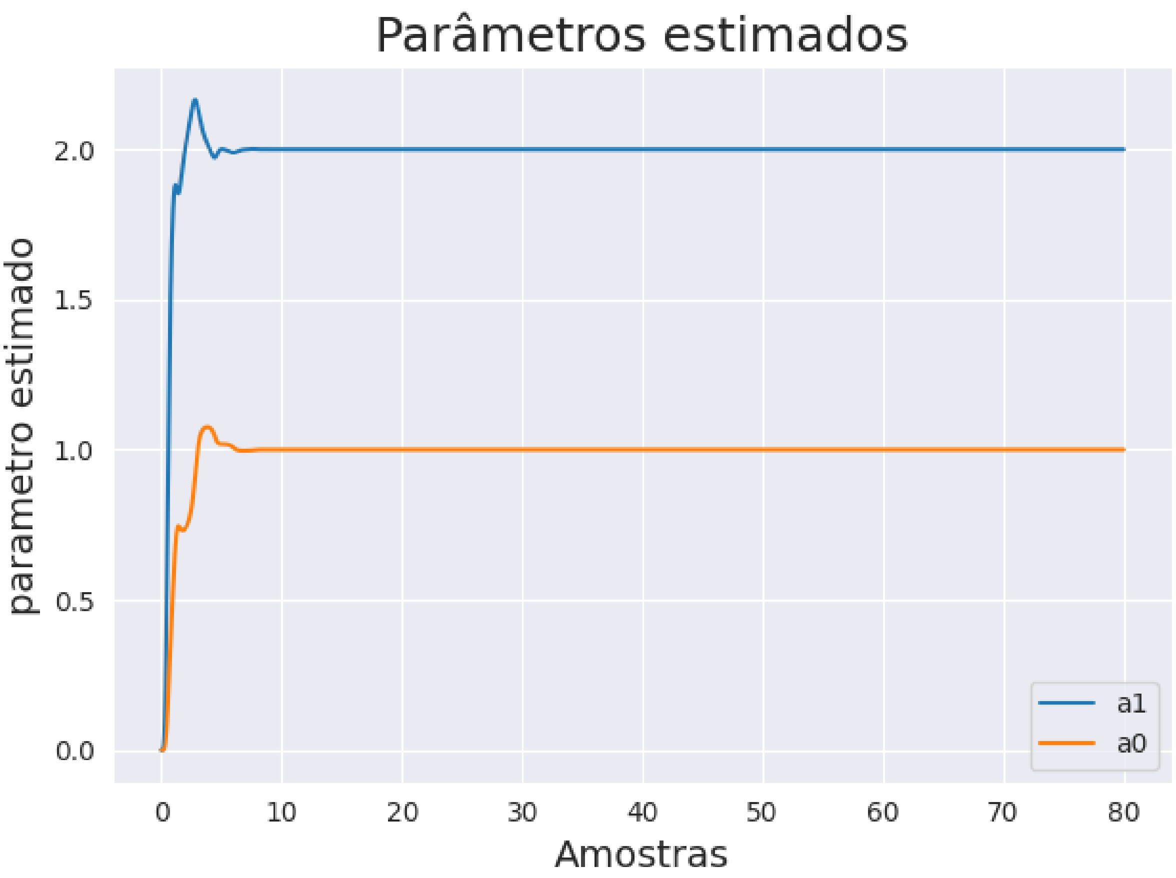
$$u = \sin(2t + \frac{\pi}{7}) + 0.9 \cos(3t)$$

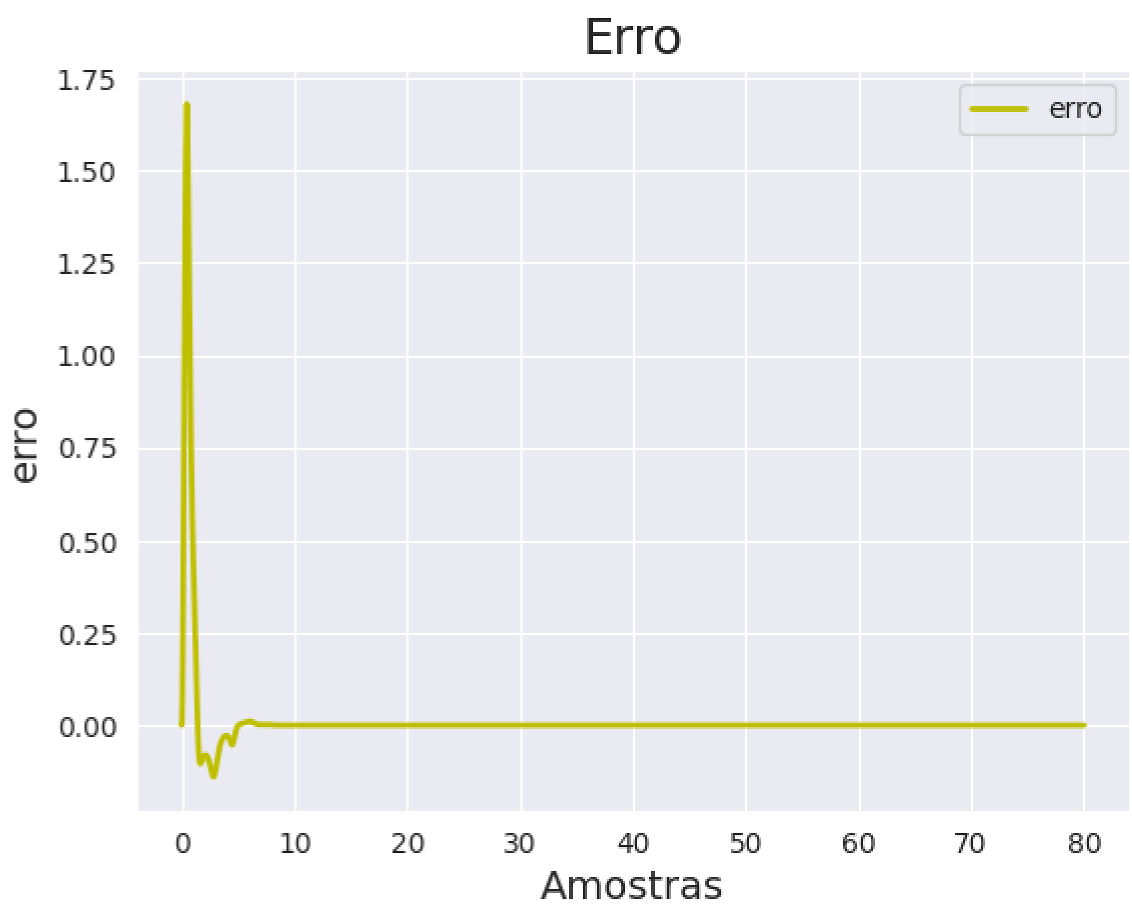


Refazendo os mesmos passos do item anterior e reescrevendo o sistema no modo SPM utilizando os mesmo parâmetros de erro e ms (checar código no fim do documento para mais detalhes).

Os resultados obtidos em termos de estimativa de parâmetros está abaixo:

- 1: gama = [[2, 0], [0, 2]];





A estimativa de parâmetros apresentou resultados bons, convergindo em menos de 10 épocas. Configurando uma boa aplicação do método proposto para estimação correta dos parâmetros queridos.

2. Código desenvolvido

Como dito anteriormente, foi utilizado Python em especial Jupyter Notebook para o exercício. O código fonte pode ser encontrado no repositório do Github do discente pelo link:

- Exercício 1:

https://github.com/Felipe-Tommaselli/AdvancedControl/blob/main/P1/p1_1.ipynb

- Exercício 2:

https://github.com/Felipe-Tommaselli/AdvancedControl/blob/main/P1/p1_2.ipynb

- Exercício 3:

https://github.com/Felipe-Tommaselli/AdvancedControl/blob/main/P1/p1_3.ipynb

Vale ressaltar que usualmente o github pelo navegador demora um pouco (1~2 minutos) para carregar o Jupyter notebook.