



UNIVERSIDADE FEDERAL DO AMAZONAS  
FACULDADE DE TECNOLOGIA  
ENGENHARIA DA COMPUTAÇÃO

## Identificação e Controle de um Levitador à Ar

Mateus Martínez de Lucena

MANAUS-AM

2018

Mateus Martínez de Lucena

## Identificação e Controle de um Levitador à Ar

Monografia apresentada à Coordenação do Curso de Engenharia da Computação da Universidade Federal do Amazonas, como parte dos requisitos necessários à obtenção do título de Engenheiro de Computação.

Orientador: Iury Valente de Bessa

MANAUS-AM

2018

*DEDICATÓRIA AQUI*

# Agradecimentos

AGRADECIMENTOS AQUI.

*EPÍGRAFE AQUI*

*(AUTOR AQUI, TÍTULO DA OBRA.)*

# Resumo

No estudo do controle de sistemas nos deparamos com variados sistemas clássicos extensivamente estudados e exauridos. Neste trabalho nos propomos a construir um sistema de túnel de vento capaz de levitar através do empuxo do fluxo de ar gerado, o estudo das propriedades do sistema, a obtenção de um modelo matemático através de testes e o controle do sistema. A fim de demonstrar a usabilidade deste sistema como material didático para a matéria de Laboratório Sistemas de Controle.

**Palavras-chave:** Levitador, Túnel de Vento, Controle.

# Abstract

ABSTRACT AQUI

**Keywords:** KEYWORDS HERE.

# Lista de Figuras

2.1	Esquema do amortecedor da roda de um carro . . . . .	5
3.1	Gráfico $v_{ar}$ x altura . . . . .	20
3.2	Simulador do túnel de vento . . . . .	21
4.1	Gráfico da saída PRBS do sistema . . . . .	24
4.2	Gráfico da saída PRBS do simulador . . . . .	24
4.3	Autocorrelação dos resíduos do modelo $ARX1$ . . . . .	26
4.4	Autocorrelação dos resíduos do modelo $ARX2$ . . . . .	27
4.5	Autocorrelação dos resíduos do modelo $ARXsim$ . . . . .	27
4.6	Autocorrelação dos resíduos do modelo $SUB1$ . . . . .	27
4.7	Resposta ao degrau do modelo $ARX1$ . . . . .	28
4.8	Resposta ao degrau do modelo $ARX2$ . . . . .	28
4.9	Resposta ao degrau do modelo $ARXsim$ . . . . .	28
4.10	Resposta ao degrau do modelo $SUB1$ . . . . .	29
5.1	Resposta ao degrau unitário do modelo $SUB1$ controlado . . . . .	32
5.2	Resposta ao degrau do modelo simulado $ARX1$ controlado . . . . .	32
5.3	Resposta ao degrau do modelo $ARX2$ . . . . .	33
5.4	Resposta ao degrau do modelo simulado $ARXsim$ . . . . .	34
5.5	Simulação do estimador do modelo $SUB1$ . . . . .	35
5.6	Resposta ao degrau do modelo $SUB1$ controlado com observador . . . . .	35
5.7	Simulação do estimador do modelo $ARX1$ . . . . .	36
5.8	Resposta ao degrau do modelo $ARX1$ controlado com observador . . . . .	36
5.9	Simulação do estimador do modelo $ARX2$ . . . . .	37
5.10	Resposta ao degrau do modelo $ARX2$ controlado com observador . . . . .	37



---

5.11	Simulação do estimador do modelo <i>ARXsim</i> . . . . .	38
5.12	Resposta ao degrau do modelo <i>ARXsim</i> controlado com observador . . . .	38
6.1	Resposta ao degrau do sistema controlado <i>SUB1</i> . . . . .	40
6.2	Resposta ao degrau do sistema controlado <i>ARX1</i> . . . . .	41
6.3	Resposta ao degrau do sistema controlado <i>AR21</i> . . . . .	41
6.4	Resposta à escadaria do sistema controlado <i>SUB1</i> . . . . .	42
6.5	Resposta à escadaria do sistema controlado <i>ARX1</i> . . . . .	42
6.6	Resposta à escadaria do sistema controlado <i>ARX2</i> . . . . .	43
6.7	Resposta à mudança de parâmetros do modelo <i>SUB1</i> . . . . .	43
6.8	Resposta à mudança de parâmetros do modelo <i>ARX1</i> . . . . .	44
6.9	Resposta à mudança de parâmetros do modelo <i>ARX2</i> . . . . .	44
6.10	Resposta à perturbação externa do modelo <i>SUB1</i> . . . . .	45
6.11	Resposta à perturbação externa do modelo <i>ARX1</i> . . . . .	45
6.12	Resposta à perturbação externa do modelo <i>ARX2</i> . . . . .	46

# Lista de Tabelas

# Lista de Abreviaturas e Siglas

**SIGLA**   NOME EXPANDIDO – do inglês *SI Gl A*

# Lista de Símbolos

## Símbolos Matemáticos

$\mathbb{R}$  conjunto dos números reais

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Levitação a ar . . . . .	1
1.2	Objetivos . . . . .	2
1.2.1	Objetivo Geral . . . . .	2
1.2.2	Objetivos Específicos . . . . .	2
1.3	Organização do trabalho . . . . .	3
<b>2</b>	<b>Revisão Bibliográfica</b>	<b>4</b>
2.1	Fundamentação Teórica . . . . .	4
2.1.1	Modelos de Sistemas Dinâmicos . . . . .	4
2.1.1.1	Equação Diferencial Ordinária . . . . .	4
2.1.1.2	Funções de Transferência . . . . .	5
2.1.1.3	Espaço de Estados . . . . .	6
2.1.1.4	Modelo ARX . . . . .	7
2.1.2	CrITÉRIOS de Estabilidade de sistemas discretos . . . . .	7
2.1.3	Identificação de Sistemas e Estimação de Parâmetros . . . . .	10
2.1.3.1	Visão Geral . . . . .	10
2.1.3.2	Identificação por Mínimos Quadrados . . . . .	11
2.1.3.3	Filtro de Kalman . . . . .	13
2.1.3.4	Identificação por Subespaços . . . . .	15
2.2	Trabalhos Relacionados . . . . .	17
<b>3</b>	<b>Modelagem e Simulação</b>	<b>19</b>
3.1	Descrição da Planta . . . . .	19
3.2	Modelo Matemático . . . . .	19

---

3.3	Estimação da Velocidade do Vento . . . . .	20
3.4	Simulação do Túnel de Vento no Simulink . . . . .	21
<b>4</b>	<b>Identificação do Túnel de Ar</b>	<b>22</b>
4.1	Escolha de estrutura . . . . .	22
4.1.1	Mínimos Quadrados . . . . .	22
4.1.2	Subespaços . . . . .	23
4.2	Experimento . . . . .	23
4.3	Estimação . . . . .	24
4.3.1	Mínimos Quadrados . . . . .	25
4.3.2	Subespaços . . . . .	25
4.4	Validação . . . . .	26
<b>5</b>	<b>Projeto de Controlador com Alocação de Polos Baseado nos Modelos Identificados</b>	<b>30</b>
5.1	Alocação de Polos com Realimentação de Estados . . . . .	30
5.1.1	Modelo <i>SUB1</i> . . . . .	30
5.1.2	Modelo <i>ARX1</i> . . . . .	32
5.1.3	Modelo <i>ARX2</i> . . . . .	33
5.1.4	Modelo <i>ARXsim</i> . . . . .	33
5.2	Projeto do Estimador de Estados . . . . .	34
5.2.1	Modelo <i>SUB1</i> . . . . .	34
5.2.2	Modelo <i>ARX1</i> . . . . .	35
5.2.3	Modelo <i>ARX2</i> . . . . .	36
5.2.4	Modelo <i>ARXsim</i> . . . . .	37
<b>6</b>	<b>Avaliação Experimental do Sistema de Controle</b>	<b>39</b>
6.1	Descrição e objetivos experimentais . . . . .	39
6.2	Resultados Experimentais . . . . .	40
6.2.1	Resultados da Resposta ao degrau unitário . . . . .	40
6.2.2	Resultados da Resposta à escadaria . . . . .	42
6.2.3	Resultados do Teste de Robustez à mudança de Parâmetros . . . . .	43
6.2.4	Resultados do Teste de Robustez à perturbação externa . . . . .	45
6.3	Discussão . . . . .	46

---

<b>7 Conclusão</b>	<b>47</b>
7.1 Considerações Finais . . . . .	47
7.2 Trabalhos Futuros . . . . .	47
<b>Referências Bibliográficas</b>	<b>48</b>

# Capítulo 1

## Introdução

### 1.1 Levitação a ar

O túnel de vento é uma instalação utilizada para estudar o desempenho aerodinâmico de objetos. Ele é feito de um duto de tamanho adequado para o objeto testado pelo qual o ar flui ao ser empurrado por uma turbina. Os testes feitos variam de velocidade de acordo com o que se deseja estudar.

Para que os testes sobre objetos tenham resultados precisos e replicáveis a velocidade dos ventos sobre eles deve ser controlável e atinja o objeto sem turbulência, apesar de haver testes onde isso é proposital. O túnel é construído com um dispositivo de fluxo laminar que possibilita reduzir o fluxo turbulento gerado pela turbina [9].

Os testes geralmente são feitos em modelos em escala de aviões, carros, naves espaciais, e para estudo da aerodinâmica de objetos na física. Os túneis podem ser verticais ou horizontais, apesar de os túneis horizontais existirem em maior quantidade. Um teste comum é feito em modelos de aviões onde se medem as propriedades das asas como envergadura ao ser submetida a velocidades específicas do vento, força de elevação para diferentes velocidades e em geral sua aerodinâmica. Em Dayton, Ohio [7], foi construído um túnel de vento vertical que possibilita que sejam feitos testes específicos para aeronaves. Um objeto é posto em queda livre virtual devido a velocidade do vento impulsionado por uma turbina e se observa o seu movimento na situação. Este teste foi essencial para a prototipagem de aviões militares e comerciais desde os anos 50.

Hoje os testes ainda são responsáveis por economizar milhões de reais na fase de prototipagem em testes aerodinâmicos, no desenvolvimento de trens de alta velocidade[3].



Em objetos que se movem em alta velocidade o vento se torna um fator preocupante, uma rajada de vento pode ser o fator determinante em um descarrilhamento de um trem, os testes possibilitam que os cientistas analisem comportamento de suas criações em situações previsíveis e os ajuda a antecipar possíveis problemas.

No entanto, existe um problema no teste aerodinâmico utilizando túneis de vento. Para os testes de queda livre como o de aeronaves, o controle de altitude do objeto é complexo. É possível obter o número de Reynolds, uma variável sem dimensão que possibilita prever o padrão de fluxo de fluídos, e é possível medir a velocidade do vento atuando em cima do objeto, porém, a real dificuldade se encontra no fato de que quando o a força do vento atua em cima de um objeto ele tende a girar, [4] de acordo com suas características aerodinâmicas, o que não é possível de se controlar sem um atuador no objeto.

Tendo em vista que o controle de altura de um objeto tendo como único atuador o vento, e que um modelo de atuação da força do vento sobre um objeto é complexo e de baixa fidelidade, e que precisamos de um modelo que possibilite um controle de altitude preciso e robusto, temos como opção a modelagem caixa-preta, conhecida como identificação de sistemas.

## 1.2 Objetivos

### 1.2.1 Objetivo Geral

O objetivo deste trabalho é identificar e controlar a altura de uma bola de tênis de mesa dentro de um túnel de vento vertical.

### 1.2.2 Objetivos Específicos

Para alcançar o objetivo geral deste projeto devemos primeiro alcançar os seguintes objetivos específicos.

- Construir uma plataforma de experimento no formato de túnel de vento.
- Medir a velocidade do vento em alturas variadas dentro do túnel de vento.
- Criar um simulador para o túnel de vento.

- Executar testes para identificação usando como sinal de controle um sinal binário pseudo aleatório (PRBS).
- Identificar o sistema por mínimos quadrados.
- Identificar o sistema por subespaços.
- Validar os modelos obtidos.
- Obter um controlador para os dois modelos usando alocação de polos.

## 1.3 Organização do trabalho

Este trabalho se encontra organizado em seis capítulos com várias subseções.

- Capítulo 2: é a revisão bibliográfica com as seções fundamentação teórica onde conceitos fundamentais ao entendimento do trabalho são explicados, tratamos dos modelos de sistemas dinâmicos, dos critérios de estabilidade de sistemas discretos, e da identificação de sistemas e estimação de parâmetros. Falamos ainda de trabalhos relacionados onde o trabalho de outros autores é relacionado a este.
- Capítulo 3: fala sobre a modelagem e simulação do túnel de vento, onde se explica o processo de modelagem utilizado e a construção de um simulador no Simulink.
- Capítulo 4: trata da identificação do túnel usando mínimos quadrados e subespaços, explicando os procedimentos usados e o método de validação do modelo.
- Capítulo 5: é projeto de controladores, onde fazemos o projeto do controlador para adequar o sistema à requisitos de sua resposta à entradas.
- Capítulo 6: trata da avaliação experimental do sistema de controle, onde o sistema é testado para avaliar sua robustez à interferências.
- Capítulo 7: é a conclusão do trabalho onde as considerações finais e trabalhos futuros serão discutidos.

# Capítulo 2

## Revisão Bibliográfica

Nesta seção serão apresentadas duas subseções importantes para o entendimento deste trabalho, a revisão bibliográfica que trata de conceitos fundamentais ao entendimento do trabalho, e trabalhos relacionados que trata de outros trabalhos que são pertinentes à este.

### 2.1 Fundamentação Teórica

Nesta subseção tratamos de conceitos essenciais para o entendimento deste trabalho. Falaremos sobre alguns modelos matemáticos, critérios de estabilidade de sistemas discretos, e métodos de identificação de sistemas e estimação de parâmetros, em especial identificação por mínimos quadrados e por subespaços.

#### 2.1.1 Modelos de Sistemas Dinâmicos

O modelo matemático de um sistema dinâmico é uma forma de descrever uma parte do fenômeno físico que queremos controlar. Nesta seção serão apresentados alguns dos vários modelos matemáticos relacionados ao projeto.

##### 2.1.1.1 Equação Diferencial Ordinária

Um sistema dinâmico pode ser modelado usando somente a descrição matemática do fenômeno desejado. Usamos como exemplo a modelagem do amortecimento em uma roda de um carro, visto na figura [2.1](#).

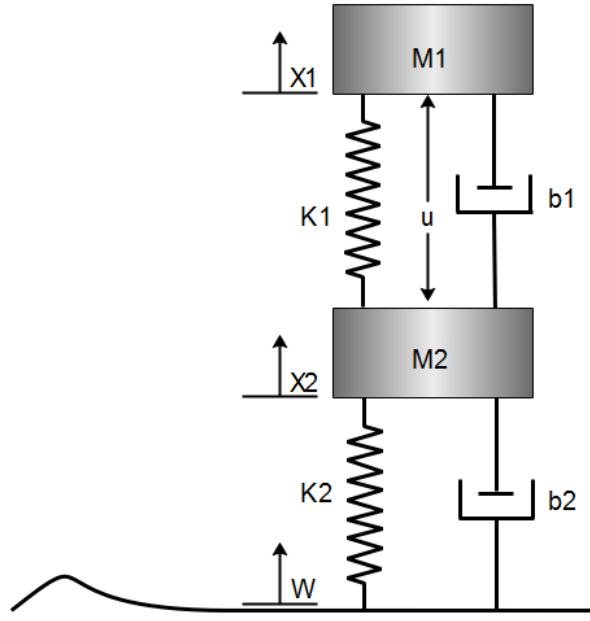


Figura 2.1: Esquema do amortecedor da roda de um carro

Onde  $X_1$  e  $X_2$  são a altura das massas  $M_1$  e  $M_2$ , massa do carro e massa da roda, respectivamente.  $K_1$  é o efeito elástico do amortecedor e  $K_2$  é o efeito elástico da roda.  $b_1$  é o efeito amortecedor da suspensão e  $b_2$  é o efeito amortecedor da roda. Esse sistema pode ser modelado em uma E.D.O. (Equação Diferencial Ordinária) usando as leis de Newton, visto nas equações (2.1) e (2.2).

$$M_1 \ddot{X}_1 = -b_1(\dot{X}_1 - \dot{X}_2) - K_1(X_1 - X_2) + U \quad (2.1)$$

$$M_2 \ddot{X}_2 = b_1(\dot{X}_1 - \dot{X}_2) + K_1(X_1 - X_2) + b_2(\dot{W} - \dot{X}_2) + K_2(W - X_2) - U \quad (2.2)$$

### 2.1.1.2 Funções de Transferência

A função de transferência é uma equação que descreve o comportamento dinâmico de um sistema relacionando uma entrada com uma saída. Ela é a transformada de Laplace da resposta ao impulso do sistema.

Usando o mesmo sistema da figura 2.1 e aproveitando as equações (2.1) e (2.2) podemos encontrar as equações de transferência referentes às saídas  $X_1$  e  $X_2$ .

Aplicamos a transformada de Laplace nas equações  $X_1$  e  $X_2$  para obter:

$$(M_1 s^2 + b_1 s + K_1)X_1(s) - (b_1 s + K_1)X_2(s) = U(s) \quad (2.3)$$

$$-(b_1s + K_1)X_1(s) + (M_2s^2 + (b_1 + b_2)s + (K_1 + K_2))X_2(s) = (b_2s + K_2)W(s) - U(s) \quad (2.4)$$

Fazendo as devidas manipulações matemáticas chegamos à função de transferência.

$$\begin{aligned} G_1(s) &= \frac{X_1(s) - X_2(s)}{W(s)} = \frac{(M_1 + M_2)s^2 + b_2s + K_2}{\Delta} \\ G_2(s) &= \frac{X_1(s) - X_2(s)}{W(s)} = \frac{-M_1b_2s^3 - M_1K_2s^2}{\Delta} \\ \Delta &= (M_1s^2 + b_1s + K_1) \cdot (M_2s^2 + (b_1 + b_2)s + (K_1 + K_2)) - (b_1s + K_1) \cdot (b_1s + K_1) \end{aligned} \quad (2.5)$$

Com as equações descritas em (2.5) temos uma equação representando a transferência da entrada para a saída.

### 2.1.1.3 Espaço de Estados

A representação em espaço de estados é uma forma mais conveniente para representar sistemas no domínio do tempo quando existe mais de uma entrada ou saída do que a função de transferência. Um modelo linear é representado tipicamente no seguinte formato:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (2.6)$$

Onde  $x \in \mathbb{R}^n$  é o vetor de estado n-dimensional.  $\dot{x} = dx/dt$ ,  $u(t) \in \mathbb{R}^p$  é o vetor de entradas formado por r funções temporais,  $y(t) \in \mathbb{R}^q$  é o vetor m-dimensional de saídas e  $A \in \mathbb{R}^{n \times n}$ ,  $B \in \mathbb{R}^{n \times p}$ ,  $C \in \mathbb{R}^{q \times n}$  e  $D \in \mathbb{R}^{q \times p}$  são as matrizes do sistema constantes.

Podemos gerar uma representação de espaço de estados a partir da E.D.O. do sistema. Vamos utilizar novamente o sistema da figura 2.1. Definimos primeiramente os estados do vetor  $x$ ,  $x_1 = X_1$ ,  $x_2 = \dot{X}_1$ ,  $x_3 = Y_1$  e  $x_4 = \dot{Y}_1$ . Onde  $Y_1 = X_1 - X_2$ , geramos então as matrizes do espaço de estados usando as equações (2.1) e (2.2):

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{-b_1b_2}{M_1M_2} & 0 & \left[ \frac{b_1}{M_1} \left( \frac{b_1}{M_1} + \frac{b_1}{M_2} + \frac{b_2}{M_2} \right) - \frac{K_1}{M_1} \right] & \frac{-b_1}{M_1} \\ \frac{b_2}{M_2} & 0 & - \left( \frac{b_1}{M_1} + \frac{b_1}{M_2} + \frac{b_2}{M_2} \right) & 1 \\ \frac{K_2}{M_2} & 0 & - \left( \frac{K_1}{M_1} + \frac{K_1}{M_2} + \frac{K_2}{M_2} \right) & 0 \end{bmatrix} \quad (2.7)$$

$$B = \begin{bmatrix} 0 & 0 \\ \frac{1}{M_1} & \frac{b_1 b_2}{M_1 M_2} \\ 0 & \frac{-b_2}{M_2} \\ \left(\frac{1}{M_1} + \frac{1}{M_2}\right) & \frac{-\tilde{K}_2}{M_2} \end{bmatrix} \quad (2.8)$$

$$\begin{bmatrix} \dot{X}_1 \\ \ddot{X}_1 \\ \dot{Y}_1 \\ \ddot{Y}_1 \end{bmatrix} = A \begin{bmatrix} X_1 \\ \dot{X}_1 \\ Y_1 \\ \dot{Y}_1 \end{bmatrix} + B \begin{bmatrix} U \\ W \end{bmatrix} \quad (2.9)$$

$$Y = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_1 \\ \dot{X}_1 \\ Y_1 \\ \dot{Y}_1 \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ W \end{bmatrix} \quad (2.10)$$

#### 2.1.1.4 Modelo ARX

O modelo ARX SISO linear tem o seguinte formato:

$$\begin{aligned} y(t) + a_1 y(t-1) + a_2 y(t-2) + \dots + a_{na} y(t-na) = \\ b_1 u(t) + b_2 u(t-1) + \dots + b_{nb} u(t-nb+1) + e(t) \end{aligned} \quad (2.11)$$

Onde  $y$  é a entrada,  $u$  é a saída e  $e$  é o ruído. Isso implica que a saída  $y(t)$  é uma predição a partir de uma média ponderada de entradas e saídas passadas. Os parâmetros  $a_{na}$  e  $b_{nb}$  podem ser estimados utilizando mínimos quadrados numa coleção de dados entrada-saída.

### 2.1.2 Critérios de Estabilidade de sistemas discretos

O conceito de estabilidade é extremamente importante para a análise de sistemas dinâmicos. Primeiramente definimos estabilidade de acordo com mudanças nas condições iniciais. Consideremos a equação (2.12).

$$x(k+1) = f(x(k), k) \quad (2.12)$$

Sejam  $x_0(k)$  e  $x(k)$  suas soluções quando as condições iniciais são  $x_0(k_0)$  e  $x(k_0)$ . Definimos:

- Estabilidade: A solução  $x_0(k)$  é estável se para um dado  $\epsilon > 0$  existe um  $\delta(\epsilon, k_0)$  tal que para todas as soluções com  $\|x(k_0) - x_0(k_0)\| < \delta$  são tais que  $\|x(k) - x_0(k)\| < \epsilon$  para todos os  $k \geq k_0$ .

- Estabilidade Assintótica: A solução  $x_0(k)$  é assintoticamente estável se é estável e um  $\delta$  pode ser escolhido tal que  $\|x(k_0) - x_0(k_0)\| < \delta$  que implica que  $\|x(k) - x_0(k)\| \rightarrow 0$  quando  $k \rightarrow \infty$ .

Existem outros tipos de estabilidade que são de interesse:

- Estabilidade BIBO (Bounded-Input Bounded-Output): Um sistema linear invariante no tempo é definido BIBO estável se dado um sinal de entrada limitado ele produz uma saída limitada para cada valor inicial.

Dado que a estabilidade de um sistema é importante para o seu estudo, os métodos para determinar a sua estabilidade são de grande interesse. Os seguintes são alguns dos métodos utilizados para determinar a estabilidade de um sistema:

- Cálculo dos autovalores da matriz A da representação de espaços de estado.
- Métodos baseados nas propriedades dos polinômios característicos.
- O método do lugar das raízes
- O método de Lyapunov

O cálculo dos autovalores de uma matriz de ordem maior que 2 à mão não é conveniente e em alguns casos é mais fácil calcular a equação característica da forma:

$$A(z) = a_0 z^n + a_1 z^{n-1} \cdots + a_n = 0 \quad (2.13)$$

E investigar suas raízes utilizando o método do lugar das raízes onde o critério de estabilidade muda para sistemas discretos determinando que para o sistema ser estável todas as raízes devem estar dentro do círculo unitário.

Outro método para determinar a estabilidade de um sistema é o critério de Jury, versão discreta do critério de Routh-Hurwitz. A tabela de Jury é formada da seguinte forma:

$$H(z) = \frac{b(z)}{a(z)} = \frac{b(z)}{a_0 z^n + a_1 z^{n-1} + \cdots + a_n} \quad (2.14)$$

$$\begin{array}{c|cccc}
a_0 & a_1 & \dots & a_n \\
a_n & a_{n-1} & \dots & a_0 \\
b_0 & b_1 & \dots & \\
b_{n-1} & b_{n-2} & \dots & \\
c_0 & c_1 & \dots & \\
\vdots & \vdots & & 
\end{array} \quad (2.15)$$

onde

$$\begin{aligned}
b_0 &= a_0 - \frac{a_n}{a_0} a_n \\
b_1 &= a_1 - \frac{a_n}{a_0} a_{n-1} \\
&\vdots \\
b_k &= a_k - \frac{a_n}{a_0} a_{n-k} \\
&\vdots \\
c_k &= b_k - \frac{b_{n-1}}{b_0} b_{n-1-k}
\end{aligned} \quad (2.16)$$

Com a tabela formada aplicamos o critério de Jury que diz: se  $a_0 > 0$ , então todas as raízes estarão dentro do círculo unitário se e somente se todos os termos da primeira coluna das linhas impares forem positivos. Se nenhum elemento da primeira coluna das linhas impares for nulo, o número de raízes fora do círculo unitário é igual ao número de elementos negativos.

O segundo método de Lyapunov é outra ferramenta útil para determinar a estabilidade de sistemas dinâmicos não lineares. A ideia é introduzir uma função de energia generalizada chamada função de Lyapunov que é zero no ponto de equilíbrio e positiva em outras posições. O equilíbrio será estável se pudermos mostrar que a função de Lyapunov diminui ao longo das trajetórias do sistema.

O primeiro passo é encontrar a função de Lyapunov definida como segue:

$V(x)$  é uma função de Lyapunov do sistema

$$x(k+1) = f(x(k)) \quad f(0) = 0 \quad (2.17)$$

se:

1.  $V(x)$  é contínuo em  $x$  e  $V(0)=0$
2.  $V(x)$  é definida positiva



3.  $\Delta V(x) = V(f(x)) - V(x)$  é definida negativa

Definimos:

• Teorema de estabilidade de Lyapunov: A solução  $x(k) = 0$  é assintoticamente estável se existir uma função de Lyapunov para o sistema da equação (2.17). Se:

$$0 < \varphi(\|x\|) < V(x) \quad (2.18)$$

onde  $\varphi(\|x\|) \rightarrow \infty$  quando  $\|x\| \rightarrow \infty$ , então a solução é assintoticamente estável para todas as condições iniciais.

A grande dificuldade do teorema de Lyapunov é encontrar uma função de Lyapunov adequada. No entanto, para sistemas lineares como o da equação (2.19), é fácil determinar uma função de Lyapunov quadrática.

$$x_0(k+1) = \Phi x_0(k) \quad x_0(0) = a_0 \quad (2.19)$$

Tomemos  $V(x) = x^T P_x$  como candidato para a função de Lyapunov. O incremento de  $V$  é dado por:

$$\begin{aligned} \Delta V(x) &= V(\Phi x) - V(x) = x^T \Phi^T P \Phi x - x^T P x \\ &= x^T (\Phi^T P \Phi - P) x = -x^T Q x \end{aligned} \quad (2.20)$$

Para que  $V$  seja uma função de Lyapunov é necessário e suficiente que exista uma matriz definida positiva  $P$  que satisfaça a equação (2.21).

$$\Phi^T P \Phi - P = -Q \quad (2.21)$$

### 2.1.3 Identificação de Sistemas e Estimação de Parâmetros

A maior parte das técnicas de controle aplicadas na indústria dependem do conhecimento de um modelo matemático do sistema a ser controlado. Apresentamos alguns dos modelos na seção 2.1.1 e nesta seção apresentaremos alguns dos métodos para obtermos os modelos matemáticos utilizados neste trabalho.

#### 2.1.3.1 Visão Geral

A identificação de sistemas e estimação de parâmetros trata de métodos e práticas que permitem construir modelos dinâmicos de um sistema real a partir de experimentos .

Muitas vezes um sistema construído que precisa ser controlado não pode ser modelado devido à limitações matemáticas ou imprecisão na interação dos componentes. Nestes casos utilizamos um método de identificação de sistemas para obter um modelo matemático. A identificação de sistemas se baseia em testar a resposta do sistema à certas entradas e a partir das respostas aproximar o modelo matemático de forma satisfatória.

### 2.1.3.2 Identificação por Mínimos Quadrados

O método de mínimos quadrados é um dos mais conhecidos e utilizados em várias áreas da ciência e tecnologia. A partir de uma coleção de dados experimentais é possível encontrar um modelo matemático ARX.

Para um sistema SISO em que não conhecemos o seu modelo matemático podemos descrever o seu comportamento durante o teste como um conjunto de funções entrada-saída como na equação (2.11), pois à partir do conhecimento que temos da modelagem física do sistema podemos deduzir aproximadamente de quantos valores passados de  $y$  e de  $u$  o nosso sistema estudado depende.

Com os vetores entrada e saída em mão montamos uma matriz de regressores  $y$  e  $u$  no seguinte formato:

$$Y = \begin{bmatrix} y(n) & y(n-1) & \dots & y(n-i) \\ y(n+1) & y(n) & \dots & y(n-i+1) \\ \vdots & \vdots & \ddots & \vdots \\ y(n+k) & y(n+k-1) & \dots & y(n+k-i) \end{bmatrix} \quad (2.22)$$

$$U = \begin{bmatrix} u(n) & u(n-1) & \dots & u(n-j) \\ u(n+1) & u(n) & \dots & u(n-j+1) \\ \vdots & \vdots & \ddots & \vdots \\ u(n+k) & u(n+k-1) & \dots & u(n+k-j) \end{bmatrix} \quad (2.23)$$

$$\psi = \begin{bmatrix} Y & U \end{bmatrix} \quad (2.24)$$

Onde  $y(n-i)$  é o primeiro valor do vetor de saídas  $y$  e  $u(n-i)$  é o primeiro valor do vetor de entradas  $u$ ,  $i$  é a quantidade de regressores  $y$  que influenciam na saída  $y$  e  $j$  é a quantidade de regressores  $u$  que influenciam na saída  $y$ ,  $n+k$  é o número de valores do

vetor de entrada.

Podemos então relacionar o vetor de saídas com a matriz de regressores  $A$  da seguinte forma:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \psi \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix} \quad (2.25)$$

$$\hat{y} = \psi \hat{\theta}$$

Onde  $\hat{y}$  é um vetor que depende da matriz de regressores  $A$  e do vetor  $\hat{\theta}$ . Conhecemos  $\hat{y}$  e  $A$ , queremos determinar  $\hat{\theta}$ . Desde que  $X$  seja não singular é possível determinar o vetor de parâmetros invertendo a matriz:

$$\theta = \psi^{-1}y \quad (2.26)$$

A matriz  $\psi$ , no entanto, não é invertível e para realizarmos  $\psi^{-1}$  utilizamos a pseudo inversa, que será utilizada em diversas partes desse trabalho, representada por  $\{\cdot\}^\dagger$ , onde  $\{\cdot\}$  é a matriz que receberá a operação pseudo inversa.

$$A^\dagger = [A^T A]^{-1} A^T \quad (2.27)$$

E a equação (2.26) se torna:

$$\theta = \psi^\dagger y \quad (2.28)$$

A equação (2.26) é a única equação que satisfaz todos os regressores do sistema de equações formado a partir da equação (2.11). Assumindo que conhecemos  $\hat{\theta}$  e que existe um resíduo  $\xi$  entre o valor observado  $y$  e o valor obtido a partir do vetor de regressores  $\psi$  da forma:

$$y = \psi^T \hat{\theta} + \xi \quad (2.29)$$

Este resíduo é o menor possível devido ao fato de que a equação (2.28) ser demonstravelmente minimizada.

Apresentamos aqui um pseudo algoritmo para a identificação por mínimos quadrados:

---

**Algoritmo 1:** IDENTIFICAÇÃO POR MÍNIMOS QUADRADOS

---

**Entrada:** Vetor de entradas  $U[N]$ , vetor de saídas  $Y[N]$ , quantidade de regressores de  $Y$   $n$ , quantidade de regressores de  $U$   $m$

**Saída:** Vetor  $\hat{\theta}$

**início**

    Faça a matriz de regressores  $\psi$  da equação (2.24)

    Calcule  $\theta$  com a equação (2.28)

**fim**

**retorna**  $\underline{\theta}$

---

### 2.1.3.3 Filtro de Kalman

O filtro de Kalman, também conhecido como estimador linear quadrático, é um algoritmo que utiliza uma série de medidas tomadas ao longo do tempo, que contém ruídos e outros tipos de erros, e produz um estimador capaz de estimar com maior confiança um certo valor. Ele é utilizado em uma grande variedade de aplicações de engenharia, como localização GPS e navegação, sua aplicação mais famosa foi no programa espacial Apollo, e é um tópico especialmente importante para a teoria de sistemas de controle, devido a sua capacidade de estimação de sistemas não lineares.

**Visão Geral** O filtro de Kalman utiliza um modelo dinâmico de um sistema, suas entradas de controle, e medidas de sensores para gerar uma estimativa das grandezas medidas do sistema. Usando um modelo recursivo para obter as estimativas, medidas passadas, ele consegue obter uma estimativa mais fiel ao sistema real do que utilizando somente uma medida. O filtro funciona em duas etapas, uma de propagação, onde se utiliza a estimativa do estado anterior para se obter uma estimativa do estado atual, e uma de assimilação, onde a estimativa do estado atual é combinada com a observação do estado real para se obter um modelo de estimativa mais preciso.

Usaremos a nomenclatura de Aguirre, onde  $t_1$  é substituído pela iteração atual indicada por  $k$ , e o  $t_2$  é substituído pela próxima iteração  $k+1$ . A notação  $(t_1 - t_1)$  é substituída por um sinal '+' para indicar o instante  $t_i$  após ter sido incluída a informação em  $t_i$ . Da mesma forma será utilizado um sinal '-' para indicar a grandeza que se refere ao instante  $t_i$  antes de ter sido incluída a informação referente àquele instante. A equação que rege a

propagação é a seguinte:

$$\hat{x}_{k+1}^- = \Phi_k \hat{x}_k^+ + \Gamma_k u_k \quad (2.30)$$

**Etapa de propagação** Conhecendo a função de densidade de probabilidade de  $x_k^+$ , indicada por  $f_k \sim \mathcal{N}(\bar{x}_k^+, P_k^+)$ , deseja-se encontrar a função de densidade de probabilidade de  $x_{k+1}^-$ . Ou seja, na etapa de propagação deseja-se saber o que acontece à  $f_k$  ao ser propagado pela equação (2.30). Assumimos que  $f_k$  é gaussiana e portanto  $f_-$  também será, deste modo basta determinar  $\bar{x}_{k+1}^-$  e  $P_{k+1}^-$  contidos em  $f_- \sim \mathcal{N}(\bar{x}_{k+1}^-, P_{k+1}^-)$  para caracterizar  $f_-$ .

Segundo Aguirre [1] encontramos:

$$\bar{x}_{k+1}^- = \Phi_k \bar{x}_k^+ + \Gamma_k u_k \quad (2.31)$$

$$P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T \quad (2.32)$$

A equação mostra que ao longo da etapa de propagação a incerteza aumenta devido à presença de ruído no modelo dinâmico usado.

**Etapa de assimilação** Vimos que na etapa de propagação o vetor de estado  $x_k^+$  é propagado para a próxima iteração resultando em  $x_{k+1}^-$ . A segunda etapa, a de assimilação, ocorre com a chegada de nova informação na iteração  $k+1$ . O objetivo é a determinação de  $f_+ \sim \mathcal{N}(\bar{x}_{k+1}^+, P_{k+1}^+)$  a partir de  $f_-$  e da medição na iteração  $y_{k+1}$ . De forma semelhante à etapa de propagação, devemos encontrar  $\bar{x}_{k+1}^+$  e  $P_{k+1}^+$ . Após os devidos passos encontramos:

$$\bar{x}_{k+1}^+ = \bar{x}_{k+1}^- + K_{k+1} [y_{k+1} - H_{k+1} \bar{x}_{k+1}^-] \quad (2.33)$$

$$P_{k+1}^+ = P_{k+1}^- - K_{k+1} H_{k+1} P_{k+1}^- \quad (2.34)$$

$$K_{k+1} = P_{k+1}^- H_{k+1}^T [H_{k+1} P_{k+1}^- H_{k+1}^T + R_{k+1}]^{-1} \quad (2.35)$$

Com estas equações completamos o conjunto de equações necessárias para entender o filtro de Kalman.

### 2.1.3.4 Identificação por Subespaços

A identificação por subespaços é um algoritmo, que, a partir de uma sequência de dados experimentais, é capaz de gerar um modelo em espaço de estados. O procedimento aqui descrito é explicado com mais detalhes por Rico [10].

Com o sistema a ser identificado em mãos precisamos executar um ou mais experimentos medindo o sinal de entrada e o sinal da saída do sistema. Para que a identificação funcione adequadamente o sinal de entrada deve atender a condição de persistência de excitação [8], um sinal binário pseudo aleatório atende esse requisito.

Aplicamos o sinal no sistema e medimos a sua resposta ao longo de um tempo  $t$  com um tempo de amostragem  $t_s$  e obtemos dois vetores,  $u[N]$ , sinal PRBS de entrada aplicados ao sistema, e  $y[N]$  que é a saída medida do sistema,  $N$  é a quantidade de medidas tomadas. Com esses dois vetores geramos um conjunto de 4 matrizes em blocos de Hankel.

$$U_{0|2i-1} = \frac{\begin{pmatrix} u_0 & u_1 & \dots & u_{j-1} \\ u_1 & u_2 & \dots & u_j \\ \vdots & \vdots & \ddots & \vdots \\ u_{i-1} & u_i & \dots & u_{i+j-2} \\ u_i & u_{i+1} & \dots & u_{i+j-1} \\ u_{i+1} & u_{i+2} & \dots & u_{i+j} \\ \vdots & \vdots & \ddots & \vdots \\ u_{2i-1} & u_{2i} & \dots & u_{2i+j-2} \end{pmatrix}}{U_f} = \frac{U_p}{U_f} \quad (2.36)$$

Onde  $i = nm$ ,  $n$  é a ordem da matriz em blocos de Hankel desejada e  $m$  é o número de entradas, e  $j = N - 2i + 1$ ,  $U_f$  é a matriz em blocos das entradas futuras,  $U_p$  é a matriz em blocos das entradas passadas. O mesmo é feito para  $y[N]$  gerando  $Y_f$  e  $Y_p$ . Com essas matrizes em mãos geramos  $W_p = [U_p \ Y_p]^T$  e podemos encontrar  $O_i$  fazendo:

$$O_i = Y_{f/U_f} W_p \quad (2.37)$$

$Y_{f/U_f} W_p$  denota uma projeção oblíqua de  $Y_f$  sobre  $W_p$  ao longo de  $U_f$  calculado por:

$$O_i = [Y_f/U_f^\perp][W_p/U_f^\perp]^\dagger W_p \quad (2.38)$$

Em seguida fazemos uma decomposição SVD de  $O_i$ :

$$O_i = USV^T = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} \quad (2.39)$$

Com  $O_i$  decomposto calculamos  $\Gamma_i$  da seguinte forma:

$$\Gamma_i = U_1 S_1^{1/2} \quad (2.40)$$

Tiramos a raiz quadrada de uma matriz com o seguinte procedimento:

$$A = QPQ^{-1} \quad (2.41)$$

Onde  $Q$  e  $P$  fazem uma decomposição de  $S_1$  e  $P$  é escolhido de forma que se conheça a sua raiz quadrada.

$$S_1 = (QP^{1/2}Q^{-1})^2 \quad (2.42)$$

$$S_1^{1/2} = QP^{1/2}Q^{-1} \quad (2.43)$$

Com  $\Gamma_i$  em mãos podemos identificar as matrizes  $A$  e  $C$  do sistema.  $\Gamma_i$  é equivalente à matriz de observabilidade do sistema, então  $C$  é a primeira linha de  $\Gamma_i$  e  $A$  pode ser encontrado por:

$$A = \underline{\Gamma_i^\dagger} \overline{\Gamma_i} \quad (2.44)$$

Onde a matriz sublinhada significa que foi removida a última linha e a matriz sobrelinhada significa que foi removida a primeira linha.

Encontramos  $B$  e  $D$  com a equação:

$$\Gamma_i^\perp Y_f U_f^\dagger = \Gamma_i^\perp H_i^d \quad (2.45)$$

Renomeamos os termos dessa equação da seguinte forma:

$$Q = \begin{pmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{i-2}B & CA^{i-3}B & CA^{i-4}B & \dots & D \end{pmatrix} \quad (2.46)$$

$$\begin{pmatrix} M_1 & M_2 & \dots & M_i \end{pmatrix} = \begin{pmatrix} L_1 & L_2 & \dots & L_i \end{pmatrix} Q \quad (2.47)$$

Reescrevemos essa equação:

$$\begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_i \end{pmatrix} = \begin{pmatrix} L_1 & L_2 & \dots & L_{i-1} & L_i \\ L_2 & L_3 & \dots & L_i & 0 \\ L_3 & L_4 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ L_i & 0 & \dots & 0 & 0 \end{pmatrix} \begin{pmatrix} I_l & 0 \\ 0 & \underline{\Gamma_i} \end{pmatrix} \begin{pmatrix} D \\ B \end{pmatrix} \quad (2.48)$$

E dessa forma obtemos todas as matrizes do sistema identificado.

**Algoritmos** Para facilitar o entendimento colocamos em forma de algoritmo os passos que devem ser tomados para identificar um sistema por subespaços.

---

**Algoritmo 2: IDENTIFICAÇÃO POR SUBESPAÇOS**

---

**Entrada:** Vetor de entradas  $U[N]$ , vetor de saídas  $Y[N]$ , ordem determinada do sistema  $n$

**Resultado:** Matrizes do espaço de estados do sistema  $A$ ,  $B$ ,  $C$  e  $D$

- 1 Gerar as matrizes de bloco de Hankel  $Y_p$ ,  $Y_f$ ,  $U_p$  e  $U_f$ , usando a equação (2.36)
- 2 Calcular  $O_i$  usando as equações (2.37) e (2.38)
- 3 Decomposição SVD de  $O_i$  (2.39)
- 4 Calcular  $\Gamma_i$  (2.40)
- 5 Obter  $A$  e  $C$  (2.44)
- 6 Obter  $B$  e  $D$  (2.48)

**retorna**  $A, B, C, D$

---

## 2.2 Trabalhos Relacionados

Nesta subseção falaremos dos trabalhos feitos por outros autores que se relacionam a este.

Em 2009, Jernigan et al. [6], publicaram um artigo falando sobre uma experiência que fizeram na construção e utilização de uma experiência de laboratório remota. Eles escolheram como experimento a levitação de uma bola de praia de plástico usando um soprador de ar. Um grupo de alunos foi encarregado de modelar, simular e controlar o sistema usando o laboratório remoto. Este trabalho foi essencial para o nosso entendi-



mento da modelagem da levitação de uma bola por fluxo de ar e especialmente do tipo de medida necessária para criar um simulador do sistema. Diferente deste trabalho Jernigan et al. não fez identificação do sistema e optou por fazer modelagem matemática.

Em 2017, Chacon et al. [5], relatam em um artigo a construção de um túnel de vento para o ensino de engenharia de controle. Eles tem o mesmo objetivo deste trabalho em relação à construir uma plataforma de experimentos barata e reproduzível mas diferem porque aplicam a plataforma para aprendizado à distância. No artigo eles descrevem um processo de identificação caixa preta usando um sinal binário pseudo aleatório bastante parecido com o que será utilizado neste trabalho, mas tratam a identificação como um exercício a ser aplicado aos alunos.

Em sua dissertação de mestrado Ricco [10] fala sobre identificação por subespaços com uma linguagem clara e de fácil entendimento. O seu trabalho se relaciona a este devido a identificação por subespaços apesar de não trabalhar com um sistema específico. Ele foi essencial ao entendimento de identificação por subespaços.

# Capítulo 3

## Modelagem e Simulação

Neste capítulo será descrito a modelagem do sistema, o processo de estimação de velocidade do vento, e a simulação do túnel de vento no Simulink Matlab.

### 3.1 Descrição da Planta

O sistema em questão se trata de um tubo PVC de 40 mm de 120 cm, ele é afixado em uma plataforma plástica através de uma base e de elásticos para estabilização. Na ponta inferior do tubo se encontra uma turbina de avião RC que empurra o ar para dentro do tubo elevando uma bola de tênis de mesa, na ponta superior do tubo se encontra um sensor infravermelho de distância. A turbina é controlado por um Arduino Mega 2560.

### 3.2 Modelo Matemático

[IB: inserir figura do sistema]

Há duas forças atuantes na esfera, a gravidade que a puxa para baixo e a força de empuxo gerada pelo vento. Obtemos a seguinte equação do movimento:

$$m\ddot{h} = F = \frac{1}{2} \cdot C_a \cdot \rho \cdot A \cdot (v_{ar} - \dot{h})^2 - m \cdot g \quad (3.1)$$

onde  $m$  é a massa da esfera,  $h$  é a posição vertical da esfera no tubo,  $\rho$  é a densidade do ar,  $A$  é a área da esfera em contato com o fluxo de ar,  $v_{ar}$  é a velocidade do ar dentro do tubo e  $C_a$  é o coeficiente aerodinâmico da esfera. O coeficiente aerodinâmico depende da velocidade relativa entre a esfera e o vento, mas para as velocidades baixas

de vento que estamos utilizando esse valor pode ser considerado constante. Consideramos  $\alpha = \frac{1}{2} \cdot C_a \cdot \rho \cdot A$ :

$$\ddot{h} = \frac{\alpha}{m} \cdot (v_{ar} - \dot{h})^2 - g \quad (3.2)$$

### 3.3 Estimação da Velocidade do Vento

Para criarmos um simulador do sistema descrito anteriormente precisamos saber o valor de  $v_{ar}$ . Ele determina a velocidade do vento para diferentes valores de tensão e altura da bola. Não foi possível adquirir um sensor de velocidade do ar devido a problemas de localização geológica, portanto, foi necessário usar de engenhosidade para adquirir a velocidade do ar em diversas alturas.

Foram adquiridas diversas bolas de tênis de mesa e se injetou uma mistura de cola e água nelas para aumentar o seu peso. Foram escolhidos 6 pesos diferentes e medidos em balança com precisão de duas casas decimais. Então para cada peso foi medida a tensão necessária para que se alcance as 6 alturas dentro da região de funcionamento do sensor, 10, 20, 30, 40, 50, 60 cm. Tendo medido a altura e a tensão necessária para alcançar as alturas, bastou utilizar a equação (3.2) fazendo  $\ddot{h} = \dot{h} = 0$  para obter a velocidade do vento relacionada com uma altura e uma tensão.

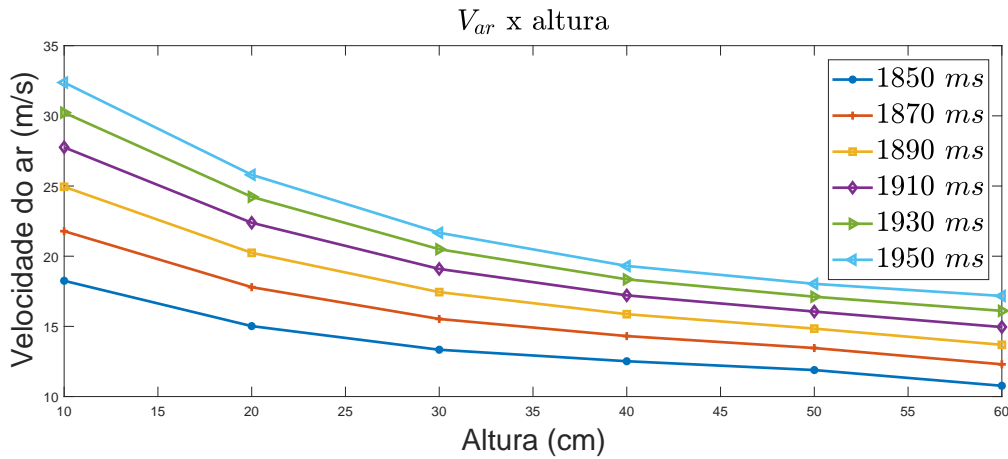


Figura 3.1: Gráfico de  $v_{ar}$  x altura

Podemos ver que a figura 3.1 lembra um gráfico similar encontrado em [6] onde o autor utiliza um sensor de velocidade de vento e mostra uma curva similar, para uma mesma tensão aplicada na turbina quanto mais elevada se encontra a bola menor vai ser a velocidade do ar que se choca com ela.

### 3.4 Simulação do Túnel de Vento no Simulink

Criamos um modelo de simulação à partir da equação 3.2 e utilizamos os dados adquiridos em 3.1 para fazer um ajuste de curva que alimenta o modelo com os valores da velocidade do vento para determinada altura e tensão no motor.

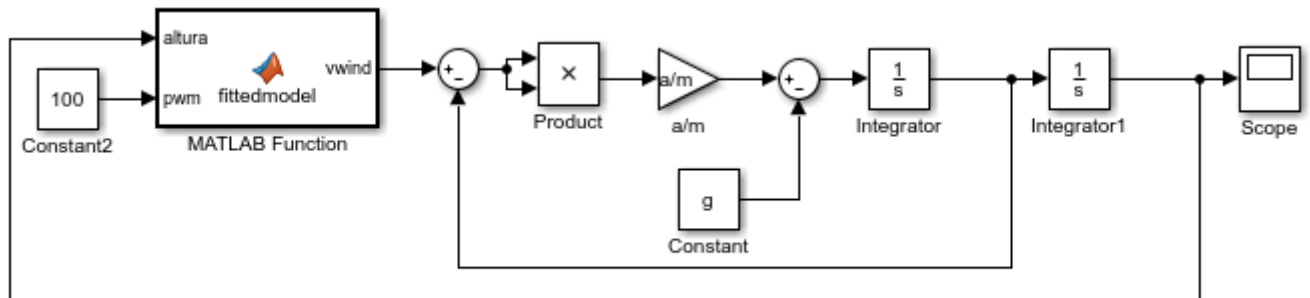


Figura 3.2: Simulador do túnel de vento

# Capítulo 4

## Identificação do Túnel de Ar

Neste capítulo, será explicada a identificação do sistema do túnel de ar. Para fazer a identificação caixa-preta de um sistema é necessário fazer um estudo prévio do funcionamento dele, conhecer entradas e saídas, obter um modelo matemático se possível, mesmo sem ter todos os parâmetros.

### 4.1 Escolha de estrutura

Para a identificação do sistema escolhemos fazer um teste que atenda a condição PE [8] usando um sinal de entrada PRBS e como saída a altura da bola medida pelo sensor infravermelho. Existem vários métodos para fazer a escolha da estrutura de identificação de um sistema como o critério de informação de Akaike [2], entretanto, decidimos escolher a ordem do sistema fazendo uma análise visual da autocorrelação dos resíduos e da resposta ao degrau do modelo identificado.

#### 4.1.1 Mínimos Quadrados

A identificação por Mínimos Quadrados gera um sistema do tipo ARX como visto na equação (2.11). A escolha da estrutura neste caso se dá escolhendo uma quantidade de regressores da saída e uma quantidade de regressores da entrada.

Para a escolha da estrutura do modelo do sistema foi feito o seguinte procedimento:

1. Escolher uma quantidade de regressores de  $y$
2. Escolher uma quantidade de regressores de  $u$

3. Fazer a identificação por Mínimos Quadrados com os regressores de  $y$  e  $u$
4. Analisar a autocorrelação dos resíduos  $\xi = y - \Psi\hat{\theta}$

Este procedimento foi repetido para valores de 1 a 10 para ambos os regressores para escolhermos quantos zeros e polos precisamos para uma estrutura que tem a seguinte forma:

$$Y[k] = \frac{A[k]}{B[k]} \quad (4.1)$$

$$A[k] = a_1U[k-1] + a_2U[k-2] + \dots + a_nU[k-n] \quad (4.2)$$

$$B[k] = 1 + b_1Y[k-1] + b_2Y[k-2] + \dots + b_mY[k-m] \quad (4.3)$$

Onde  $n$  é o número de zeros e  $m$  é o número de polos. Ao final foram identificadas duas ordens de sistema que serão comparadas posteriormente, uma com 3 polos e 3 zeros, modelo *ARX1*, e uma com 5 polos e 5 zeros, modelo *ARX2*. Geramos um terceiro modelo *ARX* aplicando a identificação no simulador gerando um modelo, *ARXsim*, com 3 polos e 3 zeros para comparação.

### 4.1.2 Subespaços

A identificação por Subespaços gera um sistema em espaço de estados como visto na equação (2.6). Neste tipo de identificação a escolha da estrutura é feita decidindo a ordem do sistema e a ordem da matriz em blocos de Hankel. E similarmente ao procedimento feito na seção 4.1.1 foram variadas as ordens do sistema e da matriz em blocos de Hankel. Foi escolhida ordem 3 para o sistema e ordem 15 para a matriz em blocos de Hankel usada para identificar o sistema. Vemos na figura 4.4 a auto correlação dos resíduos do sistema identificado.

## 4.2 Experimento

O experimento para identificação precisa de um sinal que atenda a condição PE. Para tanto foi gerado um sinal PRBS (sinal binário pseudo aleatório) que é suficientemente adequado para extrair a dinâmica do sistema. O sinal é aplicado ao sistema através do Arduino e os sinais são medidos com um tempo de amostragem de 50 ms.

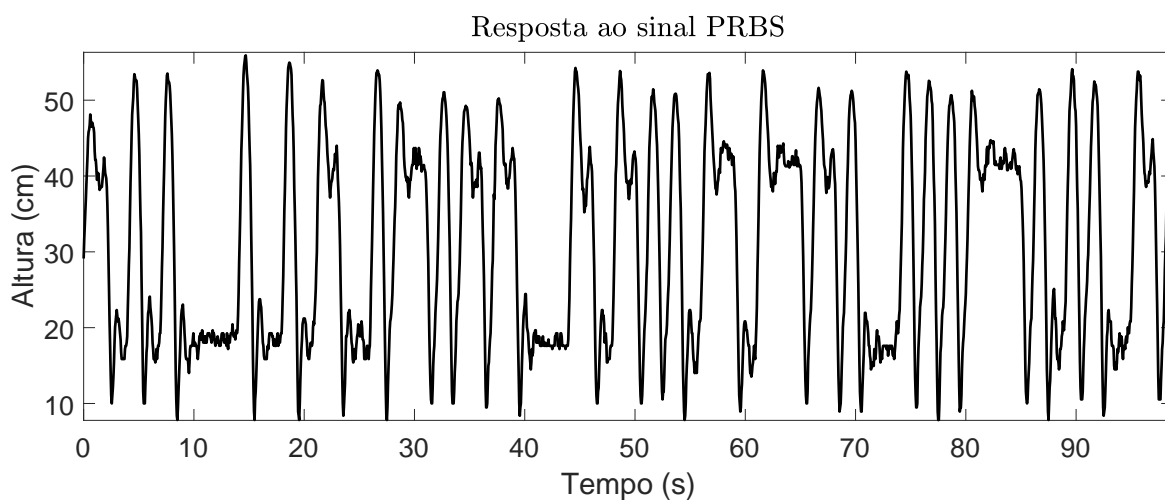


Figura 4.1: Gráfico da saída do sistema ao aplicar o sinal PRBS com tempo de amostragem de 50 ms

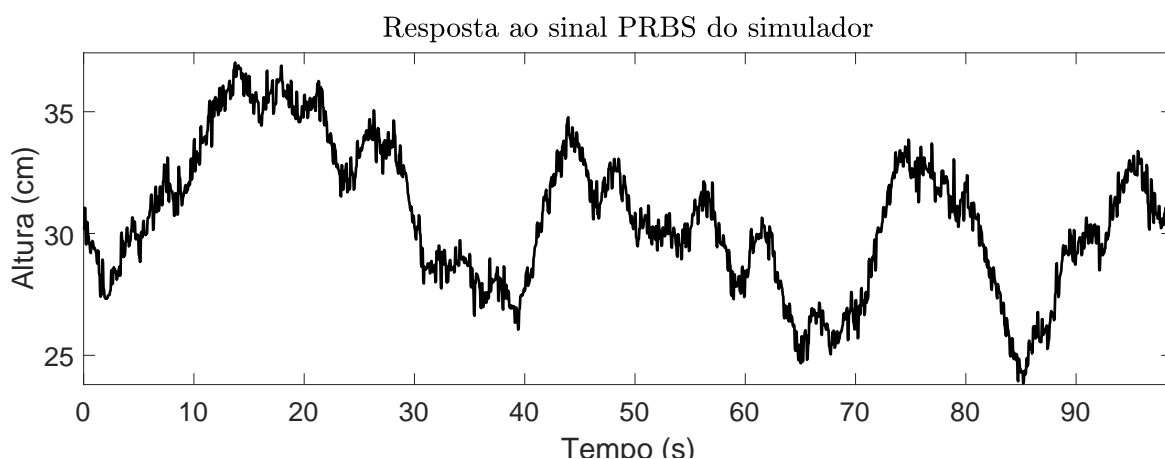


Figura 4.2: Gráfico da saída do simulador ao aplicar o sinal PRBS com tempo de amostragem de 50 ms

A figura 4.1 mostra a resposta do sistema ao sinal PRBS que foi aplicado ao sistema, e a figura 4.2 mostra a resposta do simulador. Vemos que a resposta do modelo simulado é muito diferente do sistema real.

### 4.3 Estimação

Com a resposta ao sinal PRBS em mãos podemos identificar o sistema. Usando o método dos mínimos quadrados e o método de identificação por subespaços.

### 4.3.1 Mínimos Quadrados

Para fazer a identificação por mínimos quadrados geramos a matriz de regressores  $\psi$  da equação (2.24) e usamos a equação (2.26) para obter os coeficientes dos regressores. Obtemos os coeficientes do modelo  $ARX1$ :

$$a_1 = 0.0017 \quad a_2 = 0.0011 \quad a_3 = 0.0550 \quad (4.4)$$

$$b_1 = -1.6309 \quad b_2 = 0.5828 \quad b_3 = 0.0965 \quad (4.5)$$

Obtemos os coeficientes do modelo  $ARX2$ :

$$a_1 = -0.0029 \quad a_2 = -0.0040 \quad a_3 = 0.0143 \quad a_4 = 0.0057 \quad a_5 = 0.0725 \quad (4.6)$$

$$b_1 = -1.5506 \quad b_2 = 0.7601 \quad b_3 = -0.5321 \quad b_4 = 0.4854 \quad b_5 = -0.0897 \quad (4.7)$$

Obtemos os coeficientes do modelo  $ARXsim$ :

$$a_1 = -0.0037 \quad a_2 = 0.0017 \quad a_3 = 0.0066 \quad (4.8)$$

$$b_1 = -1.0108 \quad b_2 = 0.4531 \quad b_3 = -0.4341 \quad (4.9)$$

### 4.3.2 Subespaços

Para identificar o sistema usando subespaços utilizamos o algoritmo mostrado na seção 2.1.3.4 com uma matriz em blocos de Hankel de ordem 15 para encontrar um sistema de ordem 3. Identificamos o seguinte modelo,  $SUB1$ , com o formato da equação (2.6):

$$A = \begin{bmatrix} 0.9761 & 0.1933 & -0.0438 \\ -0.1817 & 0.9841 & -0.1489 \\ 0.0840 & 0.3107 & 0.6994 \end{bmatrix} \quad (4.10)$$

$$B = \begin{bmatrix} 0.1466 \\ 0.2515 \\ 0.8460 \end{bmatrix} \quad (4.11)$$

$$C = \begin{bmatrix} -1.0104 & -0.3354 & 0.2496 \end{bmatrix} \quad (4.12)$$



$$D = \begin{bmatrix} 0.0010 \end{bmatrix} \quad (4.13)$$

## 4.4 Validação

Ao identificar um sistema precisamos validar o modelo obtido para garantir que ele é adequado para representar o sistema real. Para isso fazemos uma análise da auto correlação dos resíduos  $\xi = y - \Psi\hat{\theta}$  para o ARX e  $\xi = y - y_{sim}$  para o espaço de estados. Vemos nas figuras 4.3, 4.4 e 4.6, que os resíduos do modelo ARX são muito menos correlacionados do que os do modelo em espaço de estados. Isso acontece porque para o modelo ARX estamos analisando os seus regressores com a matriz  $\Psi$  que os gera, o que retorna uma resposta muito mais próxima do sistema. Já para o espaço de estados estamos analisando a simulação do sistema, que apresenta pequenas diferenças em relação ao sistema real, como a ausência de ruído.

Olhamos também a resposta ao degrau do sistema, figuras 4.7, 4.8, 4.9 e 4.10, e vemos que para os dois sistemas identificados ela tem uma oscilação similar ao sistema real. Concluimos, através da análise da auto correlação dos resíduos e da análise da resposta ao degrau dos sistemas identificados, que os modelos são adequados para mostrar o funcionamento do sistema.

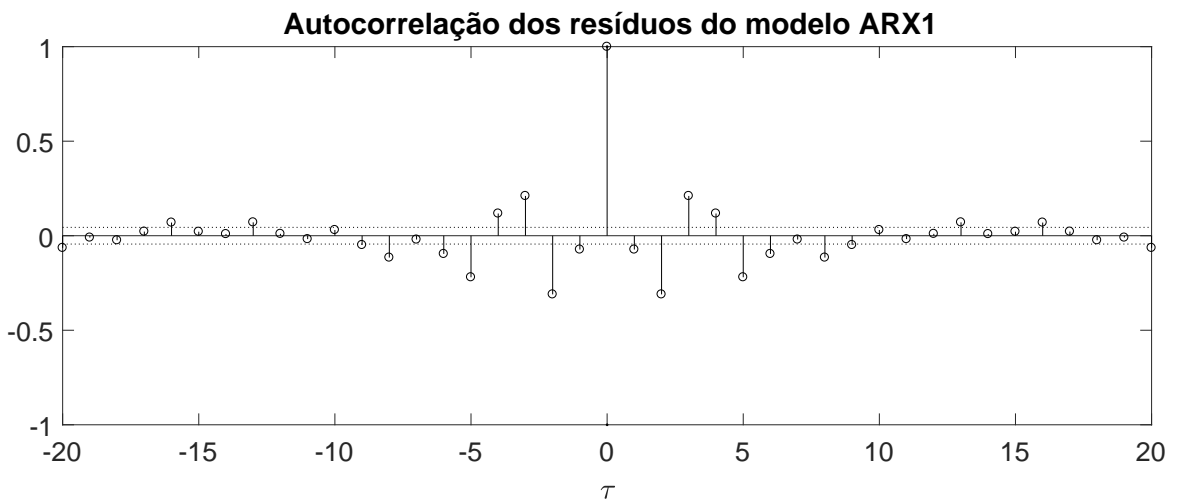
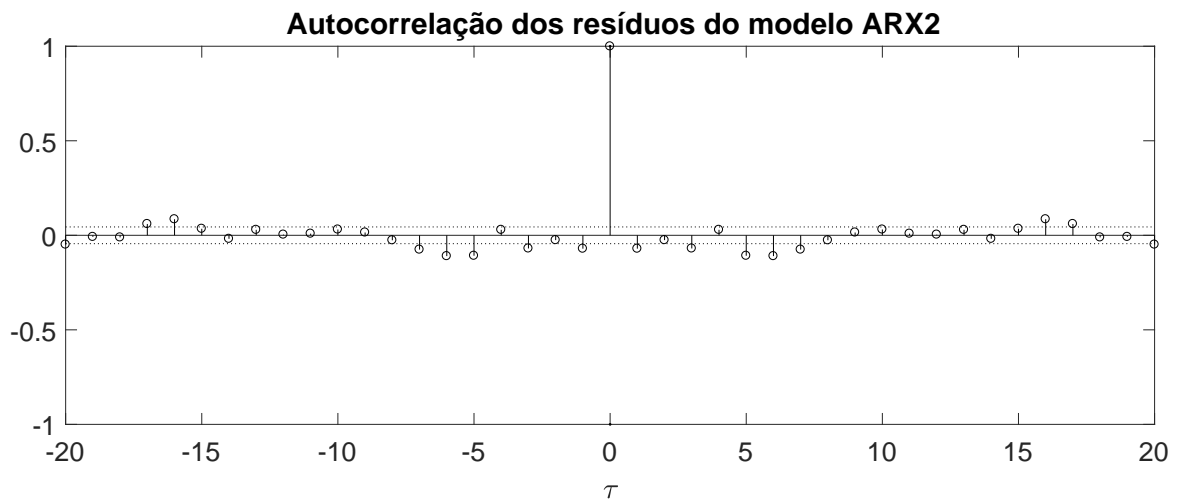
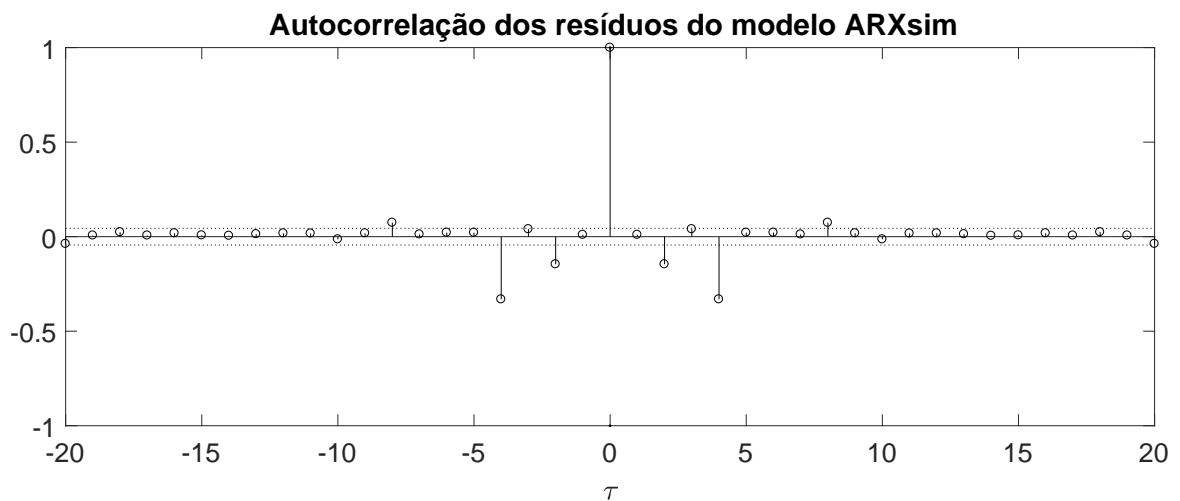
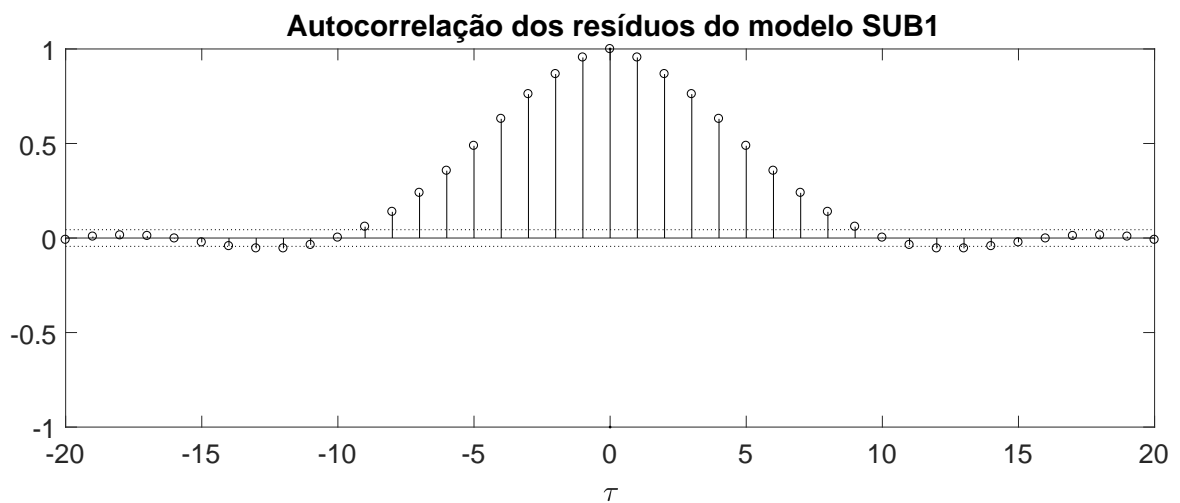
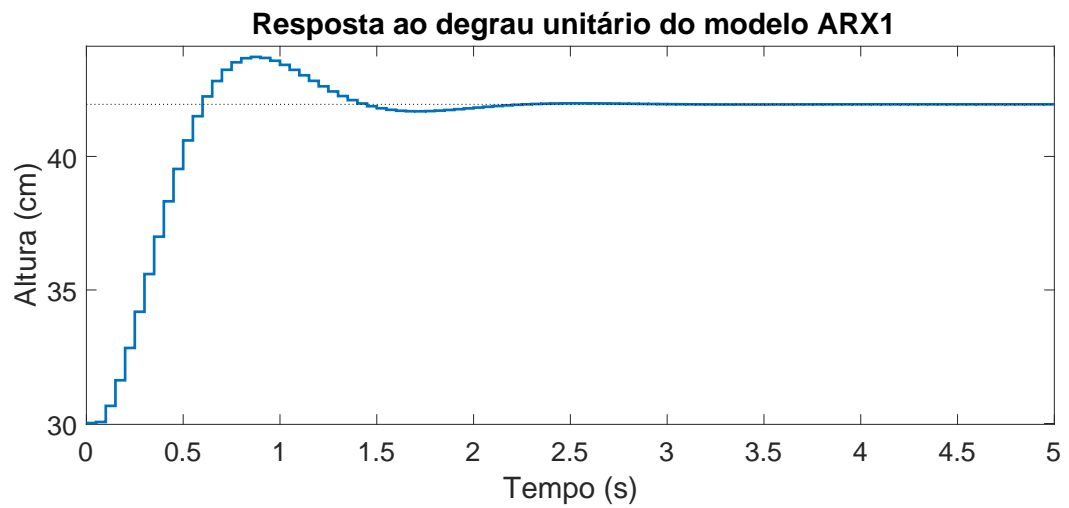
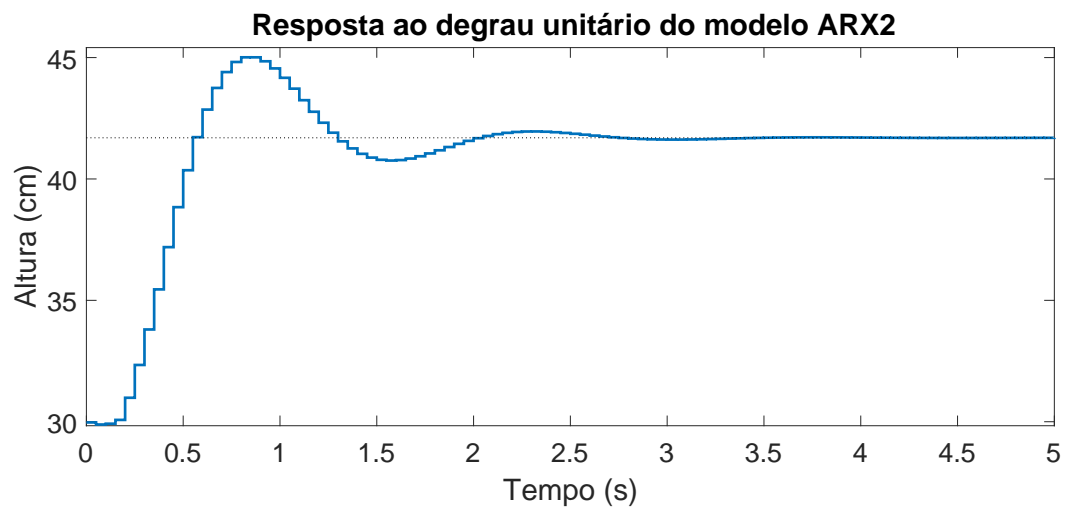
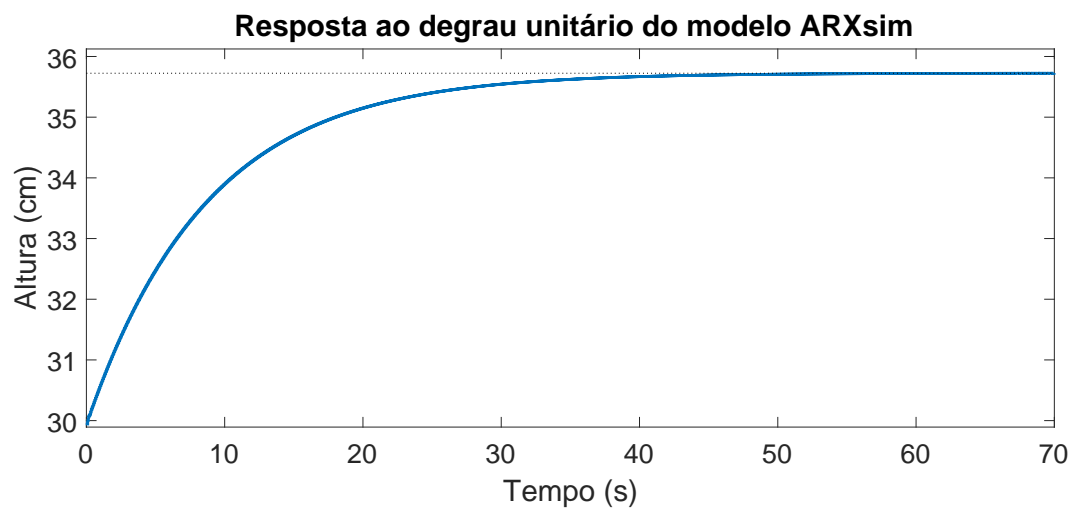


Figura 4.3: Autocorrelação dos resíduos do modelo *ARX1*

Figura 4.4: Autocorrelação dos resíduos do modelo *ARX2*Figura 4.5: Autocorrelação dos resíduos do modelo *ARXsim*Figura 4.6: Autocorrelação dos resíduos do modelo *SUB1*

Figura 4.7: Resposta ao degrau unitário do modelo  $ARX1$ Figura 4.8: Resposta ao degrau unitário do modelo  $ARX2$ Figura 4.9: Resposta ao degrau unitário do modelo  $ARX_{sim}$

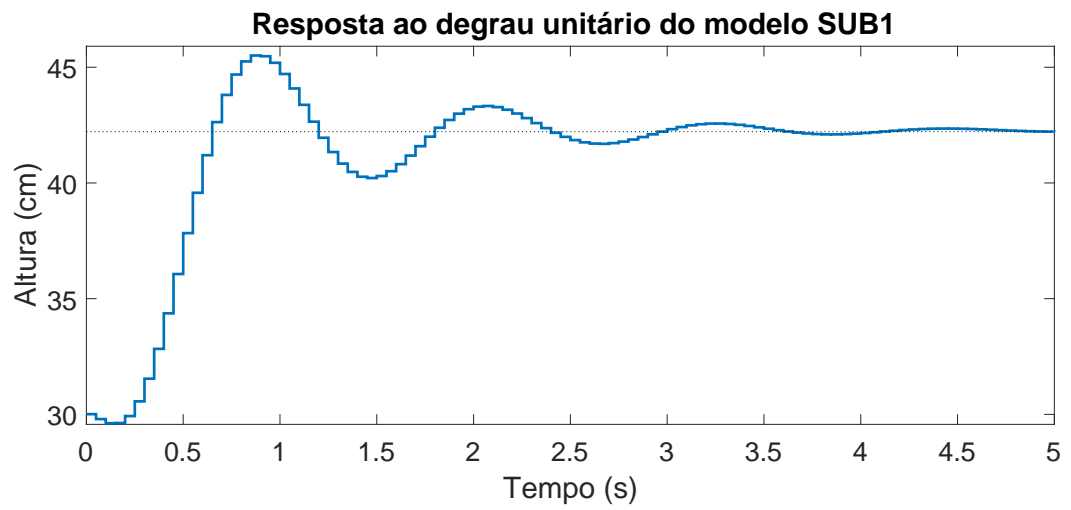


Figura 4.10: Resposta ao degrau unitário do modelo *SUB1*

## Capítulo 5

# Projeto de Controlador com Alocação de Polos Baseado nos Modelos Identificados

Com o sistema identificado, podemos finalmente controlá-lo, faremos isso através de realimentação de estados para os quatro modelos identificados.

### 5.1 Alocação de Polos com Realimentação de Estados

Para fazer o controle por realimentação de estados precisamos passar os 3 modelos ARX obtidos para modelos em espaço de estados. Fazemos isso usando a função do Matlab `tf2ss`, onde entramos com o nominador e denominador do modelo ARX e a função nos retorna as matrizes A, B, C e D do sistema em espaço de estados.

Queremos controlar os 4 modelos fazendo alocação de polos através da realimentação de estados. Projetamos os controladores para que atendam como requisitos um sobrevalor menor que 4% do valor final e tempo de assentamento menor que 2 segundos quando recebe uma entrada do tipo degrau unitário.

#### 5.1.1 Modelo *SUB1*

Para atender os requisitos de tempo de assentamento  $t_s < 2\text{ s}$  e sobrevalor máximo de  $ovs < 4\%$  calculamos os valores de  $\zeta$  e  $\omega_n$  usados na equação característica de um sistema de segunda ordem (5.1), usando as equações (5.2) e (5.3).

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (5.1)$$

$$ovs = e^{\frac{\zeta\pi}{\sqrt{1-\zeta^2}}} < 4\% \quad (5.2)$$

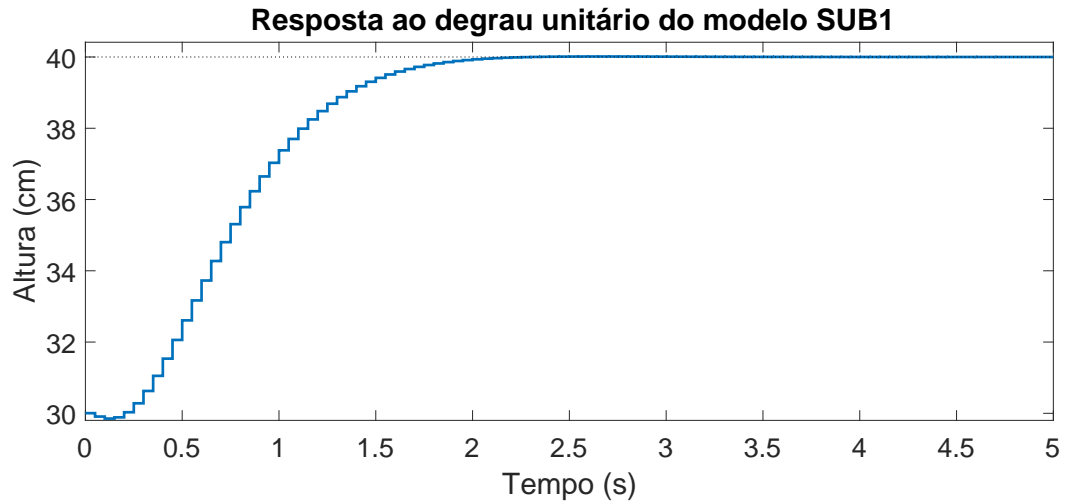
$$t_s = \frac{4}{\zeta\omega_n} < 2 \text{ s} \quad (5.3)$$

Encontramos  $\zeta > 0.7157$  e  $\omega_n > 2.7945$ , e escolhemos os valores de  $\zeta = 0.9$  e  $\omega_n = 4$ . Ao aplicar esses valores na equação (5.1) obtemos os polos que precisamos alocar no domínio  $s$ ,  $s_1 = -3.6000 + 1.7436i$  e  $s_2 = -3.6000 - 1.7436i$ . Como o modelo  $SUB'$  tem ordem 3 precisamos de um terceiro polo distante da parte real dos anteriores, escolhemos  $s_3 = -8$ . Passamos os três polos para o domínio  $z$  e obtemos  $z_1 = 0.8321 + 0.0727i$ ,  $z_2 = 0.8321 - 0.0727i$  e  $z_3 = 0.6703$ .

Para fazer a alocação de polos usamos a equação de Lyapunov (5.4):

$$(A - B\bar{k})T = TF \quad (5.4)$$

Onde  $A$  e  $B$  são matrizes do sistema,  $\bar{k}$  é um vetor arbitrário escolhido de forma que o par  $F$  e  $\bar{k}$  seja observável,  $F$  é uma matriz em blocos diagonais com os polos que se deseja alocar e  $k = \bar{k} * T^{-1}$  nos retorna o vetor de realimentação de estados  $k$ . Obtemos  $k = [0.7552, 0.2637, 0.1750]$  Com essa realimentação de estados obtemos a seguinte resposta ao degrau:

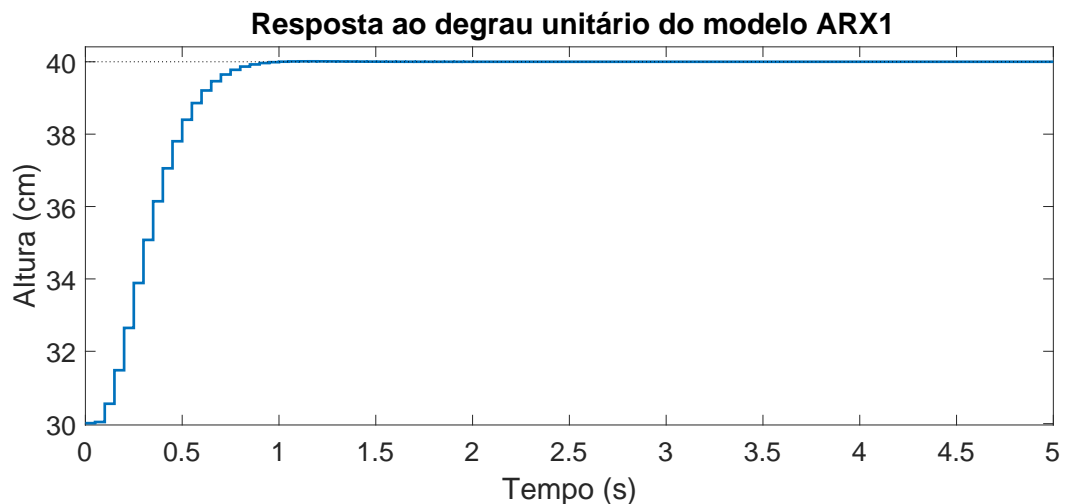
Figura 5.1: Resposta ao degrau unitário do modelo simulado *SUB1* controlado

A resposta ao degrau do sistema como vista na figura 5.1 apresenta um tempo de assentamento de 1.8 segundos e um sobrevalor de 0.14%.

### 5.1.2 Modelo *ARX1*

Para o modelo *ARX1* escolhemos os valores de  $\zeta = 0.9$  e  $\omega_n = 7$ . Ao aplicar esses valores na equação (5.1) obtemos os polos que precisamos alocar no domínio  $s$ ,  $s_1 = -6.3000 + 3.0512i$  e  $s_2 = -6.3000 - 3.0512i$ , para o polo distante alocamos  $s_3 = -15$ . Que no domínio  $z$  são  $z_1 = 0.7213 + 0.1109i$ ,  $z_2 = 0.7213 - 0.1109i$  e  $z_3 = 0.4966$ . Repetimos o mesmo procedimento de alocação de polos usado no modelo *SUB1* e obtemos  $k = [-0.2840, 0.6313, -0.3481]$ .

O sistema realimentado tem a seguinte resposta ao degrau:

Figura 5.2: Resposta ao degrau do modelo simulado *ARX1* controlado

A resposta ao degrau da figura 5.2 apresenta um tempo de assentamento de 0.8 segundos e um sobrevalor de 0,125%.

### 5.1.3 Modelo $ARX2$

Para o modelo  $ARX2$  escolhemos  $\zeta = 0.9$  e  $\omega_n = 9$ . Aplicando na equação (5.1) obtemos os polos para alocar no domínio  $s$ ,  $s_1 = -8.1000 + 3.9230i$  e  $s_2 = -8.1000 - 3.9230i$ . Este modelo ainda tem 3 polos extras que devem ser alocados distantes dos polos  $s_1$  e  $s_2$ , escolhemos  $s_3 = -14$ ,  $s_4 = -14.1$  e  $s_5 = -14.2$ . Que no domínio  $z$  são  $z_1 = 0.6542 + 0.1300i$ ,  $z_2 = 0.6542 - 0.1300i$ ,  $z_3 = 0.4966$ ,  $z_4 = 0.4941$  e  $z_5 = 0.4916$ . Fazemos a alocação de polos da mesma forma que foi feita para o modelo  $SUB1$  obtemos  $k = [-1.2401, 2.3566, -1.2062, -0.0018, 0.0360]$ .

O sistema controlado tem a seguinte resposta ao degrau unitário:

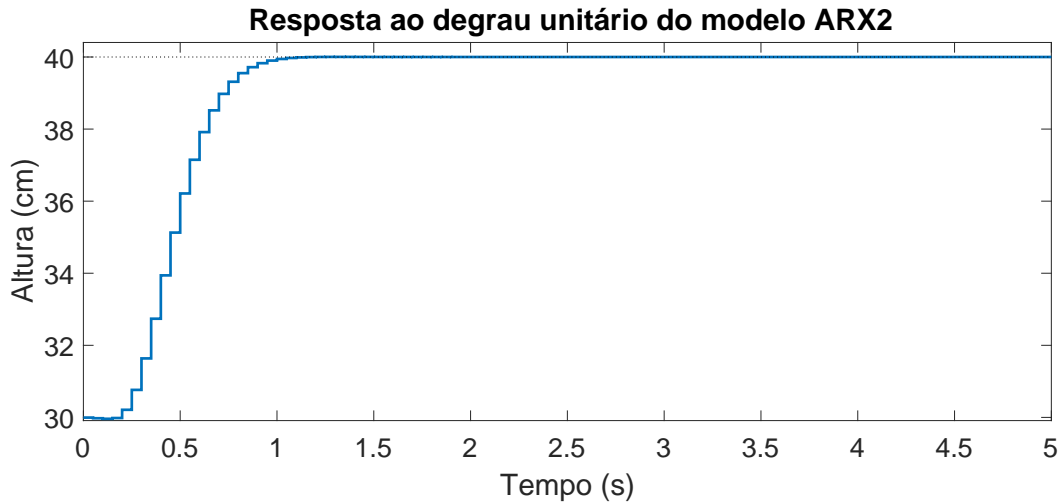


Figura 5.3: Resposta ao degrau do modelo simulado  $ARX2$  controlado

Vemos na figura 5.3 que a resposta ao degrau unitário do sistema controlado tem tempo de assentamento de 0.9 segundos e sobrevalor de 0.0326%

### 5.1.4 Modelo $ARXsim$

Para o modelo  $ARXsim$  escolhemos  $\zeta = 0.9$  e  $\omega_n = 9$ . Aplicados na equação (5.1) obtemos os polos no domínio  $s$ ,  $s_1 = -3.6000 + 1.7436i$  e  $s_2 = -3.6000 - 1.7436i$ , e escolhemos o polo distante  $s_3 = -8$ . No domínio  $z$  os polos são  $z_1 = 0.8321 + 0.0727i$ ,  $z_2 = 0.8321 - 0.0727i$  e  $z_3 = 0.4966$ . Fazemos a alocação de polos como foi feito para



o modelo *SUB1* e obtemos  $k = [-1.1499, 1.0710, 0.0876]$ . O sistema controlado tem a seguinte resposta ao degrau unitário:

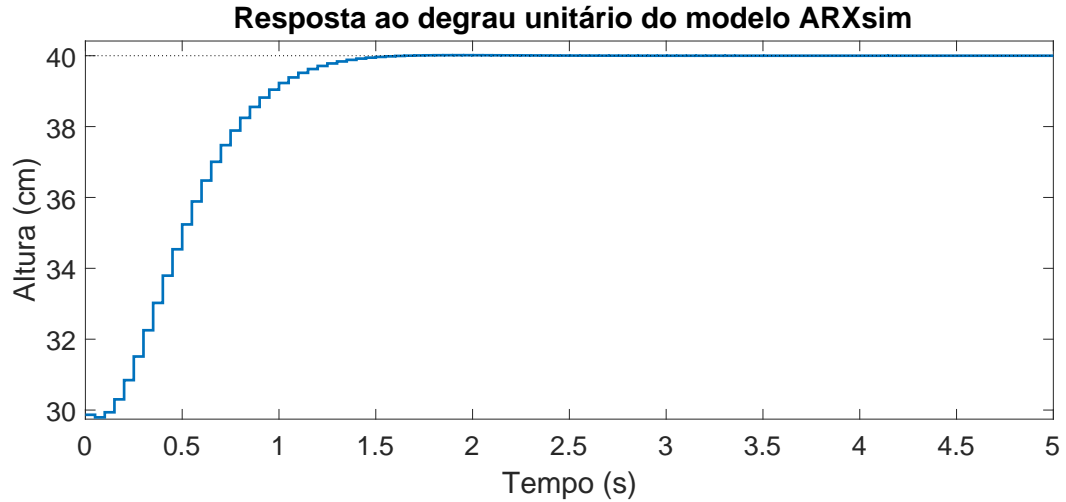


Figura 5.4: Resposta ao degrau do modelo simulado *ARXsim* controlado

Vemos na figura 5.4 que a resposta ao degrau unitário do sistema controlado tem tempo de assentamento de 1.3 segundos e máximo sobrevalor de 0.1487%.

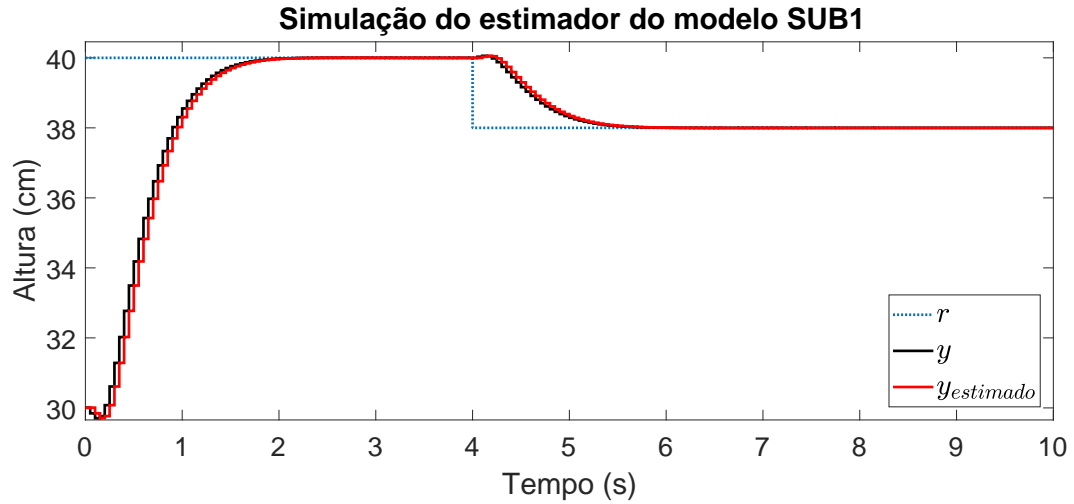
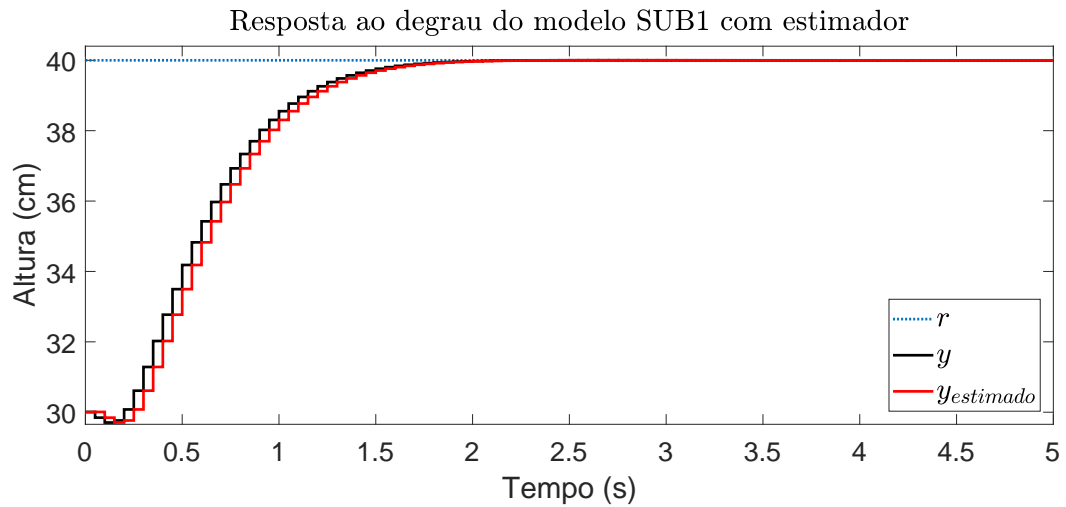
## 5.2 Projeto do Estimador de Estados

Para realizar um controlador por realimentação de estados precisamos ser capazes de medir todos os estados do nosso modelo a cada tempo de amostragem. O nosso sistema, no entanto, não tem sensores para medir cada um dos estados necessários, a única medida disponível é a saída do sistema na forma da altura da bola. Portanto, precisamos implementar um estimador de estados para cada um dos controladores projetados na subseção anterior.

### 5.2.1 Modelo *SUB1*

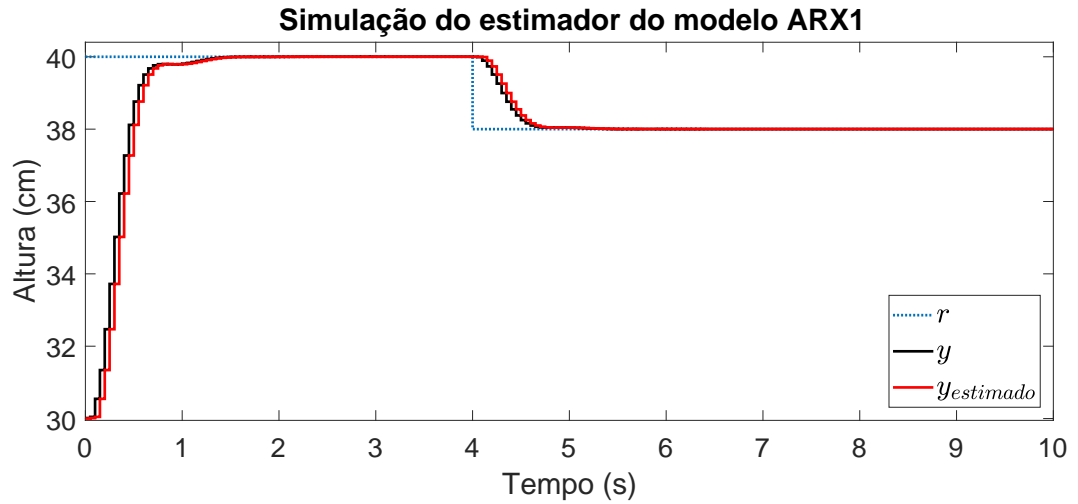
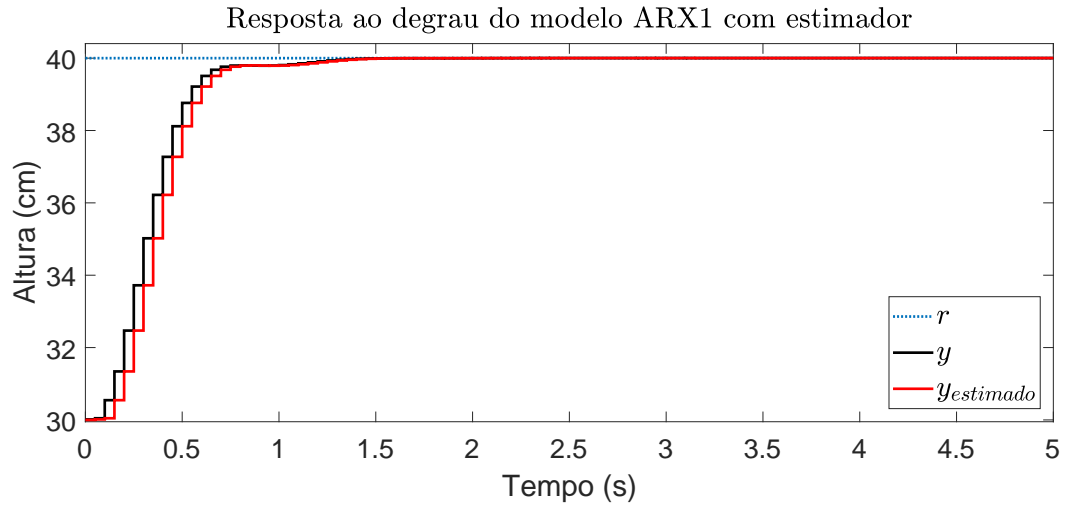
Um estimador de estados é projetado de forma similar ao controlador por realimentação de estados, precisamos escolher polos adequados para que o estimador funcione da forma correta. A forma mais simples de escolher esses polos é fazer com que eles reajam mais rápido que o sistema à entrada recebida, conseguimos isso multiplicando a parte real dos polos no domínio  $s$  por um número para que sejam mais rápidos.

Projetamos o estimador de estados com os seguintes polos  $z_1 = 0.1647 + 0.0144i$ ,  $z_2 = 0.1647 - 0.0144i$  e  $z_3 = 0.0183$  e obtemos o ganho do estimador  $L = [3.5436, -7.2192, 13.9077]^T$ . Nas figuras 5.5 e 5.6 vemos como uma simulação do sistema se comporta.

Figura 5.5: Simulação do estimador do modelo *SUB1*Figura 5.6: Resposta ao degrau do modelo *SUB1* controlado com observador

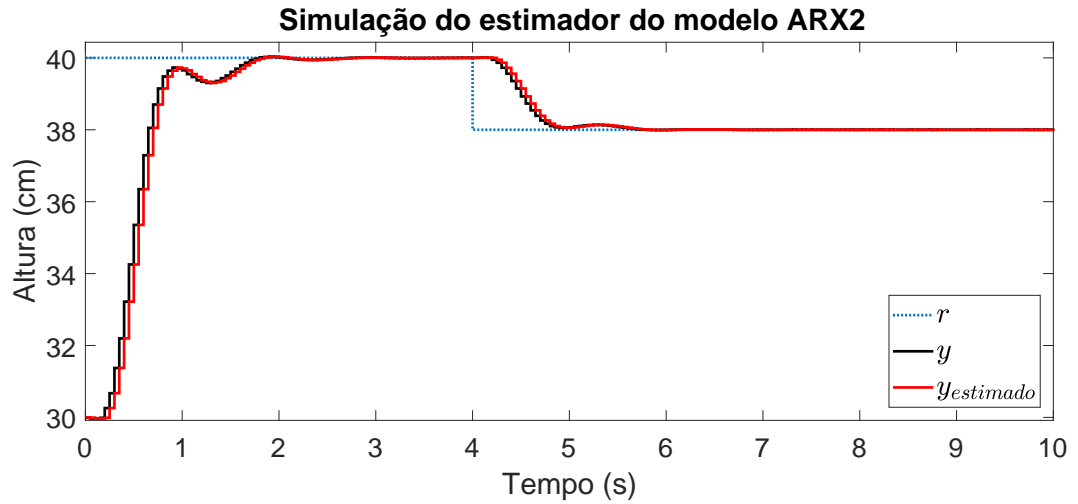
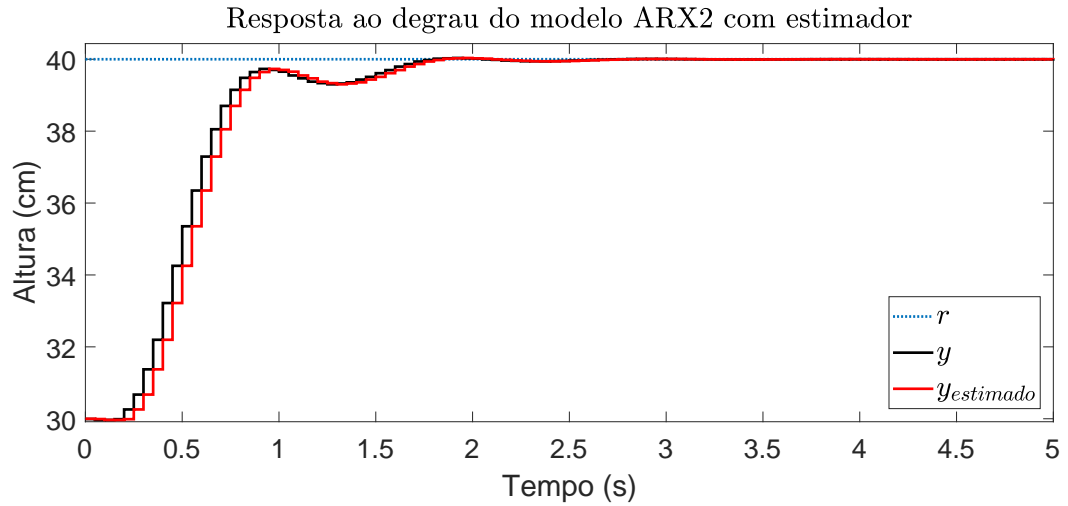
### 5.2.2 Modelo *ARX1*

Para o estimador do modelo *ARX1* alocamos os polos em  $z_1 = 0.0309 + 0.0048i$ ,  $z_2 = 0.0309 - 0.0048i$  e  $z_3 = 0.0003$ . Obtemos o ganho do estimador  $L = [33.9534, 26.6709, 16.6024]^T$ . Nas figuras 5.7 e 5.8 vemos como uma simulação do sistema se comporta.

Figura 5.7: Simulação do estimador do modelo  $ARX1$ Figura 5.8: Resposta ao degrau do modelo  $ARX1$  controlado com observador

### 5.2.3 Modelo $ARX2$

Para o estimador do modelo  $ARX2$  alocamos os polos em  $z_1 = 0.0304 + 0.0074i$ ,  $z_2 = 0.0304 - 0.0074i$ ,  $z_3 = 0.0074$ ,  $z_4 = 0.0072$  e  $z_5 = 0.0069$ . Obtemos o ganho do estimador  $L = [22.1925, 20.4118, 16.9038, 16.9671, 11.1584]^T$ . Nas figuras 5.9 e 5.10 vemos como uma simulação do sistema se comporta.

Figura 5.9: Simulação do estimador do modelo  $ARX2$ Figura 5.10: Resposta ao degrau do modelo  $ARX2$  controlado com observador

#### 5.2.4 Modelo $ARXsim$

Para o estimador do modelo  $ARX2$  alocamos os polos em  $z_1 = 0.3383 + 0.0296i$ ,  $z_2 = 0.3383 - 0.0296i$  e  $z_3 = 0.0150$ . Obtemos o ganho do estimador  $L = [27.5247, 73.1490, 146.4144]^T$ . Nas figuras 5.11 e 5.12 vemos como uma simulação do sistema se comporta.

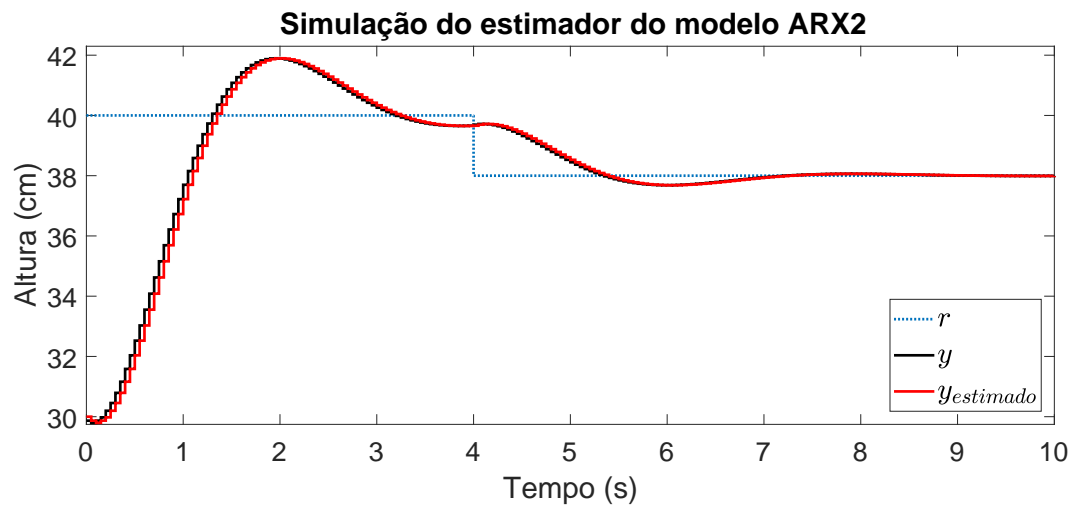


Figura 5.11: Simulação do estimador do modelo *ARXsim*

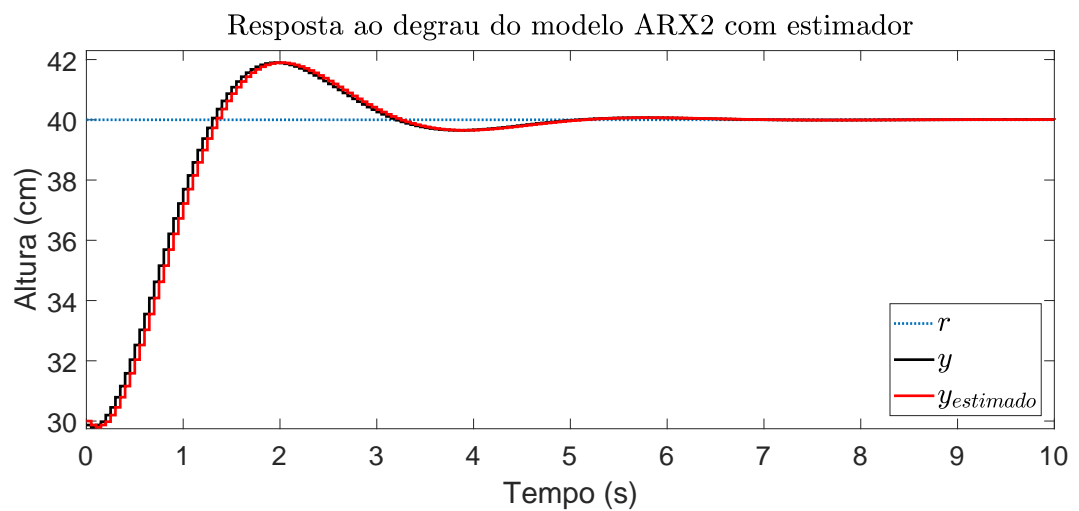


Figura 5.12: Resposta ao degrau do modelo *ARXsim* controlado com observador

## Capítulo 6

# Avaliação Experimental do Sistema de Controle

Após projetar os controladores faremos uma avaliação do seu funcionamento aplicando cada um no sistema real.

### 6.1 Descrição e objetivos experimentais

Os experimentos descritos neste capítulo tem como objetivo verificar e avaliar o funcionamento do sistema ao ser controlado pelos controladores projetados no capítulo 5. Serão executados 4 experimentos no sistema que funcionam da seguinte forma:

**Resposta ao Degrau Unitário** A resposta ao degrau unitário é um experimento onde o sistema recebe uma referência unitária e reage à ela. Com este experimento iremos verificar se o sistema se mantém estável ao ser controlado e se ele atende aos requisitos definidos quando o controlador foi projetado.

**Resposta à escadaria** A resposta à escadaria é um experimento onde o sistema recebe alguns níveis de referência ao longo do tempo e medimos sua resposta para ver como reage à mudança de referência.

**Teste de Robustez à Mudança de Parâmetros** O teste de robustez à mudança de parâmetros serve para determinar se o controlador é capaz de reagir quando algum

parâmetro é alterado. Neste experimento alteramos o peso da bola e verificamos se o sistema mantém a referência.

**Teste de Robustez à Perturbação Externa** O teste de robustez à perturbação externa mostra se o sistema controlado é capaz de rejeitar perturbações externas ao sistema. Para executar este teste iremos medir a resposta do sistema enquanto obstruímos a saída de ar através dos buracos inferiores do duto do túnel de vento, efetivamente aumentando o fluxo de ar que eleva a bola.

## 6.2 Resultados Experimentais

Nesta seção serão apresentados os resultados dos experimentos descritos anteriormente.

### 6.2.1 Resultados da Resposta ao degrau unitário

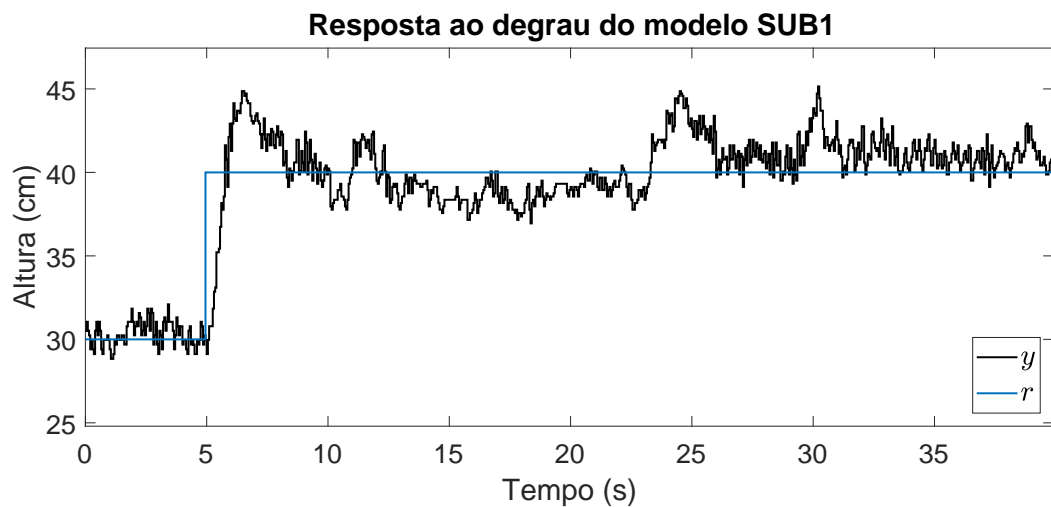
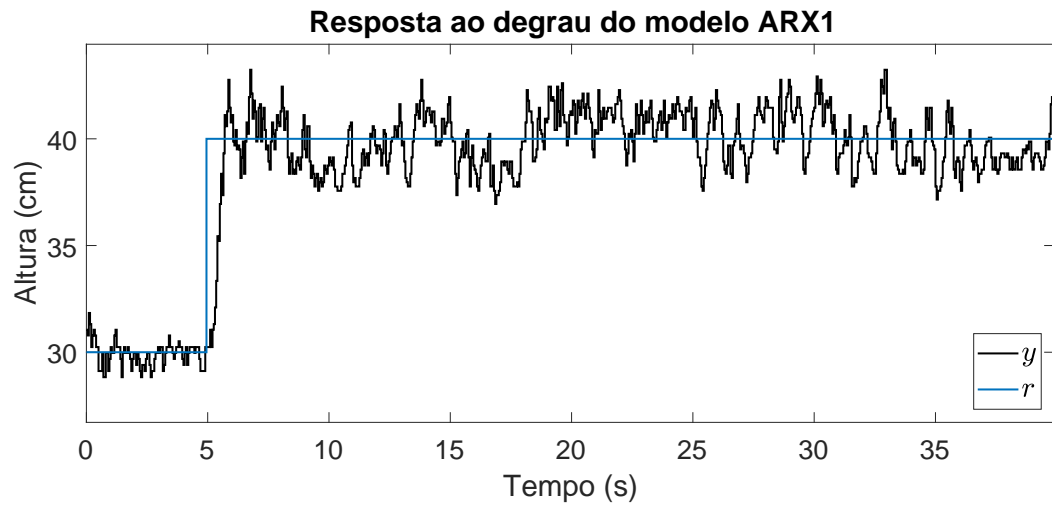
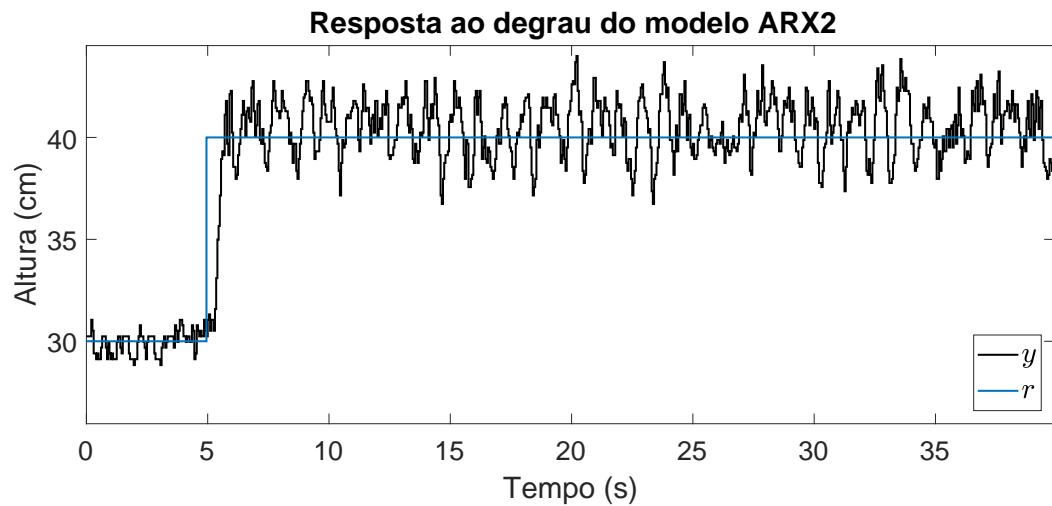


Figura 6.1: Resposta ao degrau do sistema usando o controlador *SUB1*

Figura 6.2: Resposta ao degrau do sistema usando o controlador  $ARX1$ Figura 6.3: Resposta ao degrau do sistema usando o controlador  $ARX2$



### 6.2.2 Resultados da Resposta à escadaria

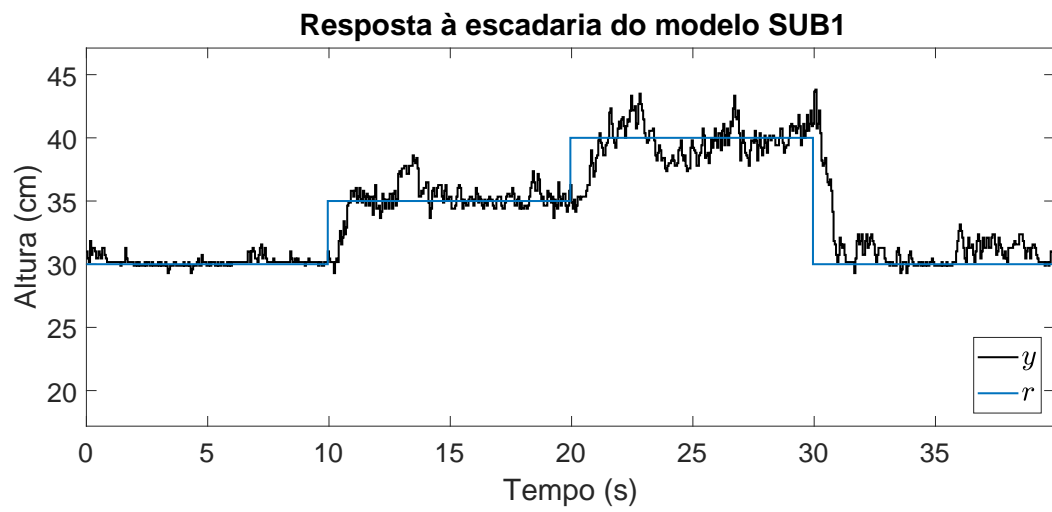


Figura 6.4: Resposta à escadaria do sistema usando o controlador *SUB1*

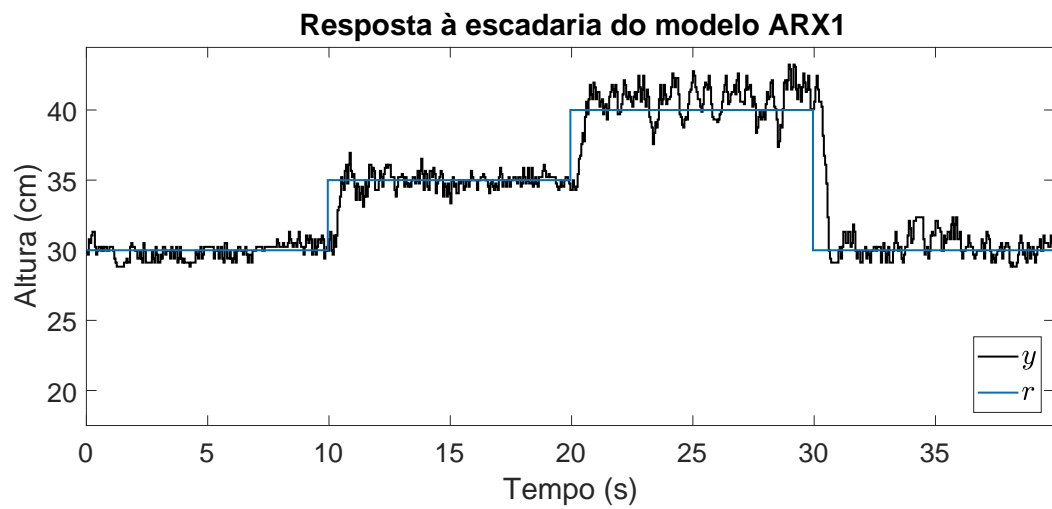


Figura 6.5: Resposta à escadaria do sistema usando o controlador *ARX1*

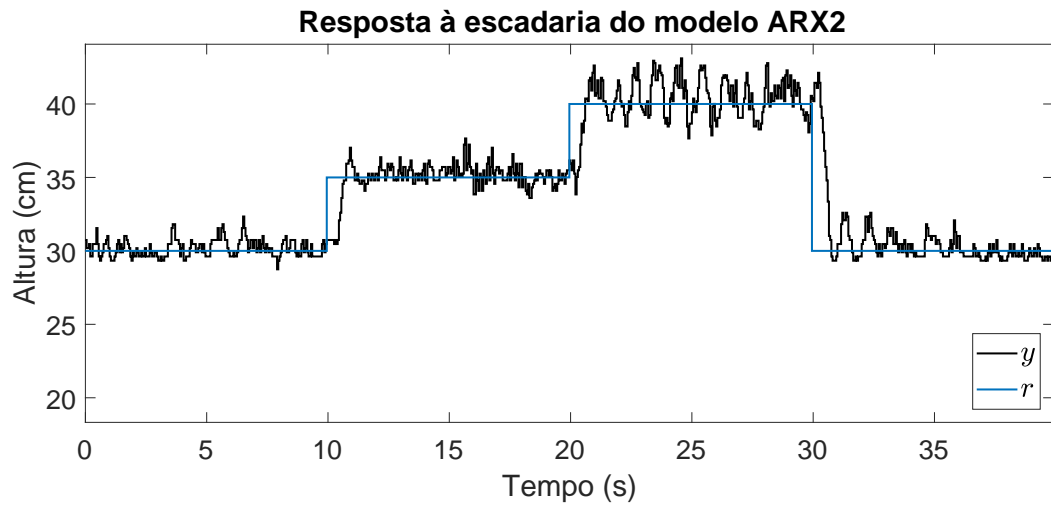


Figura 6.6: Resposta à escadaria do sistema usando o controlador  $ARX2$

### 6.2.3 Resultados do Teste de Robustez à mudança de Parâmetros

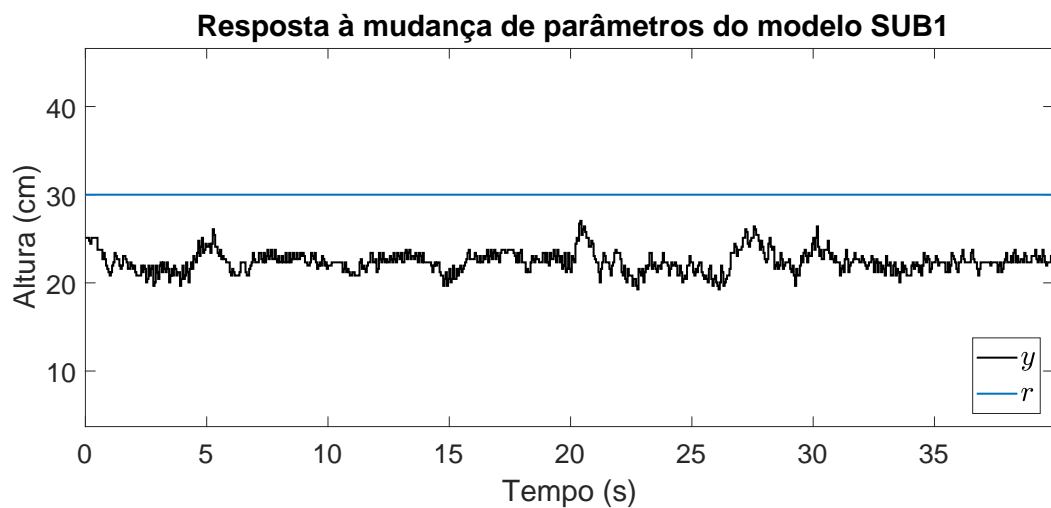


Figura 6.7: Resposta à mudança de parâmetros do sistema com o controlador do modelo  $SUB1$

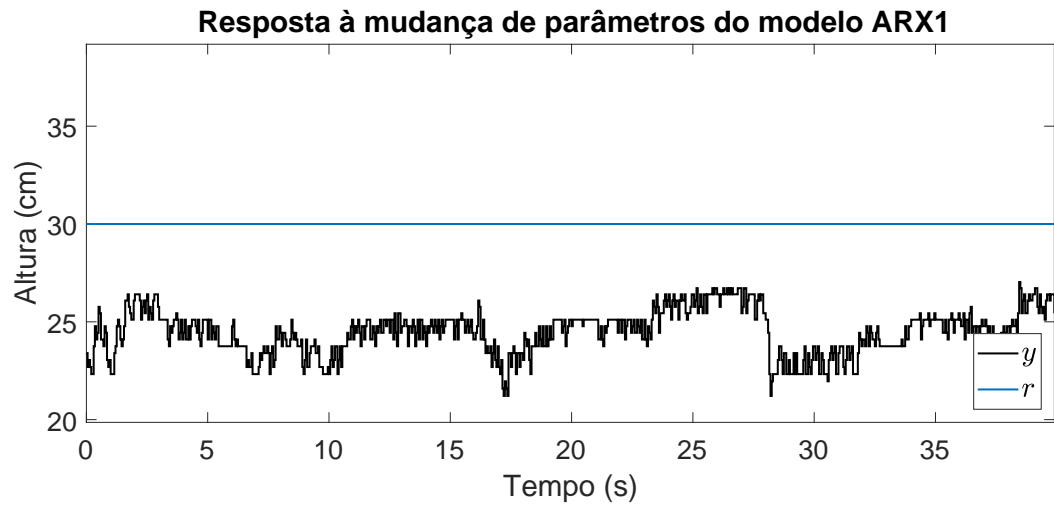


Figura 6.8: Resposta à mudança de parâmetros do sistema com o controlador do modelo *ARX1*

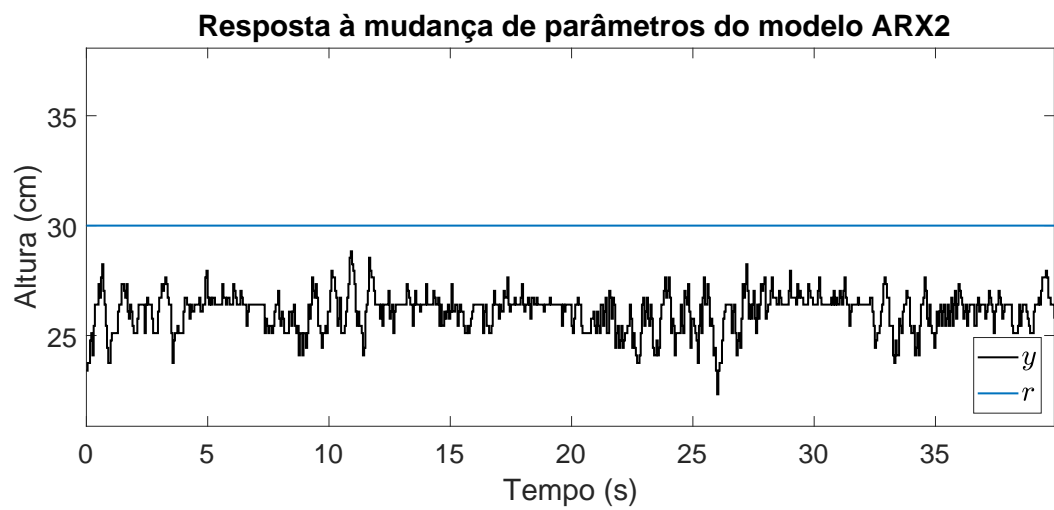


Figura 6.9: Resposta à mudança de parâmetros do sistema com o controlador do modelo *ARX2*

### 6.2.4 Resultados do Teste de Robustez à perturbação externa

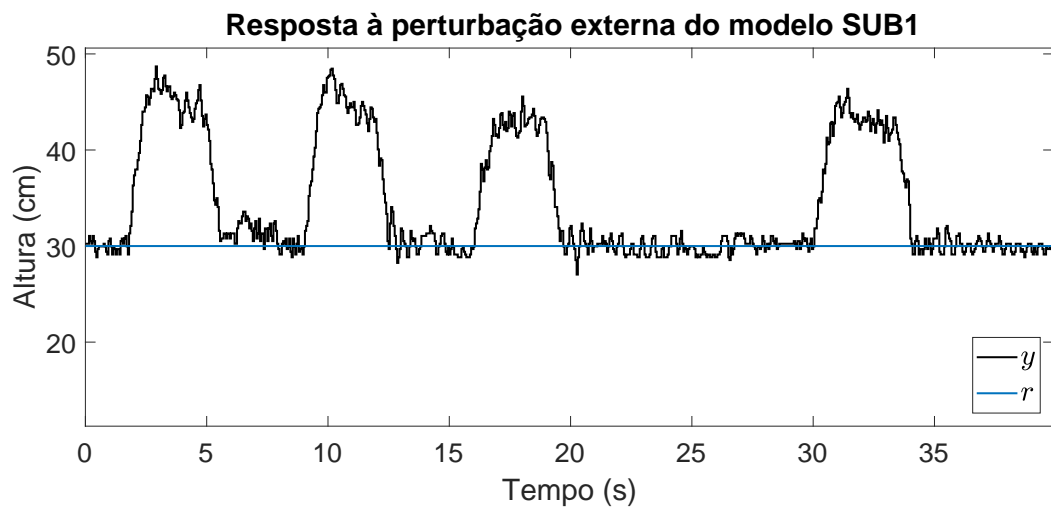


Figura 6.10: Resposta à perturbação externa do sistema com o controlador do modelo *SUB1*

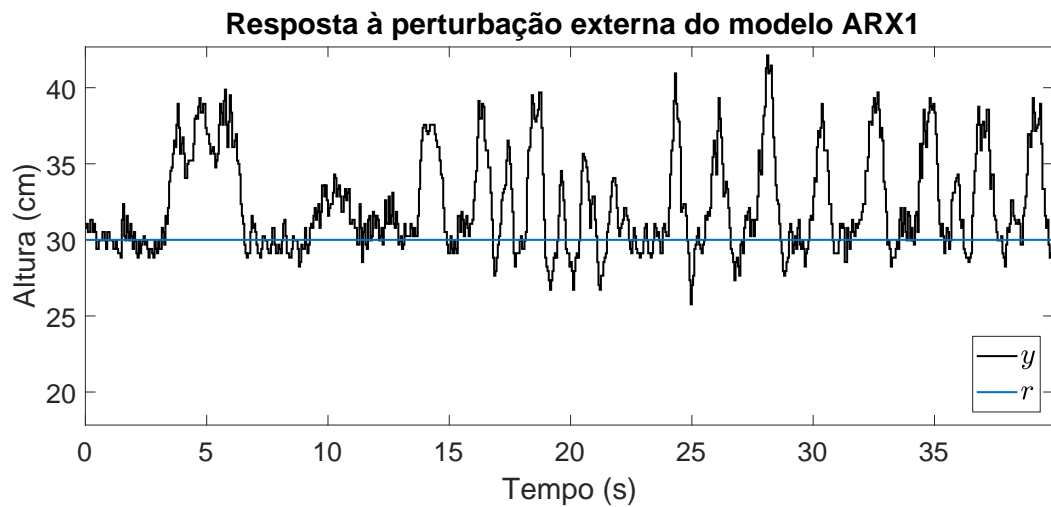


Figura 6.11: Resposta à perturbação externa do sistema com o controlador do modelo *ARX1*

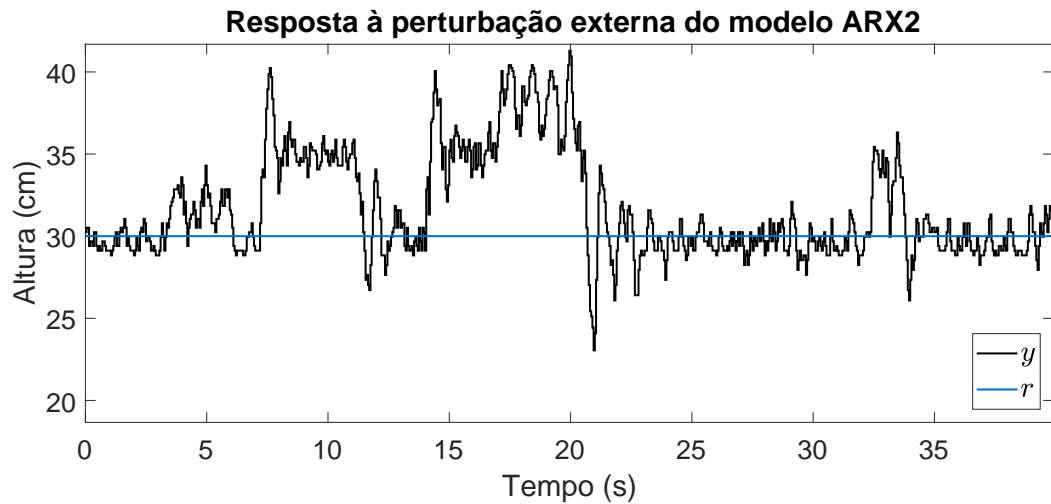


Figura 6.12: Resposta à perturbação externa do sistema com o controlador do modelo  $ARX2$

### 6.3 Discussão

Vemos nas figuras da seção 6.2.1 que os modelos  $SUB1$ ,  $ARX1$  e  $ARX2$  atendem ao requisito de tempo de assentamento mas não atendem ao requisito de máximo sobrevalor, apesar de que eles não cheguem a estabilizar completamente em um ponto. Isso acontece devido uma série de fatores possíveis, ruído no sensor, reflexo dos raios infra vermelhos na parede do túnel de vento, atuação falha no motor.

Na seção 6.2.2 vemos que com o controlador projetado é capaz de fazer o sistema seguir os diferentes níveis de tensão e estabilizar. Para os modelos  $ARX1$  e  $ARX2$  o sistema se mostra estabilizado enquanto a referência está em 35 cm.

Na seção 6.2.3 fica claro que nenhum dos controladores projetados tem capacidade para reagir à mudança de parâmetros.

Na seção 6.2.4 vemos que nenhum dos controladores rejeita perturbações externas.

# Capítulo 7

## Conclusão

### 7.1 Considerações Finais

Durante a execução deste trabalho foi construída uma plataforma de experimentos de túnel de vento vertical, atuada por uma turbina de avião RC onde a altura de uma bola é controlado pela força do vento. Foram aplicados métodos de identificação caixa preta para extrair 3 modelos diferentes do sistema e um método de identificação caixa cinza.

### 7.2 Trabalhos Futuros

Refazer o código de atuação do motor no sistema

# Referências Bibliográficas

- [1] Aguirre, L.  
2015. Introdução à Identificação de Sistemas – Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais. Editora UFMG.
- [2] Akaike, H.  
1974. A new look at the statistical model identification. IEEE Transactions on Automatic Control, 19(6):716–723.
- [3] bin Kwon, H., Y.-W. Park, D. ho Lee, and M.-S. Kim  
2001. Wind tunnel experiments on korean high-speed trains using various ground simulation techniques. Journal of Wind Engineering and Industrial Aerodynamics, 89(13):1179 – 1195.
- [4] Briggs, L. J.  
1959. Effect of spin and speed on the lateral deflection (curve) of a baseball; and the magnus effect for smooth spheres. American Journal of Physics, 27:589–596.
- [5] Chacon J, Saenz J, d. l. T. L. D. J. E. F.  
2017. Design of a low-cost air levitation system for teaching control engineering. Sensors (Basel).
- [6] Jernigan, S. R., Y. Fahmy, and G. D. Buckner  
2009. Implementing a remote laboratory experience into a joint engineering degree program: Aerodynamic levitation of a beach ball. IEEE Transactions on Education, 52(2):205–213.
- [7] journal, E. E.  
1946. Vertical wind tunnel. Electrical Engineering, 65(6):265–266.

- 
- [8] Katayama, T.  
2005. Subspace Methods for System Identification, Communications and Control Engineering. Springer London.
- [9] McDade, J., G. Phillips, H. Sivinski, and W. Whitfield  
1969. Chapter v principles and applications of laminar-flow devices. In Methods in Microbiology, J. Norris and D. Ribbons, eds., volume 1 of Methods in Microbiology, Pp. 137 – 168. Academic Press.
- [10] Ricco, R.  
2012. Identificação de sistemas utilizando métodos de subespaços.