

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
CURSO DE ENGENHARIA DE ELETRICIDADE

DANIEL TROVÃO SIMÕES

IDENTIFICAÇÃO DE SISTEMA DE VEÍCULO AÉREO NÃO
TRIPULADO (VANT) TIPO QUADRIMOTOR.

São Luís – MA

2017

DANIEL TROVÃO SIMÕES

IDENTIFICAÇÃO DE SISTEMA DE VEÍCULO AÉREO NÃO
TRIPULADO (VANT) TIPO QUADRIMOTOR.

Monografia apresentada ao Curso de
Engenharia de Eletricidade da UFMA,
como requisito para a obtenção do grau
de BACHAREL em Engenharia de
Eletricidade.

João Viana da Fonseca Neto, Dr.
(Orientador)

São Luís - MA
2017

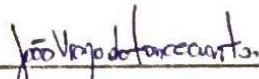
DANIEL TROVÃO SIMÕES

**IDENTIFICAÇÃO DE SISTEMA DE VEÍCULO AÉREO NÃO
TRIPULADO (VANT) TIPO QUADRIMOTOR**

Monografia apresentada ao Curso de Engenharia de Eletricidade da UFMA, como requisito para a obtenção do grau de BACHAREL em Engenharia de Eletricidade.

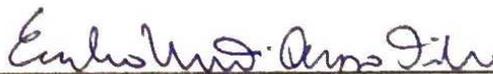
Aprovada em 25 de julho de 2017

BANCA EXAMINADORA



João Viana da Fonseca Neto, Dr.

(Orientador)



Evandro Martins Araújo Filho, Eng.

(MEMBRO)



Pedro Baptista Fernandes, Msc.

(MEMBRO)

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

Trovão Simões, Daniel.

IDENTIFICAÇÃO DE SISTEMA DE VEÍCULO AÉREO NÃO TRIPULADO
VANT TIPO QUADRIMOTOR / Daniel Trovão Simões. - 2017.
72 p.

Orientador(a): João Viana da Fonseca Neto.

Monografia (Graduação) - Curso de Engenharia Elétrica,
Universidade Federal do Maranhão, São Luís, 2017.

1. Identificação de Sistemas. 2. Microcontrolador
Arduino. 3. Quadricóptero. 4. Veículo aéreo não
tripulado (VANT). I. da Fonseca Neto, João Viana. II.
Título.

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus pela força de todos os dias para concluir a minha graduação para ser um Engenheiro Eletricista. Em segundo lugar, a minha mãe por ter me ajudado e servido de porto seguro todos esses anos.

Ao meu professor e orientador Prof. Dr. João Viana da Fonseca Neto pelo suporte financeiro e por ter me dado à oportunidade de trabalhar com pessoas incríveis no Laboratório de Sistemas Embarcados e Controle Inteligente (LABSECI).

Agradeço imensamente ao meu amigo Evandro pela grande ajuda e apoio nesta pesquisa, sem seu esforço e auxílio com certeza este trabalho não seria concluído. Pessoas com capacidade de síntese, análise e aprendizado constante estão cada vez mais raras nos tempos atuais.

À minha namorada, Ana Paula, por estar nesses anos finais de minha graduação me dando apoio incondicional nos momentos difíceis.

Aos amigos de graduação, Deocarolo, Guilherme, Igor, Matheus, Juarez, Patrícia, Elis, Marcos e Yan que estão desde o começo do curso e sempre presente nos momentos felizes e difíceis que passamos e levarei vocês para sempre no meu coração. Também agradeço aos meus colegas de laboratório Carol, Bruno, Welson e Prof. Ernesto.

Aos amigos de infância, Pedro, Gabriel, Alisson, Breno e Arthur como também aos amigos que fizeram parte da minha história, mas agora estão longe. Todos os momentos de risada e as boas histórias me fizeram caminhar e seguir em direção ao fim dessa jornada de forma agradável.

A todas essas pessoas o meu profundo agradecimento e o orgulho que sinto em olhar para trás e ver as boas amizades que cultivei até agora.

RESUMO

Esta monografia realizou um projeto sobre veículos aéreos não tripulados (VANTs) com enfoque no aeromodelo quadricóptero como também desenvolveu uma placa de aquisição de dados com o microcontrolador Arduino, que foi acoplada ao quadrimotor comercial SYMA X6, colheu os dados necessários para identificação de um sistema (SISO). O modelo matemático utilizado foi o Auto-Regressivo com Entradas Exógenas (ARX) e seus coeficientes foram calculados pelo software MATLAB. A ideia principal foi obter um modelo matemático confiável do quadricóptero.

Palavras-chave: Veículo aéreo não tripulado (VANT). Quadricóptero. Microcontrolador Arduino. Identificação de Sistemas.

ABSTRACT

This work proposes a study about Unmanned Aerial Vehicles (UAV) focusing in an aircraft, the quadcopter. Besides, this thesis intends to assembly a data acquisition system (DAQ) with the microcontroller Arduino. The DAQ will be attached on the quadcopter SYMA X6 that it will acquire the necessary data for identification of a SISO system. The mathematical model used was the Auto-Regressive with eXogenous inputs (ARX) and its coefficients were calculated by the MATLAB. The main idea was to obtain a reliable mathematical model of quadcopter.

Keywords: Unmanned Aerial Vehicles (UAV). Quadcopter. Microcontroller Arduino. System Identification.

LISTA DE FIGURAS.

Figura 1.1- Estrutura do Trabalho.	4
Figura 2.1- Helicóptero não tripulado QH-50 DASH.	5
Figura 2.2- Classificação de Aeromodelos.	6
Figura 2.3- Bréguet-Richet Gyroplane 1.	7
Figura 2.4- Quadricóptero Moderno.	8
Figura 2.5- Estrutura Simplificada do Quadricóptero.	9
Figura 2.6- Comando Empuxo Quadricóptero.	10
Figura 2.7- Comando Rolagem Quadricóptero.	10
Figura 2.8- Comando de Arfagem Quadricóptero.	11
Figura 2.9- Comando de Guinada do Quadricóptero.	12
Figura 2.10- Referencial Inercial e de Corpo de um quadricóptero.	12
Figura 2.11- Sistema de aquisição de dados e seus elementos.	16
Figura 2.12- Fluxograma de uma identificação de sistema.	22
Figura 3.1- Quadricóptero Syma X6.	30
Figura 3.2- Placa de Aquisição (NI-myDAQ).	32
Figura 3.3 - Microcontrolador Arduino Uno.	33
Figura 3.4 - Placa de aquisição ACBOARD.	34
Figura 3.5 - MPU6050.	34
Figura 3.6 - Placa de comunicação sem fio APC 220.	35
Figura 3.7 - Leitora de Cartão SD.	36
Figura 3.8 - SONAR HF-SRF04.	36
Figura 3.9 - ACBOARD acoplada à placa física.	37
Figura 3.10 - Construção da ACBOARD de dados na protoboard.	38
Figura 3. 11: Esquema de prototipagem da placa de aquisição de dados.	38
Figura 3.12 - ACBOARD com Syma X6.	39
Figura 3.13 - Fluxograma do algoritmo de aquisição da ACBOARD.	39
Figura 3.14 - Tela do programa Hercules.	40
Figura 3.15 - Processo de aquisição e transmissão de dados ACBOARD.	40
Figura 3.16 - Diagrama do processo do procedimento experimental.	41
Figura 4.1 - Dados coletados para identificação.	45
Figura 4.2 - Dados coletados para validação.	46
Figura 4.3 - Dados de identificação filtrados.	47

Figura 4.4 - Dados de validação filtrados.....	47
Figura 4.5 - Convergência dos Parâmetros Identificados.....	49
Figura 4.6 - Erro de predição do estimador.	50
Figura 4.7 - Identificação do SYMA X6.	51
Figura 4.8 – Validação do Sistema SYMA X6.	52
Figura 4.9 - Validação de dados com a ferramenta IDENTIFICATION.	53
Figura 4.10 - Validação de dados com Algoritmo ARX.	53
Figura 4.11 - Gráfico de polos e zeros no círculo unitário.	54
Figura 4.12 - Gráfico de polos e zeros no círculo unitário Ampliado.	54
Figura 4.13 - Análise de $G(z)$ e $S(z)$ ao impulso.	55
Figura 4.14 - Análise de $G(z)$ e $S(z)$ ao degrau.	56

LISTA DE TABELAS

Tabela 4.1 - Valores pré-configurados do filtro de Kalman.....	46
Tabela 4.2 - Valores de Entrada para o Algoritmo ARX.	48

LISTAS DE SIGLAS E ABREVIATURAS.

VANT	VEÍCULO AÉREO NÃO TRIPULADO.
ACBOARD	ACQUISITION BOARD.
ANEEL	ASSOCIAÇÃO NACIONAL DE ENERGIA ELÉTRICA.
A/D	ANALÓGICO/ DIGITAL.
RTD	RESISTENCE TEMPERATURE DETECTOR.
DAC	DIGITAL ANALOG CONVERTER
VTOL	VERTICAL TAKE OFF LANDING.
USB	UNIVERSAL SERIAL BUS.
DAQ	DATA ACQUISITION.
DAS	DATA ACQUISITION SYSTEM.
ARX	AUTO REGRESSIVE EXOGENOUS INPUTS.
BLUE	BEST LINEAR UNIVERSAL ESTIMATOR
SISO	SINGLE INPUT SINGLE OUTPUT.
MIMO	MULTIPLE INPUTS MULTIPLE OUTPUTS.
FPE	FINAL PREDICTION ERROR.
MATLAB	MATRICIAL LAB.
IDE	INTEGRATED DEVELOPMENT ENVIRONMENTAL.
Bps	BITS PER SECONDS.
CHESF	COMPANHIA HIDRELÉTRICA DO SÃO FRANCISCO.

SUMÁRIO

1 INTRODUÇÃO.....	1
1.1 Objetivos	2
1.1.1 Objetivo geral.....	2
1.1.2 Objetivos específicos	2
1.2 Motivação.....	2
1.3 Estrutura do Trabalho.....	3
2 TEORIA E ESTADO DA ARTE DA IDENTIFICAÇÃO DE SISTEMA E QUADRIMOTORES.	5
2.1 Veículos Aéreos Não Tripulados (VANT).....	5
2.2 Quadricóptero.....	6
2.2.1 Histórico.....	7
2.2.2 Conceitos Básicos	8
2.2.3 Modelo Newton-Euler.....	12
2.3 Sistema de Aquisição de Dados.	16
2.3.1 Elementos de um Sistema de Aquisição.	17
2.3.2 Características dos Sistemas de Aquisição.	20
2.4 Identificação de Sistemas.	20
2.4.1 Etapas para Identificação.	21
2.4.2 Projeto Experimental.....	22
2.4.3 Coleta de Dados.	23
2.4.4 O MODELO ARX.	23
2.4.5 Estimação de Parâmetros.	24
2.4.6 Validação de Dados.....	27
3 PROJETO DA PLACA DA ACBOARD E PROCEDIMENTOS PARA A IDENTIFICAÇÃO DE SISTEMA DO QUADRICÓPTERO.	30
3.1 Materiais Para Identificação do Sistema SYMA X6.....	30
3.1.1 Quadricóptero Syma X6.....	30

3.1.2	MATLAB.....	31
3.1.3	Sistema de Aquisição de Dados (Ni-mydaq).....	32
3.1.4	O Microcontrolador Arduino.....	33
3.1.5	Placa de Aquisição de Dados.....	33
3.2	Métodos Para Identificação do Sistema SYMA X6.....	37
3.2.1	Desenvolvimento da Placa de Aquisição de Dados ACBOARD.....	37
3.2.2	O Algoritmo e Processo de Aquisição de Dados da ACBOARD.....	39
3.2.3	O Projeto do Experimento Utilizado para Identificação do Sistema.....	41
3.2.4	Análise e Tratamento dos Dados Coletados.....	42
3.2.5	Filtragem dos Dados.....	43
3.2.6	O Algoritmo de Identificação de Sistema.....	43
3.2.7	Validação do Sistema.....	44
4	EXPERIMENTOS E RESULTADOS.....	45
4.1	Dados de Entrada e Saída Coletados e Filtrados.....	45
4.2	Algoritmo ARX.....	48
4.3	Validação do SYMA X6.....	51
4.4	Comparação com a Ferramenta “Identification” do MATLAB®.....	52
5	CONCLUSÃO.....	57
	REFERÊNCIAS BIBLIOGRÁFICAS.....	59
	ANEXOS.....	62

1 INTRODUÇÃO

No cenário atual, uma realidade cada vez mais presente é a utilização de dispositivos tecnológicos como os **smartphones**, **tablets** e **laptops** que entraram de forma gradual no cotidiano humano. Outro dispositivo que está começando a conquistar um espaço na atualidade são os VANTs (Veículos Aéreos Não Tripulados). VANTs são veículos com capacidade de realizar voos que foram construídos ou modificados para não comportar um piloto humano e são operados por controle remoto ou autônomo (DE CASTRO JORGE e INAMASU, 2014).

No Brasil, os VANTs estão sendo utilizados com ênfase nas áreas agrícolas com fotografias de solo, análise hídrica, identificação de pragas e deterioração de plantios. Um exemplo de sua aplicação em empresas pode ser visto na Companhia Hidrelétrica do São Francisco (CHESF), que utiliza VANTs para gerenciar o circuito externo em cumprimento das normas da Agência Nacional de Energia Elétrica (ANEEL) de cabos e postes. A Petrobrás também começou a investir nessa modalidade para monitoria de vazamentos em dutos. Além disso, o território nacional necessita de constante patrulhamento nas fronteiras, algo que é impossível devido à dificuldade de acesso em certas regiões do país, o que torna os VANTs uma solução atrativa para o governo nacional (FURTADO, VEIGA GIMENES e CAMARGO JUNIOR, 2008).

Há vários modelos enquadrados na categoria de VANTs como aviões, helicópteros e dirigíveis, porém um tem alcançado grande interesse comercial, o quadricóptero que é um veículo aéreo com quatro motores dispostos em cruz, dada a sua simplicidade de construção e o desafiante sistema de controle (DE CASTRO RIBEIRO RESENDE, 2015). Este será o objeto de estudo desta monografia.

Esta monografia tem como objetivo o estudo do VANT modelo quadricóptero comercial SYMA X6, seguido da construção de uma placa de aquisição de dados com o intuito de coletar dados de funcionamento para serem utilizados no processo de identificação do sistema obtendo um modelo matemático do mesmo.

1.1 Objetivos

1.1.1 Objetivo geral

Desenvolver um procedimento para identificação e uma placa de aquisição de dados no intuito de obter o modelo matemático do quadrimotor.

1.1.2 Objetivos específicos

- Estudar a dinâmica dos quadrimotores, comportamentos e controle;
- Pesquisa de artigos e trabalhos na área em estudo;
- Construir uma placa de aquisição de dados **Aquisition board (ACBOARD)** utilizando o microcontrolador Arduino;
- Implementar algoritmos em linguagem **Wring** na coleta de dados para identificação do modelo quadrimotor;
- Estudar técnicas de identificação de sistemas e escolher a técnica mais apropriada;
- Realizar a identificação do modelo matemático da planta quadrimotor utilizando o software **MATLAB®**.

1.2 Motivação

Os trabalhos desenvolvidos com o mesmo propósito desta monografia, a identificação da dinâmica de um quadrimotor, mostram que são recentes e ainda estão em desenvolvimento.

Em 2011, STANCULEANU propõe um trabalho identificando o modelo da dinâmica de um quadricóptero usando uma técnica do tipo caixa-preta. O diferencial deste trabalho é a realização da identificação do VANT funcionando em malha fechada utilizando o método de predição de erro. STANCULEANU modela as variáveis angulares que são: Arfagem, Rolagem e Guinada. Elas serão explicadas durante os próximos capítulos. (STANCULEANU e BORANGIU, 2011)

Em 2014, OLIVEIRA também desenvolve um experimento de identificação e validação de um quadrimotor em modelagem caixa-preta utilizando a **toolbox** de identificação do MATLAB. Ele utilizou modelos de processos e posteriormente aplicou processos

gaussianos. Desta forma ele cria dois modelos diferentes para a dinâmica (OLIVEIRA MARTINS, 2014)

Em 2015, SALAMEH e AMMAR propõe um trabalho de identificação de um quadricóptero utilizando aquisição de dados. O grande diferencial desta pesquisa é, vendo a complexidade da criação de um modelo matemático SALAMEH e AMMAR realizam vários voos com o veículo e coletam dados para criar um modelo ARX (Auto-Regressivo com entradas Exógenas). Este tipo de modelo é o mesmo proposto nesta monografia. (SALAMEH, AMMAR e TUTUNJI, 2015)

Também em 2015, KURAL desenvolve um trabalho identificando com um equipamento especial de telemetria coletando dados e criando um modelo que relaciona os ângulos de Euler, que será explicado posteriormente nesta monografia, com as velocidades dos motores através também de um modelo ARX. (SARıOğLU e KURAL, 2015)

Recentemente, VICTOR MANUEL e EDGAR RIVAS aplicaram outro tipo de identificação utilizando a ferramenta MATLAB com modelo de função de transferência em Laplace para os ângulos de Rolagem e Arfagem no ano de 2016. (RESENDIZ e ARAIZA, 2016)

No ano de 2017, este trabalho de conclusão de curso se propõe a realizar uma identificação da dinâmica do quadricóptero SYMA X6 desde a criação de uma placa de aquisição de dados até a identificação. A variável de entrada foi à tensão do manche de subida e descida e a saída à altura do quadrimotor medida por um sonar.

1.3 Estrutura do Trabalho

Nesta monografia é apresentado um projeto de construção de uma placa de aquisição de dados (ACBOARD) com a tecnologia Arduino para identificar um sistema de um quadrimotor. Como também, os procedimentos necessários para uma identificação de sistema.

Figura 1.1- Estrutura do Trabalho.



Fonte: Daniel Trovão.

2 TEORIA E ESTADO DA ARTE DA IDENTIFICAÇÃO DE SISTEMA E QUADRIMOTORES

Neste capítulo apresenta-se a teoria de quadricópteros e identificação de sistemas nos tópicos: Veículos Aéreos Não Tripulados; Quadricóptero, que se dividem em Histórico, Conceitos Básicos e Modelo Newton Euler; Sistemas de Aquisição de dados, com subtópicos elementos e características de um sistema de aquisição e por fim o tópico, Identificação de Sistemas.

2.1 Veículos Aéreos Não Tripulados (VANT)

A terminologia VANTs, abreviação de Veículos Aéreos Não Tripulados vem do termo em inglês UAV (Unmanned Aircraft Vehicle) que se define por um veículo que não apresenta nenhuma presença humana em sua estrutura, tanto piloto quanto passageiro. Assim, pela ausência de presença humana, o controle da aeronave deve ser feita por operação humana de forma remota ou utilizando computadores de bordo. (VALAVANIS e VACHTSEVANOS, 2014).

Os primeiros relatos de aviação e tentativa de voo não tripulada provêm do século XIX com modelos criados e seguidos pelos inventores Cayley, Stringfellow e Du Temple. O primeiro projeto construído com sucesso ocorreu em 1916 foi o avião automático **Hewitt-Sperry**, nomeado em homenagem aos seus inventores, este aeromodelo só conseguiu ser realizado graças aos trabalhos anteriores de Sperry com dispositivos giroscópios, o que garantiu a estabilidade em voo (NEWCOME, 2004)(ZALOGA, 2008).

Figura 2.1- Helicóptero não tripulado QH-50 DASH.



Fonte: (NEWCOME, 2004).

Com o êxito, Sperry atraiu a atenção da marinha norte-americana. O que resultou no desenvolvimento da **Curtiss Sperry Aerial Torpedo** durante o mesmo período a força aérea

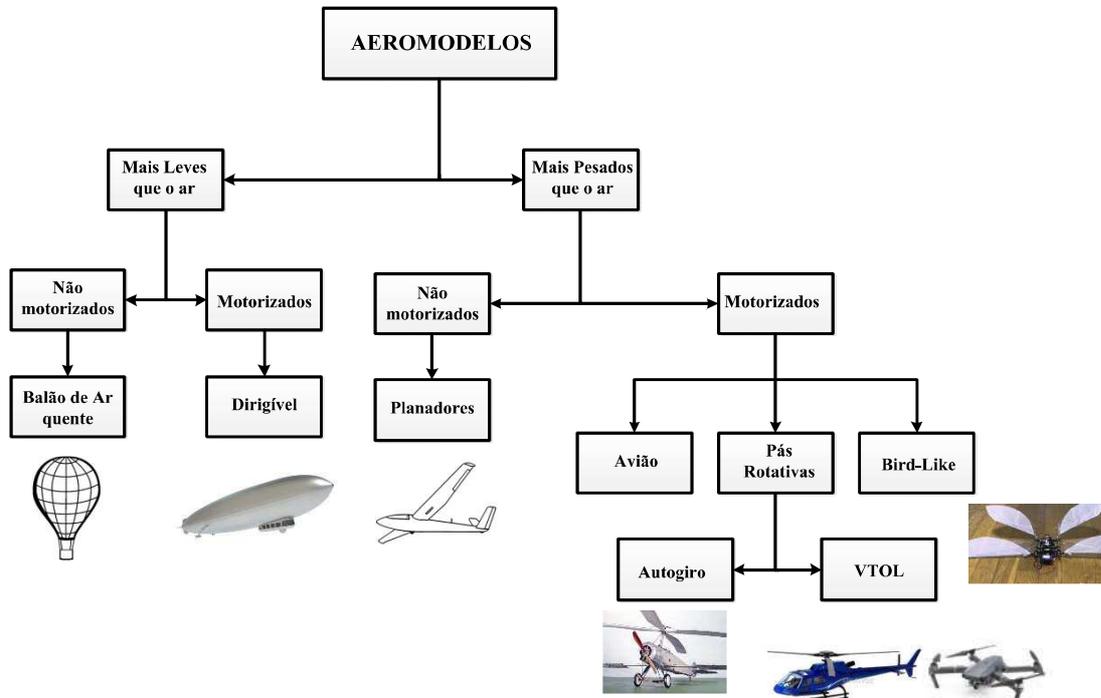
americana patrocinava o *Liberty Eagle Aerial Torpedo* of Charles Kettering. (ZALOGA, 2008). Nos anos de 1960 e 1970, o helicóptero não tripulado chamado *QH-50 Dash* construído pela empresa *Gyrodyne Company* foi adquirido pela marinha norte-americana. Este tinha como principal objetivo lançar torpedos antissubmarinos, porém, seu uso não ficou restrito a isto, logo foi utilizado para outras aplicações entre elas vigilância e transporte de cargas. (ZALOGA, 2008).

Hoje em dia, a construção de modelos VANTs procura focar que estes retornem a base de onde partiram diferente dos modelos criado por Sperry e pela força área americana, o que proporciona a vantagem de não serem mais destruídos a cada missão.

2.2 Quadricóptero

As classificações dos aeromodelos se dividem em duas áreas: os mais leves e os mais pesados, que o ar. Na categoria dos mais leves há duas divisões que são os não motorizados como exemplo: os balões de ar quente e os motorizados, os dirigíveis. Esse esquemático aparece na Figura 2.2. (BOUABDALLAH, 2007)

Figura 2.2- Classificação de Aeromodelos.



Fonte: (BOUABDALLAH, 2007).

Na divisão mais pesados que o ar também à mesma subdivisão dos mais leves com planadores representando os não motorizados. Porém a diferença aparece nos motorizados que possuem mais três categorias: os aviões, onde entra os diversos modelos como monomotores e bimotores; os **Bird-like**, que tem como construção baseada nos animais (pássaros) e finalmente os de pás rotativas que se repartem em autogiros ou conhecidos como girocópteros e os VTOL. (BOUABDALLAH, 2007)

Dando ênfase a exploração dos VTOL's (**Vertical Take-off Landing**) que são os aeromodelos de decolagem vertical entram os helicópteros: simples, coaxiais e os quadricópteros. Em relação aos outros modelos de decolagem vertical, os quadricópteros possuem vantagens como: mecânica simples, alta capacidade de carga e a redução dos efeitos giroscópios, mas sua principal desvantagem é o alto consumo energético. (BOUABDALLAH, 2007).

2.2.1 Histórico

O conceito de um aeromodelo de quatro hélices surgiu desde o começo do século XX. Precisamente, no ano de 1907, os irmãos Bréguet projetaram o primeiro helicóptero capaz de levantar voo carregando um ser humano. Eles o chamaram de **Bréguet-Richet Gyroplane 1**, o helicóptero era constituído por quatro motores, o que na verdade trata-se de um quadricóptero, disposto em X como visto na Figura 2.3. Com este invento, eles conseguiram abordar de forma bastante científica comparado a outros que buscavam resolver a problemática de se criar uma helicóptero na época. A partir dos cálculos de Louis Bréguet, os irmãos montaram a aeronave com as quatro hélices medindo 8 metros e 10 centímetros de diâmetro. (LEISHMAN, 2004)

Figura 2.3- Bréguet-Richet Gyroplane 1.



Fonte: (VALAVANIS e VACHTSEVANOS, 2014).

Entretanto, a máquina apresentava problemas de estabilidade, o que não foi levado em consideração durante o projeto, já que o principal objetivo era alcançar uma decolagem do solo com um piloto humano. (LEISHMAN, 2004)

A máquina foi relatada como tendo decolado a uma altura de 1,5 metros durante um curto período de tempo. Não tendo um controle de estabilidade, somente havia um acelerador do motor central que fornecia rotação para todos os motores. A máquina foi relatada com pouca controlabilidade. (LEISHMAN, 2004)

Durante o início século XX não foram desenvolvidos muitos projetos com aeronaves de sustentação rotórica. Os primeiros projetos viáveis de um quadricóptero foram desenvolvidos por George DeBothezt e Etienne Oemichen. O projeto do quadricóptero de Oemichen é o mais antigo relato de um veículo de quatro motores capaz de decolar e estabilizar no ar durante um voo que teve duração de 14 minutos em 1923. (BASTA, 2012).

Atualmente, a evolução da tecnologia, melhorias em estabilidade e segurança de voo permitiu o uso de quadricóptero em escala comercial. Hoje em dia, eles são utilizados em várias tarefas como: filmagem, monitoramento e na agricultura. (RIBEIRO RESENDE e IVES ROQUE PACHECO, 2014).

Figura 2.4- Quadricóptero Moderno.



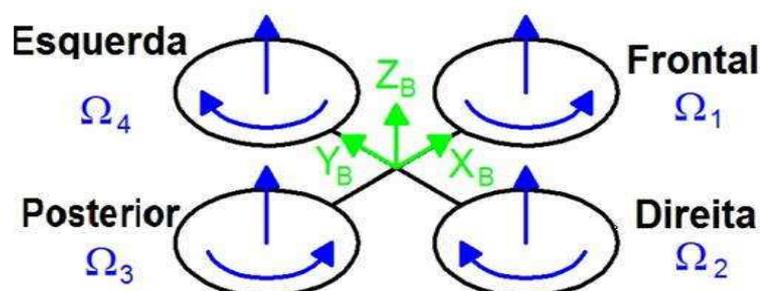
Fonte: (DE CASTRO JORGE e INAMASU, 2014).

2.2.2 Conceitos Básicos

Estruturalmente, a forma mais simples de se analisar um quadricóptero está na disposição mecânica em formato de cruz, na qual, cada haste está equidistante do centro da cruz. Nessa estrutura, cada ponta possui um motor que está acoplado por uma hélice através de uma caixa de engrenagem. Apesar de a estrutura parecer leve e fina, sua simplicidade e a rigidez das hastes confere ao molde uma grande robustez. Com todos os motores girando a mesma velocidade, a estrutura exibe um movimento de ascensão. Contudo, se houver

alguma diferença de rotação entre os motores a estrutura tenderá a rotacionar em seu próprio eixo.

Figura 2.5- Estrutura Simplificada do Quadrimotor.



Fonte: (BRESCIANI, 2008).

Para descrição dos movimentos básicos do quadrimotor não será necessária à análise sobre motor e a caixa de engrenagem, pois somente é necessário relacionar as diferenças na rotação das pás para demonstrar os seus movimentos.

A Figura 2.5 mostra as hastes pretas como a estrutura do quadrimotor com as setas verdes, azuis curvadas e azuis retas representando, respectivamente, o referencial de corpo fixo, a direção de rotação e a velocidade angular. Há também uma nomenclatura ao lado de cada pá representada por um disco mostrando as palavras: frontal, posterior, esquerda e direita.

Analisando a Figura 2.5, já é possível descrever uma condição em que as hélices frontal e posterior giram no mesmo sentido, o anti-horário, esta ideia é vista de forma oposta nas hélices da direita e esquerda que giram no sentido horário. Esta figura representa a estabilidade aérea visto que todos os motores rotacionam com a mesma velocidade.

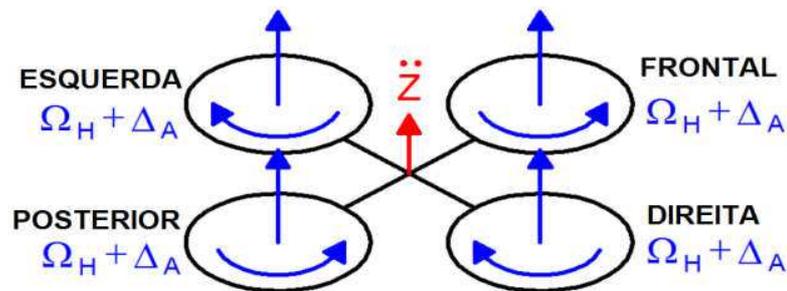
O quadrimotor têm seis graus de liberdade, na qual, se divide em duas descrições: o plano cartesiano (X, Y e Z) e os ângulos de rotação em cada eixo do plano cartesiano chamados de rolagem para eixo X, Arfagem para eixo Y e Guinada para o eixo Z. Embora possua seis graus de liberdade à estrutura montada por somente quatro motores controla diretamente quatro dos seis graus de liberdade. Já outros dois graus são controlados através da combinação graus que serão descritos a seguir. (BRESCIANI, 2008)

- Empuxo (Z, E1[N])

Esse comando acontece quando todos os motores possuem as velocidades iguais e sofrem uma redução ou aumento de velocidade na mesma proporção. Como consequência, a estrutura realizará deslocamentos descendentes ou ascendentes no eixo Z, ou seja, subida ou

descida do quadricóptero. A Figura 2.6 mostra como é feito esse comando na estrutura do quadrimotor.

Figura 2.6- Comando Empuxo.



Fonte: (BRESCIANI, 2008).

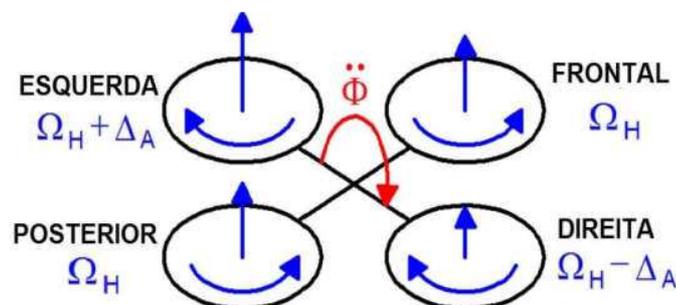
Percebe-se que todas as setas retas azuis possuem o mesmo comprimento indicando uma intensidade igual para cima e seus somatórios será um vetor de aceleração resultante para cima o (\ddot{z}) em vermelho. Adicionalmente na Figura 2.5 embaixo de cada descrição da orientação do Quadricóptero há uma expressão algébrica por $\Omega_H + \Delta_A$. Onde Ω_H é a velocidade que mantém a estrutura planando no ar e Δ_A é o acréscimo de velocidade rotacional dado em cada pá. (BRESCIANI, 2008)

Um fator bastante interessante para se pontuar é que o valor de Δ_A não pode ser muito grande para não gerar grandes não linearidades e saturação dos motores.

- Rolagem (θ , $E2[N\ m]$)

Nesse tipo de manobra o quadricóptero realizará um giro entorno do eixo X. Para que ocorra esse movimento é necessário que haja uma diferença rotacional entre a velocidade rotacional das pás da direita e esquerda. Resultando em uma variação angular no modelo, que mostrado na Figura 2.7, tenderá a imprimir um giro no sentido horário quando se é observado à figura pelo motor posterior. A este ângulo se dar o nome de Rolagem.

Figura 2.7- Comando Rolagem.



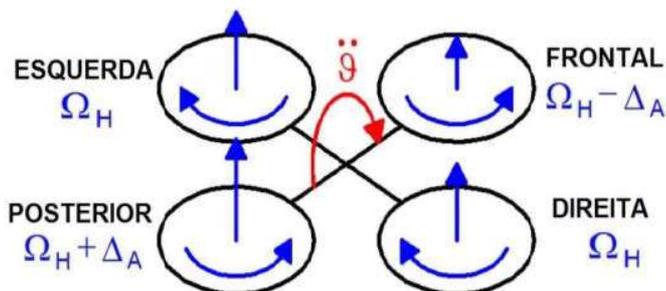
Fonte: (BRESCIANI, 2008).

Além disso, quando se é feito esse comando ocorre uma tendência de um pequeno deslocamento no eixo Y e é desta forma, indireta, que acontece o deslocamento no tal eixo (LUUKKONEN, 2011). Os valores de ΔA devem ser pequenos para que não haja fortes não linearidades e saturação no modelo, como também se esses valores forem altos a estabilidade de voo será comprometida.

Arfagem (ϕ , E3[N m])

Este movimento ocorre de forma similar à rolagem; porém, a rotação ocorre no entorno do eixo Y e os motores atuantes nesse fenômeno são o frontal e posterior que variam com proporção similar. Os valores de ΔA permanecem na mesma faixa no movimento de rolagem pelos mesmos motivos citados no movimento anterior (BRESCIANI, 2008).

Figura 2.8- Comando de Arfagem.



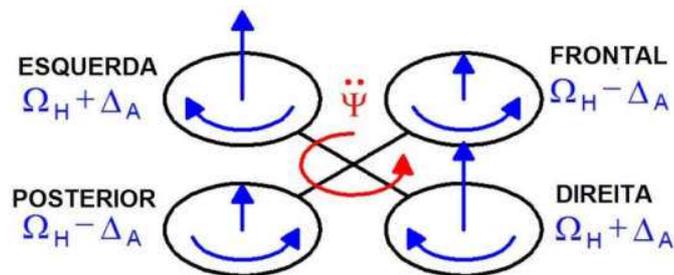
Fonte: (BRESCIANI, 2008).

Igualmente ao movimento de rolagem, o quadricóptero tenderá a ter um deslocamento, só que agora, no eixo X que também acontece de forma indireta (LUUKKONEN, 2011).

- *Guinada* (ψ , E4[N m]).

O movimento de guinada não é nada mais que o giro no eixo Z, todavia, este ocorre uma forma diferente aos movimentos anteriores e explica o porquê é excluída a necessidade de um motor de cauda presente nos helicópteros monomotores. Observando a Figura 2.9 é possível perceber que há um par de hélices girando no sentido horário enquanto o outro par no sentido inverso devido às velocidades angulares serem as mesmas o torque resultante será nulo.

Figura 2.9- Comando de Guinada.



Fonte: (BRESCIANI, 2008).

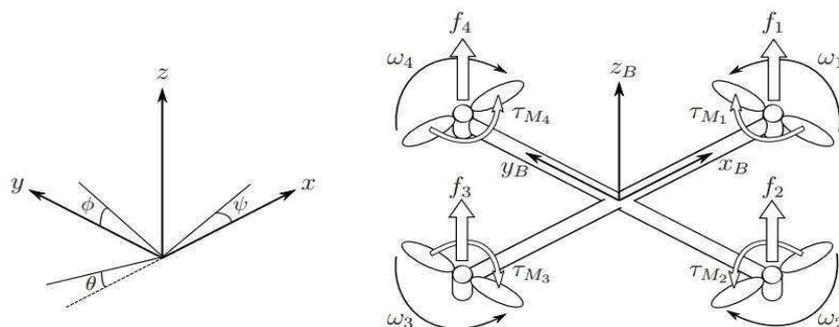
Na Figura 2.9, os motores (frontal e posterior) reduzem a velocidade em uma variação ΔA e os motores (esquerda e direita) aumentam a velocidade com uma variação ΔA para que seja mantida a estabilidade de voo. Como resultado a velocidade de rotação no sentido horário será maior que do sentido anti-horário o que resultará em um torque resultante no sentido horário. (BRESCIANI, 2008)

Entender estes movimentos é essencial para se criar um modelo próprio ou entender o modelo que será discutido no próximo tópico.

2.2.3 Modelo Newton-Euler

A Figura utilizada para descrever o modelo matemático Newton-Euler será representado pela Figura 2.10. Onde as siglas ω , T_m e f são, respectivamente, velocidade angular, torques e forças com a numeração indicada de cada motor. O modelo matemático a seguir é baseado no trabalho de (LUUKKONEN, 2011).

Figura 2.10- Referencial Inercial e de Corpo de um Quadricóptero.



Fonte: (LUUKKONEN, 2011).

O vetor ξ representa a distância da origem do veículo em relação ao referencial inercial nos eixos x , y e z . O referencial de corpo é descrito pelo vetor η representa as posições angulares de Euler em relação ao referencial inercial

que são: Arfagem ϕ , Rolagem θ e Guinada ψ . O vetor q contém os vetores de posição linear e angular. A notação para as posições angulares e deslocamentos são dadas por

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \eta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, q = \begin{bmatrix} \xi \\ \eta \end{bmatrix}. \quad (1)$$

A origem do referencial de corpo se dá no centro de massa do quadricóptero. Neste referencial, velocidades lineares e angulares são representadas pelos vetores V_b e v que são dados por

$$\mathbf{V}_b = \begin{bmatrix} V_{xB} \\ V_{yB} \\ V_{zB} \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \quad (2)$$

A matriz rotacional do referencial de corpo para o inercial apresenta-se por

$$\mathbf{R} = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}. \quad (3)$$

Para nível de simplificação S_x é $\text{seno}(x)$ e C_x é $\text{cos}(x)$. A matriz rotacional \mathbf{R} tem como característica a ortogonalidade, assim, \mathbf{R}^{-1} é igual a \mathbf{R}^T que é a matriz do referencial inercial para o de corpo.

A matriz de transformação das velocidades angulares do referencial inercial para o de corpo é \mathbf{W}_η , e do referencial de corpo para o inercial é \mathbf{W}_η^{-1} , como mostrado em (LUUKKONEN, 2011). As matrizes são dadas por

$$\dot{\eta} = \mathbf{W}_\eta^{-1} \mathbf{v}, \quad \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi / C_\theta & C_\phi / C_\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4)$$

e

$$\mathbf{v} = \mathbf{W}_\eta \dot{\boldsymbol{\eta}}, \quad \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}, \quad (5)$$

sendo T_x a $\tan(x)$. A matriz \mathbf{W}_η é invertível, se $\theta \neq \frac{(2\beta-1)\phi}{2}$, ($k \in \mathbb{Z}$). Onde β é a ordem da matriz \mathbf{W}_η .

A matriz inercial I que é dada por

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}, \quad (6)$$

Devido à consideração de que o quadricóptero possui uma estrutura simétrica com os quatro braços alinhados com os eixos x e y no referencial de corpo. Portanto para esta situação, tem-se que $I_{xx} = I_{yy}$.

Cada velocidade angular do rotor ω_i^2 multiplicado por uma constante k , sendo i variando de 1 a 4, cria uma força f_i na direção do eixo do rotor. A velocidade angular e aceleração do rotor cria um torque T_{M_i} entorno do eixo de cada rotor. A equação é dada por

$$f_i = k\omega_i^2, \quad T_{M_i} = b\omega_i^2 + I_M\omega_i. \quad (7)$$

Em que a constante de sustentação é dada por k , a constante de arrasto é b e o momento de inercia do rotor é I_M . De acordo com (LUUKKONEN, 2011), geralmente, o efeito de ω_i é então omitido por ser pouco influente no modelo.

A combinação das forças criadas pelos motores cria a força de empuxo T na direção do eixo Z do referencial de corpo, o torque τ_B consiste dos torques τ_ϕ , τ_θ , τ_ψ na direção correspondente dos ângulos do referencial de corpo.

$$T = \sum_{i=1}^4 f_i = k \sum_{i=1}^4 \omega_i^2, \quad T = \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix}, \quad (8)$$

$$\tau_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} lk(-\omega_2^2 + \omega_4^2) \\ lk(-\omega_1^2 + \omega_3^2) \\ \sum_{i=1}^4 \tau_{M_i} \end{bmatrix}, \quad (9)$$

Em que l é a distância entre o rotor e o centro de massa do quadricóptero.

As equações Newton-Euler são apresentadas a seguir. Considerando o quadricóptero uma estrutura de corpo rígido as equações de Newton-Euler podem ser aplicada para descrever a dinâmica do sistema. No referencial de corpo, a força requerida para a aceleração da massa $m\dot{\mathbf{V}}_B$ e a força centrífuga $\mathbf{v}(m\mathbf{V}_B)$ são iguais para a gravidade $\mathbf{R}^T \mathbf{G}$ e o empuxo total dos rotores \mathbf{T}_B . A EDO é dada por

$$m\dot{\mathbf{V}}_B + \mathbf{v}(m\mathbf{V}_B) = \mathbf{R}^T \mathbf{G} + \mathbf{T}_B. \quad (10)$$

De acordo com (LUUKKONEN, 2011). A força centrífuga é anulada no referencial inercial. Assim, somente a força gravitacional e a magnitude da direção do empuxo estão atuando na aceleração do quadricóptero. A equação é dada por

$$m\ddot{\boldsymbol{\xi}} = \mathbf{G} + \mathbf{R}\mathbf{T}_B, \quad (11)$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -\frac{g}{m} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} C_\psi S_\theta C_\phi + S_\psi S_\phi 0 \\ S_\psi S_\theta C_\phi - C_\psi S_\phi 0 \\ C_\theta C_\phi \end{bmatrix}. \quad (12)$$

No referencial de corpo, a aceleração angular inercial $\mathbf{I}\dot{\mathbf{v}}$, as forças centrípetas $\mathbf{v}(\mathbf{I}\mathbf{v})$ e a força giroscópica $\boldsymbol{\Gamma}$ são iguais para o torque externo $\boldsymbol{\tau}$. A equação de aceleração angular é dada por

$$\mathbf{I}\dot{\mathbf{v}} + \mathbf{v}(\mathbf{I}\mathbf{v}) + \boldsymbol{\Gamma} = \boldsymbol{\tau}, \quad (13)$$

$$\dot{\mathbf{v}} = \mathbf{I}^{-1} \left(- \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} I_{xx} p \\ I_{yy} q \\ I_{zz} r \end{bmatrix} - I_r \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega_r + \boldsymbol{\tau} \right). \quad (14)$$

Manipulando as Equações (13) e (14) chega-se a Equação (15).

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})qr/I_{xx} \\ (I_{zz} - I_{xx})pr/I_{yy} \\ (I_{xx} - I_{yy})pq/I_{zz} \end{bmatrix} - I_r \begin{bmatrix} q/I_{xx} \\ -p/I_{yy} \\ 0 \end{bmatrix} \omega_r + \begin{bmatrix} \tau_\phi/I_{xx} \\ \tau_\theta/I_{yy} \\ \tau_\psi/I_{zz} \end{bmatrix} \quad (15)$$

Em que $\omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4$ e I_r é a inércia gerada pelo conjunto hélice e motor. As acelerações angulares no referencial inercial são então anexadas das acelerações do referencial de corpo com a matriz de transformação \mathbf{W}_η^{-1} de sua derivada no tempo.

$$\begin{aligned} \ddot{\eta} &= \frac{d}{dt}(\mathbf{W}_\eta^{-1}\mathbf{v}) = \frac{d}{dt}(\mathbf{W}_\eta^{-1})\mathbf{v} + \mathbf{W}_\eta^{-1}\dot{\mathbf{v}} \\ &= \begin{bmatrix} 0 & \dot{\phi}C_\phi T_\theta + \dot{\theta}S_\phi/C_\theta^2 & \dot{\phi}S_\phi C_\theta + \dot{\theta}C_\phi/C_\theta^2 \\ 0 & -\dot{\phi}S_\phi & -\dot{\phi}C_\phi \\ 0 & \dot{\phi}C_\phi/C_\theta + \dot{\phi}S_\phi T_\theta/C_\theta & \dot{\phi}S_\phi/C_\theta + \dot{\theta}C_\phi T_\theta/C_\theta \end{bmatrix} \mathbf{v} \\ &\quad + \mathbf{W}_\eta^{-1}\dot{\mathbf{v}} \end{aligned} \quad (16)$$

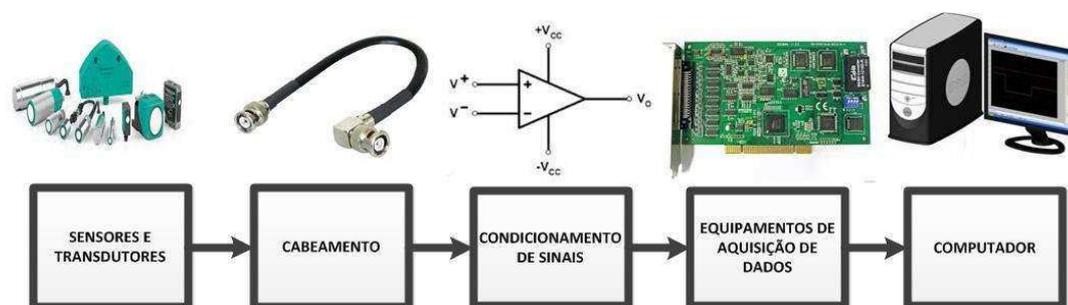
2.3 Sistema de Aquisição de Dados

A aquisição de dados é o nome do ramo da engenharia que é responsável por lidar com a coleta de informação de certo número de fontes analógicas e convertê-las para uma forma digital que seja mais adequada para a transmissão a um computador, uma impressora ou um **display** alfanumérico.

Um sistema de dados é um dos principais instrumentos utilizados por cientistas e engenheiros em suas pesquisas, principalmente, para testes, medições, automações entre outras aplicações. Os elementos básicos de um sistema de aquisição de dados são: (PARK e MACKAY, 2003)

- Sensores e Transdutores.
- Cabeamento e Condicionamento de Sinais.
- Placa eletrônica de Aquisição de dados.
- Programa de Aquisição de Dados.
- Computador (com um sistema operacional).

Figura 2.11- Sistema de aquisição de dados e seus elementos.



Fonte: Daniel Trovão.

2.3.1 Elementos de um Sistema de Aquisição

Os elementos de um sistema de aquisição podem ser divididos em: Sensores e Transdutores, Cabeamento, Condicionamento de Sinais, Equipamentos de Aquisição de Dados, Programa de Aquisição de Dados e um Computador. Cada um desses elementos é discutido neste tópico.

- **SENSORES E TRANSDUTORES.**

Os transdutores podem ser usados para detectar uma variedade de diferentes fenômenos físicos como: movimentação, sinais elétricos, energia radiante, térmica, magnética ou mecânica. Eles são utilizados para converter um tipo de energia para outro. Há diferentes tipos de transdutores cada um com características de entradas e saídas diferentes e sua aplicação fica a critério do projetista. (PARK e MACKAY, 2003)

Os sensores são elementos que produzem tensão e corrente de acordo com a variação do fenômeno físico que está sendo medido. Há dois tipos de sensores: ativos e passivos. Os sensores ativos são aqueles que necessitam de uma fonte externa para funcionar, como exemplo: microfones e sensores de temperatura (LM317). Já os sensores passivos não necessitam de qualquer fonte, exemplo: divisores de tensão, encoders, tacômetros. (H. ROSEMARY, 1997)

Sensores e transdutores pertencem à mesma categoria, transdutores.

- **Cabeamento.**

O cabeamento representa a conexão física dos sensores e transdutores para o equipamento condicionador de sinais ou o aparelho de aquisição de dados. Quando a aquisição de dados é feita de forma remotamente ao Computador, então o cabeamento prover

uma ligação física entre os equipamentos de aquisição e o PC podendo esta comunicação ser cabeada ou não. (PARK e MACKAY, 2003)

Como o cabeamento é geralmente a mais longa parte do sistema ela sempre está mais susceptível a efeitos como ruídos externos especialmente em ambientes industriais. Se feito um isolamento ou aterramento de forma correta evitará muitos problemas com este elemento do sistema. (EMILIO DI PAOLO, 2013)

- **Condicionamento de sinais.**

Os sinais elétricos gerados por transdutores são frequentemente necessários serem convertidos para uma forma aceitável de tensão ou corrente para serem lidos pelos aparelhos de aquisição, particularmente os conversores A/D recebem sinais de dados analógicos e passam para o formato digital. As principais tarefas que um condicionador de sinais devem fazer são: filtragem, amplificação, linearização, isolamento e excitação. (BAPTISTA, 2008)

A) Filtragem.

Em ambientes ruidosos, é muito difícil captar pequenos sinais, como exemplo: sensores termopares (em ordens de mV) sem seres comprometidos. Ruídos com ordem de magnitude igual ou maior que os sinais a serem medidos devem passar por filtros. (BAPTISTA, 2008)

B) Amplificação

Após a filtragem do sinal desejado, este deve ser amplificado para aumentar sua resolução. A resolução máxima que pode ser obtida amplificando um sinal de entrada é igual a faixa de variação do conversor analógico-digital que há nos sistemas de aquisição. (PARK e MACKAY, 2003)

C) Linearização.

Alguns transdutores, como termopares, não possuem uma linearidade para quantidade física que é requerida para medição; portanto, devem ser linearizados. Vários métodos de linearização são utilizados dependendo de como o sinal foi condicionado. Por exemplo, no caso dos termopares, provendo um equipamento para amplificar e linearizar o sinal ao mesmo tempo. (BAPTISTA, 2008)

D) Isolamento.

Equipamentos de condicionamento de sinais podem ser usados para prover isolamento dos sinais gerados pelos transdutores, onde há altos transientes como descargas atmosféricas, do computador. Assim, protege o sistema que monitora os dados de qualquer falha elétrica oriunda dos sensores. (BAPTISTA, 2008)

E) Excitação.

Um condicionador de sinais deve também prover uma excitação para alguns sensores ativos. Por exemplo: extensores, termistores e RTDs, (PARK e MACKAY, 2003)

- Equipamentos de aquisição de dados.

Os equipamentos de aquisição e controle de dados são disponíveis de diversas formas por diferentes fabricantes como placas que são ligadas diretamente a placa mãe dos computadores ou ligadas por portas seriais, USB ou qualquer outro tipo de conexão de dados. Esses equipamentos devem realizar as seguintes funções: (PARK e MACKAY, 2003)

- Realizar processamento e conversão para formato digital usando ADCs, de dados de medições analógicas dos transdutores para que os dados sejam transferidos para o um computador para ser mostrado, armazenado e analisado.
- Processar as informações digitais e transformá-las em sinais analógicos pela conversão DACs, dos sinais digitais do computador. O controle analógico de sinais é usado para controlar o sistema ou processo. (PARK e MACKAY, 2003)

Destas duas formas, que equipamentos de aquisição e controle de dados devem trabalhar.

- Programa de aquisição de dados.

O equipamento de aquisição de dados não trabalha sem um software, porque é o software que roda no computador e realiza a análise e controle dos dados para o usuário em forma de gráficos, tabelas ou sinalizações. (PARK e MACKAY, 2003)

Esses aplicativos rodam nos sistemas operacionais mais comuns como: Windows, Linux, Unix permitindo que o usuário possa rodar e alternar várias tarefas ou medições.

- Computador (com um sistema operacional).

As características dos computadores utilizados nos sistemas de aquisição afetam diretamente a coleta de dados. Pois, um computador com barramentos de baixa taxa de transmissão, baixo tempo de acesso ao disco rígido e pouca capacidade de processamento diminuirá a coleta contínua e a precisão dos dados adquiridos. Um bom procedimento para a escolha de um computador é saber o propósito do sistema de aquisição. (PARK e MACKAY, 2003)

Por exemplo, se o sistema necessitar coletar poucos dados em um curto ou longo período de tempo, talvez, não seja necessário à utilização de uma máquina com processadores potentes ou com barramentos em alta velocidade. Porém, se a aplicação necessitar processar

uma quantidade massiva de dados e realizar processos em tempo real será bem provável a necessidade de um computador de custo elevado. (PARK e MACKAY, 2003)

2.3.2 Características dos Sistemas de Aquisição

Para projetar um sistema de aquisição de dados, deve-se levar em conta saber de alguns parâmetros importantes. O foco desta seção é descrever a maioria das características dos sistemas para um melhor projeto que são Acurácia e precisão, Ruído, Tempo de Acomodação e Tempo de aquisição.

- **Acurácia e Precisão.**

A acurácia de um sistema de medição é o grau de quão próxima é a quantidade de medição do dado real. A precisão em um sistema de medição é chamada de reprodutibilidade ou repetitividade da medição. A acurácia relativa é uma medida que indica a capacidade do sistema de aquisição de Dados para corrigir os códigos de saída de acordo com a variação completa da escala. (EMILIO DI PAOLO, 2013)

- **Ruído**

É uma interferência entre dois sinais. Um fator, como ruído comum, indica a interferência que aparece em ambas às entradas de medidas. Cada medição gera ruído como uma combinação de mais sinais (EMILIO DI PAOLO, 2013)

- **Tempo de Acomodação**

O tempo de acomodação de um dispositivo eletrônico é o tempo transcorrido da aplicação de um degrau unitário ideal para o tempo. Os fatores que compõem o tempo de acomodação são: o atraso de propagação e o tempo requerido para obter o valor da saída. (EMILIO DI PAOLO, 2013)

- **Tempo de aquisição.**

O tempo de aquisição é a um aspecto de um sistema de aquisição de dados que mostra a presença de um conversor analógico digital. Um fator importante é que o tempo aquisição em cada situação define a acurácia do sistema. (EMILIO DI PAOLO, 2013).

2.4 Identificação de Sistemas

A definição de sistema se dar por um conjunto de diferentes tipos de variáveis que interagem produzindo sinais observáveis. Para controlar um sistema é necessário se ter um conhecimento sobre este para então ser capaz de controlá-lo. (AGUIRRE, 2004)

A identificação de sistema é uma poderosa ferramenta na mão de um projetista que deseja controlar um sistema e não possui recursos ou tempo para analisá-lo e precisa controlá-lo. (SANTOS TORRES, 2012)

Segundo a definição Aguirre,

“A identificação de sistemas é a área do conhecimento que desenvolve técnicas e algoritmos para obter modelos de sistemas dinâmicos através dos próprios dados gerados por eles (AGUIRRE, 2004)”.

Entrando nas subdivisões de sistemas existem três conceitos de modelagem: caixa branca, caixa cinza e caixa preta. Na modelagem caixa branca mais conhecida como modelagem fenomenológica ou conceitual descreve-se o sistema através de um conjunto de leis matemáticas bastante conhecidas como: aceleração, torque, energia e força. Contudo, como citado em trechos anteriores o engenheiro, às vezes, não dispõe de tempo ou recursos para modelar desta maneira o sistema de estudo. (SANTOS TORRES, 2012)

Consequentemente, utiliza-se das técnicas de identificação que na subdivisão de modelagem é classificada como caixa-preta ou empírica; na qual, não se há nenhum conhecimento sobre o sistema em estudo. Todavia, esse tipo de modelagem é raríssimo de acontecer o que leva a uma modelagem intermediária que é a definição de caixa cinza. (SANTOS TORRES, 2012)

Neste tipo de modelagem, o modelador possui um conhecimento relativo sobre o sistema de estudo. Este tipo de modelagem é o mais comum, pois normalmente para se criar um modelo o engenheiro deve ter um conhecimento básico sobre o processo.

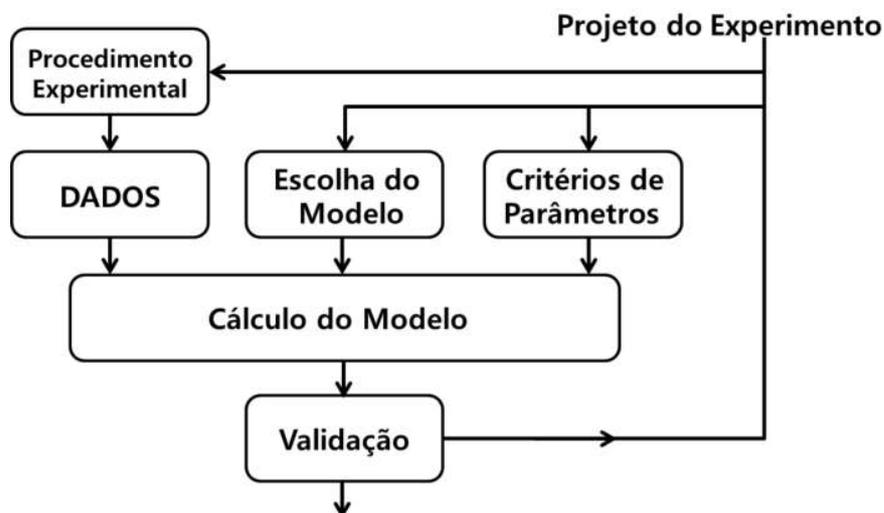
Assim, modelos matemáticos podem ser usados para projetos, operações de plantas, incluindo estudo de viabilidade econômica de processos industriais, as aplicações mais importantes são (SANTOS TORRES, 2012):

- Projeto de processos, plantas, equipamentos e respectivos sistemas de controle.
- Otimização de processos.
- Operação e pré-operação de controle de plantas conseguindo parâmetros de ajustes do controlador.

2.4.1 Etapas para Identificação

Na identificação de sistemas há etapas que devem ser obedecidas para que a identificação seja realizada com sucesso. Essas etapas são demonstradas a partir do fluxograma abaixo (LJUNG, 1999):

Figura 2.12- Fluxograma de uma Identificação de Sistema.



Fonte: (LJUNG, 1999).

Na figura 2.12 há uma descrição do procedimento para a identificação de um sistema. Esse fluxograma é composto basicamente por seis etapas e serão descritas somente cinco. Pois a etapa de cálculo do modelo é o algoritmo que irá rodar as equações do modelo escolhido e o critério de parametrização. As demais etapas serão descritas nos tópicos posteriores. Outro aspecto a observar é a validação do modelo, nesse tópico será analisado se realmente o modelo é procedente ou não, caso o resultado seja negativo é necessário uma revisão e análise de cada tópico anterior para se localizar o erro na identificação. (LJUNG, 1999)

2.4.2 Projeto Experimental

Nesta etapa o projetista deve elaborar como o procedimento irá ocorrer para a coleta de dados ser bem sucedida. Aqui são escolhidas quais variáveis de entrada e saída irão para coleta de dados, qual será a taxa de amostragem para coleta de dados, quais as condições para a realização dos experimentos e se há presença de ruído. (LJUNG, 1999)

Para realização de um experimento é importante que as condições, na qual o sistema será submetido, devam ser iguais ou parecidas com o que irá ser abordado no modelo para que este se torne o mais fidedigno com o sistema real e diminuindo o risco de insucesso na validação. (LJUNG, 1999)

Nas primeiras gravações de dados usa-se uma resposta ao degrau para o sistema. Procurando uma boa frequência de amostragem para coleta de dados.

2.4.3 Coleta de Dados

Nem sempre os dados coletados são aptos para a identificação ou quando são utilizáveis estes apresentam ruídos, flutuações, descontinuidades e outros fenômenos que afetam as medições. Portanto, é necessário o uso de técnicas que irão lapidar esses dados tornando possíveis para identificação. Geralmente quando se realizar uma coleta certas características vêm atreladas com o sinal coletado como: (LJUNG, 1999)

- Distúrbios em alta frequência.
- Ocasionais falhas, falta de dados ou descontinuidades nos dados gravados.
- Flutuações e compensações para baixas frequências de distúrbios e características periódicas.

Quando se trabalha com dados há várias de formas de se lidar com as flutuações e compensações. Desta forma, há algumas técnicas; porém duas técnicas para tratar com flutuações que se destacam em relação às outras devidas as suas baixas complexidades e os altos usos por pesquisadores. Elas são: a remoção dos distúrbios pelo tratamento completo dos dados e um modelo de ruído para lidar com os distúrbios. (LJUNG, 1999)

Para compensação de sinais há outras seis formas de tratamento:

- 1. Deixar os sinais de entrada e saída entorno de um ponto de equilíbrio
- 2. Substituir as médias das amostras.
- 3. Estimar uma compensação explícita.
- 4. Usar modelo de ruído com integração.
- 5. Estender o modelo de ruído.
- 6. Utilizar um filtro passa-alta

2.4.4 O MODELO ARX

Na escolha do modelo a representar o sistema SYMA X6, o modelo função de transferência discreta foi escolhido por apresentar a ideia mais fácil de ser entendida. O principio de funcionamento por detrás deste tipo de modelo pode ser apresentado pela expressão a seguir. As deduções e representações matemáticas do modelo ARX serão baseadas nos trabalhos de (SANTOS TORRES, 2012) e (AGUIRRE, 2004). Vendo as equações (16) e (17)

$$y(t) = b_1 u(t-1) + b_2 u(t-2) + \dots + e(t) \quad (17)$$

$$y(t) = B(q)u(t) + e(t), \quad (18)$$

Sendo o $B(q) = \sum_{k=1}^n B_k q^{-k}$, q^{-k} um operador de atraso, assim o operador ($q^{-k} = u(t-k)$) é utilizado simplesmente para simplificar as equações e melhorar o entendimento do assunto. O termo $e(t)$ é um erro de ruído branco. Estendendo a análise da equação (16) podemos expandi-la nos termos de $y(t)$ e $u(t)$ correlacionando os fatores de polos e zeros da função de transferência que o projetista desejar que pode ser visto na equação (18).

$$"y(t) + A_1 y(t-1) + \dots + A_{n_a} y(t-n_a) = B_1 u(t-1) + \dots + B_{n_b} u(t-n_b) + e(t) \quad (19)$$

Onde n_a é relacionado ao número de polos e n_b é a quantidade de zeros da função de transferência. Colocando em evidência os termos de $[A_1, \dots, A_{n_a}]$ no $y(t)$ e $[B_1, \dots, B_{n_b}]$ no $u(t)$. Temos a seguinte relação.

$$A(q^{-n_a}) = 1 + A_1 q^{-1} + A_2 q^{-2} + \dots + A_{n_a} q^{-n_a} \quad (20)$$

$$B(q^{-n_b}) = 1 + B_1 q^{-1} + B_2 q^{-2} + \dots + B_{n_b} q^{-n_b} \quad (21)$$

Correlacionando os termos de $A(q^{-n_a})$ e $B(q^{-n_b})$ e deixando $y(t)$ isolado no lado esquerdo da equação (18). Temos a equação (21).

$$y(t) = \frac{B(q)}{A(q)} u(t) + \frac{1}{A(q)} e(t) \quad (22)$$

A relação de (21) é conhecida como modelo ARX (Modelo Auto-Regressivo com entradas Exógenas) onde o termo $\frac{B(q)}{A(q)}$ é o vetor de auto-regressores e n_b deve ser menor que n_a para que o modelo tenha convergência.

Este modelo é uma estrutura bastante usada nos procedimentos de identificação de sistemas. Então, para a realização desse trabalho o modelo utilizado foi o ARX devido a sua simplicidade e facilidade de entendimento.

2.4.5 Estimação de Parâmetros

A estimação de parâmetros é uma etapa posterior à escolha do modelo, escolhido o modelo é necessário se determinar os parâmetros, pois esses descrevem características como estabilidade e controlabilidade do sistema através da relação de dados de entradas e saídas.

Na estimação de parâmetros, os dados de entrada e saída e a própria análise dos parâmetros são usados para compreender melhor a dinâmica do sistema alvo. (SODERSTROM e STOICA, 1989)

A área da estimação de parâmetros pode ser dividida em duas classificações: a estimação *online* e a *off-line*. A estimação *online* atualiza os dados a cada instante que os dados aparecem, esse processo é tratado forma recursiva que é um termo para os algoritmos que atualizam e processam os dados a cada ciclo de instrução. Já a *off-line* realiza a estimação tratando todo o conjunto de dados no modo Batelada. O modo Batelada é uma forma de processar o conjunto de dados de uma só vez. (SANTOS TORRES, 2012)

As características desejadas para um estimador são a variância mínima e a não polarização. Por definição, a variação mínima significa:

“Um estimador de variância mínima se a variância do valor estimado por este é mínima em relação à variância do valor estimado por qualquer outro estimador. (SANTOS TORRES,2012,p.43)”

A definição de Não-Polarizado se dá por: “A polarização de um estimador é a diferença entre a média dos valores estimados dos parâmetros e seus valores reais. Portanto, um estimador é dito não polarizado quando o valor médio de cada parâmetro estimado converge para seu respectivo valor verdadeiro. (SANTOS TORRES,2012,p.42)”

Outra característica desejada a um estimador é a linearidade, pois este fator transforma todo o processo de estimação em simples cálculos, já que a linearização de um sistema é a forma mais simples de análise e de cálculo onde a máquina gasta menos tempo computacional calculando grandes quantidades de dados, pois não precisa utilizar nenhum processo por aproximação numérica.

Quando um estimador apresenta as características de linearidade, variância mínima e não polarização. Este tipo de estimador denomina-se com o nome de BLUE e é considerado o melhor estimador de parâmetros dados às características ideias e a facilidade de cálculo com a linearidade. (MENDEL, 1987)

O estimador utilizado nesta monografia foi o mínimo quadrático. Ele foi desenvolvido por Gauss e aplicado primeiramente para cálculos astronômicos. Gauss propôs e comprovou

matematicamente que este tipo de estimador era a melhor forma de se aproximar a um valor real. De acordo com ASTROM, o método de mínimos quadráticos é definido como:

“[...] a soma dos quadrados das diferenças entre os valores efetivamente observados e os valores calculados, multiplicados por números que medem o grau de precisão, é um mínimo (ASTRÖM e WITTENMARK, 2008), p. 42)”.

Na dedução algébrica dos mínimos quadráticos foram utilizados os trabalhos de (DA SILVA, 2015) e (KEESMAN, 2011).

Uma forma interessante de estimar os parâmetros é pela análise da predição do erro que é:

$$\varepsilon(t) := (y(t) - \varphi(t)^T \theta)^2 \quad (23)$$

Sendo $\theta = [A_1, \dots, A_{n_a}, B_1, \dots, B_{n_b}]$ que é composto primeiramente pelos coeficientes dos polos de A_1, \dots, A_{n_a} , e em seguida dos coeficientes dos zeros B_1, \dots, B_{n_b} da função de transferência. Este vetor é chamado vetor de parâmetros.

Já $\varphi(t)^T = [-y(t), \dots, -y(t - n_a + 1), u(t - 1), \dots, u(t - n_b + 1)]$ é o vetor de regressores que é composto pelos valores medidos de entrada e saída do sistema.

O valor de $\varepsilon(t)$ deve ser o mais próximo de zero e a multiplicação de $\varphi(t)^T \vartheta$ representa o $y_{estimado}$.

Desta forma,

$$\varepsilon(t) := (y_{real} - y_{estimado})^2 \quad (24)$$

Analisando o erro médio quadrático se pode obter,

$$J(\vartheta) = \sum_{t=1}^N \varepsilon^2(t) = \sum_{t=1}^N (y(t) - \varphi(t)^T \theta)^2 \quad (25)$$

Onde $J(\vartheta)$ é uma função escalar. O objetivo é tentar diminuir o valor dessa função através dos valores dos parâmetros e assim achar uma função que mais se adeque a relação de dados de entrada e saída do sistema. Utilizando a equação (24) e com conhecimento de operações matriciais chega-se a seguinte equação.

$$J(\vartheta) = \varepsilon^T \varepsilon = (y^T - \varphi^T \theta^T)(y - \varphi \theta) \quad (26)$$

Pela propriedade matricial de que $\varphi^T \theta^T = (\varphi \theta)^T$ (veja em KEESMAN no apêndice A na parte de propriedade matricial e operações). Sendo J um escalar a sua minimização só irá acontecer se o gradiente de J com o respectivo ϑ é zero.

$$\frac{\partial a^T \theta}{\partial \theta} = a \quad (27)$$

E

$$\frac{\partial \vartheta^T A \vartheta}{\partial \vartheta} = (A + A^T) \vartheta \quad (28)$$

Que pode ser facilmente verificado, escrevendo todos os elementos e tomando os derivados. Portanto, como J é uma função escalar, os termos individuais são escalares para que, Com $y, \Phi \theta \in R^N$, $y \varphi^T \theta = \theta \vartheta y^T$

$$\begin{aligned} J(\vartheta) &= y^T y - y^T \varphi \theta - y \varphi^T \theta^T + \varphi^T \theta^T \varphi \theta \\ &= y^T y - 2y \varphi^T \theta^T + \varphi^T \vartheta^T \varphi \theta \end{aligned} \quad (29)$$

Conseqüentemente, tomar a derivada em relação à ϑ dá um zero para o primeiro termo, $-2\varphi^T y$ para o segundo termo, e $(A + A^T)\theta$ com matriz simétrica $A := \Phi^T \Phi$ ($A^T = A$) para o último termo, então

$$\frac{\partial J(\theta)}{\partial \theta} = -2y \varphi^T + 2\varphi^T \varphi. \quad (30)$$

(Veja o Apêndice A de (KEESMAN, 2011) para detalhes). O gradiente de J (θ) é zero se e somente se

$$\varphi^T \varphi \hat{\theta} = \varphi^T y \quad (31)$$

Que são chamadas de equações normais. De (30) podemos deduzir a estimativa ordinária dos mínimos quadráticos multiplicando ambos os lados com $(\varphi^T \varphi)^{-1}$:

$$\hat{\theta} = (\varphi^T \varphi)^{-1} \varphi^T y \quad (32)$$

2.4.6 Validação de Dados

Quando se chega à etapa de validação de dados é necessário anteriormente avaliar três aspectos sobre o modelo de sistema que atenderá de forma satisfatória o que o engenheiro deseja obter. Estes aspectos são: (LJUNG, 1999)

- O modelo: se é linear, não linear ou em espaço de estado.
- O tamanho do modelo: a ordem do sistema (no caso em espaço de estado), grau do polinômio característico (funções de transferência) ou o número de neurônios (caso rede neural). (KEESMAN, 2011)
- O critério de parametrização.

Escolhidos os três aspectos gerais do sistema a próxima etapa será analisar a qualidade deste modelo. Os melhores modelos possuem duas características: (LJUNG, 1999)

- Flexibilidade: é a capacidade de descrição a diferentes situações do sistema.
- Passimônia: descrita como a não utilização de parâmetros desnecessários.

Analisada a qualidade do sistema, outro ponto chave é o preço do modelo. Na qual, se pauta mais três aspectos cruciais para o bom desempenho do sistema. Estes são definidos como:

O primeiro é a complexidade do algoritmo de identificação. Este fator é de longe algo necessário ter em mente devido ao custo computacional que pode ser alto para determinado processador afetando no seu desempenho. Uma aplicação onde se é bastante frisado este ponto é no controle adaptativo. (AGUIRRE, 2004)

O segundo, a pretensão de uso do modelo. Nesse caso, se a ordem do modelo for elevada poderá ser requeridos muito mais cálculos e também mais complexo será seu controle. Preferencialmente, o projetista deve escolher o modelo de menor ordem que atenda as necessidades pedidas. (KEESMAN, 2011)

E por fim, a propriedade do critério de função.

Como já foi discutido anteriormente. As primeiras considerações que devem ser feitas são: tipo, ordem e parametrização do modelo. Após essas considerações deve-se passar para a próxima etapa que é aplicar técnicas de análise preliminar dos dados que se divide em: (LJUNG, 1999)

- A estimação do tipo de modelo.
- Análise se o sistema é SISO ou MIMO.

Depois de feita as primeiras considerações e aplicada às técnicas de análise preliminar vem-se a etapa de testes do modelo que podem ser por:

- Comparação do modelo com uma segunda coleta de dados. Que nada mais é que a coleta de dados de entrada e saída do sistema passando os novos dados de entrada no modelo gerado pela identificação de sistema e comparando saída do modelo com os novos valores de saída dos dados coletados. Um dos critérios para avaliar se a comparação das saídas foi satisfatória ou não é a utilização do critério de predição de erro final (FPE). (LJUNG, 1999)
- Testes Estocásticos.

Desta forma, termina-se a identificação de um sistema quando se faz o projeto do experimento, análise dos dados, a escolha do modelo, o critério de parametrização e por fim a validação de dados. Após o sucesso destas etapas o sistema encontra-se apto para ser utilizado e para aplicação de diversas técnicas de controle.

3 PROJETO DA PLACA DA ACBOARD E PROCEDIMENTOS PARA A IDENTIFICAÇÃO DE SISTEMA DO QUADRICÓPTERO

Neste capítulo abordam-se os materiais utilizados assim como os métodos aplicados para o sucesso deste trabalho. Ele é dividido em suas seções Materiais Para Identificação do Sistema SYMA X6 e Métodos Para Identificação do Sistema SYMA X6.

3.1 Materiais Para Identificação do Sistema SYMA X6

Os materiais utilizados foram diversos; contudo, alguns materiais foram usados de maneira mais frequente que outros e são destacados e descritos neste subtópico de metodologia. Segue abaixo os materiais utilizados nesse trabalho e os componentes de cada material.

Equipamentos utilizados:

- Quadricóptero **Syma X6**.
- MATLAB.
- Sistema de aquisição (Ni-MyDaQ).
- Microcontrolador Arduino.
- Placa de aquisição de dados.

3.1.1 Quadricóptero Syma X6

O objeto de estudo utilizado nesse trabalho foi o quadricóptero SYMA X6. Este drone é um aparelho utilizado para atividades recreativas, mas possui boas características para este trabalho, pois possui um sistema estável de voo, uma duração de bateria boa, um modelo estrutural adequado ao trabalho e baixo custo que para o mercado brasileiro está em uma faixa de R\$606,00 reais e R\$ 99,00 dólares na **Amazon**. (AMAZON, 2017)

Figura 3.1- SYMA X6.



Fonte: (AMAZON, 2017).

Dentre todas as vantagens citadas sobre este modelo a que foi diferencial para este trabalho foi o modelo estrutural. Porque este modelo apresentava fácil possibilidade, a princípio, de levantamento de um modelo matemático mais simples devido ao fato de apresentar:

- Motores DC
- Estrutura simples em cruz.
- Caixa de engrenagem.
- Baixo peso.

Dado esses fatores, a estrutura parecia ser bastante simples de calcular seus coeficientes, pois os motores DC possuem modelos matemáticos bem consolidados, a estrutura em forma de cruz era similar ao levantamento matemático citado no Modelo Newton-Euler em Quadricópteros, sobre a caixa de engrenagem há o trabalho de (BRESCIANI, 2008), na qual, o autor utiliza um aeromodelo com as mesmas características e inclusive modela a relação de engrenagem que embora seja um quadricóptero criando por ele poderia se encaixar perfeitamente ao utilizado neste trabalho.

E por fim, o baixo peso que em relação entre a potência de empuxo possibilitava uma carga adicional de peso sem comprometer o desempenho da máquina.

3.1.2 MATLAB

O software desenvolvido pela Mathworks MATLAB (Matricial Lab) foi o programa utilizado para desenvolver algoritmos para tratamento de dados coletados, modelagem matemática, controle, simulações e identificação de sistemas. Dentro desta ferramenta foi usado o aplicativo: (MATHWORKS, 2016)

- System Identification.

O “System Identification” foi utilizado para levantar o modelo matemático do quadricóptero baseado nos valores das variáveis de entrada e saída selecionadas para criar um modelo matemático. Desenvolvido por Ljung, este aplicativo do MATLAB é uma poderosa ferramenta de auxílio a pesquisadores e engenheiros, pois possui uma alta gama de técnicas de identificação e várias possibilidades de representação de modelos (função de transferência, espaço de estado, polinomial etc.).

3.1.3 Sistema de Aquisição de Dados (Ni-mydaq)

O sistema de aquisição de dados Ni-mydaq, Figura 3.2, não é somente uma placa de aquisição de dados, mas também uma plataforma que auxilia estudantes iniciantes em eletrônica, pois está possui um multímetro, osciloscópio, fonte de tensão variável, gerador de função, entradas e saídas digitais incorporadas ao dispositivo. No caso deste trabalho, foram utilizadas todas suas funções além da aquisição de dados realizadas pelas entradas analógicas. (NATIONAL INSTRUMENTS, 2017)

Figura 3.2- Placa de Aquisição (NI-myDAQ).



Fonte: (NATIONAL INSTRUMENTS, 2017).

Nesse aparelho há os seguintes componentes:

- Entradas Analógicas: (\pm)10V de 200KS/s por canal.
- Saídas Analógicas: (\pm)10V de 200KS/s por canal.
- 2 Canais de áudios.
- 3 Fontes de Tensão (+15V,-15V e 5V) de potência de 500 a 100mW.
- 1 Multímetro Digital.

Com esses componentes e o software NI-myDAQ é possível simular os seguintes aparelhos.

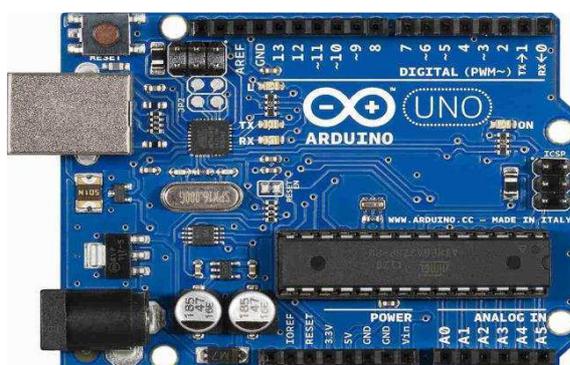
- Multímetro Digital.
- Osciloscópio.
- Analisador de Bode.
- Gerador de função.
- Analisador de dinâmica de sinais.
- Gerador de onda arbitraria.
- Leitor digital.
- Escritor digital.

Todos esses componentes podem ser visto e analisados no manual deste instrumento que pode ser acessado no guia do usuário NI-myDAQ.

3.1.4 O Microcontrolador Arduino

O microcontrolador Arduino, Figura 3.3, é utilizado como cérebro da placa de aquisição desenvolvida para este trabalho, devido a sua leveza (peso), fácil programação, alto número de bibliotecas disponíveis, baixo custo (aproximadamente R\$ 30,00 reais) e uma gama de dispositivos que podem ser facilmente conectados ao microcontrolador.

Figura 3.3 - Microcontrolador Arduino Uno.



Fonte: (CORP, 1998).

O Arduino utilizado foi o Microcontrolador Atmega328P em uma plataforma física (UNO) e com esta placa foi possível desenvolver uma placa de aquisição em formato de **Shield**. As características do Atmega328P são (CORP, 1998):

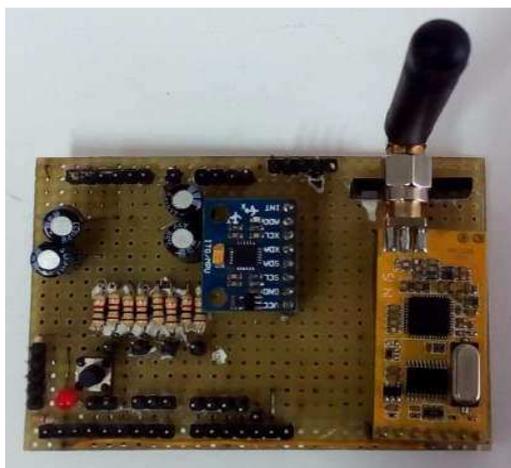
- O tipo de memória de programa: Flash.
- Memória de programa: 32 KB.
- Velocidade de CPU: 20 MIPS.
- Memória Ram: 2 KB.
- Data EEPROM: 1024 bytes.
- Periféricos de comunicação digital: 1-UART,2-SPI e 1-I2C.
- Canais ADC: 6 canais
- Canais PWM: 6 canais.

Essas características são mais que suficiente para desenvolver uma placa de aquisição de dados para pequenas aplicações.

3.1.5 Placa de Aquisição de Dados

Com a plataforma física (UNO), a ACBOARD foi possível ser desenvolvida em formato **shield**. Assim, esta placa é composta pelos seguintes componentes que serão decorridos neste tópico que são:

Figura 3.4 - Placa de aquisição ACBOARD.



Fonte: Daniel Trovão.

- Acelerômetro e Giroscópio MPU-6050.
- Placa de comunicação em radio frequência (RF) APC 220.
- Componente de escrita em cartão SD.
- Divisores de tensão.
- Sonar HC-SR04.

Como primeiro componente descrito o MPU-6050 é um acelerômetro, giroscópio e sensor de temperatura de 6 graus de liberdade para o acelerômetro e três eixos para o giroscópio. O sensor de temperatura também mede uma grande variedade de -40 a 85 C. Este sensor possui alta precisão devido ao conversor analógico de 16bits em cada canal. O que possibilita a captura dos canais X, Y e Z no mesmo instante. (EIRELI, 2017)

Figura 3.5 - MPU6050.



Fonte: (FELIPEFLOP, 2017)

Características:

- Chip: MPU-6050.
- Tensão de Operação: 3-5V.
- Conversor AD: 16bits
- Comunicação: Protocolo padrão I2C.
- Faixa do Giroscópio: $\pm 250, 500, 1000, 2000^\circ/s$.
- Faixa do Acelerômetro: $\pm 2, \pm 4, \pm 8, \pm 16g$.

A placa de comunicação APC 220 é responsável pelo envio de dados da ACBOARD para o computador, ela possui uma baixa potência, uma rápida resposta e uma boa taxa de envio de dados com uma fácil instalação sem a necessidade configurar qualquer protocolo de comunicação. (FELIPEFLOP, 2017)

Figura 3.6 - Placa de comunicação sem fio APC 220.



Fonte: (EIRELI, 2017).

O dispositivo possui dois conjuntos gêmeos de antenas e ambos podem ser utilizados como transmissor e receptor bastando ser plugado uma interface USB para se comunicar ao computador, o módulo que será ligado ao Arduino os pinos RX e TX serem conectados ao terminais TX e RX da placa física (UNO) e um software de recepção de dados, no caso o foi o Hercules.

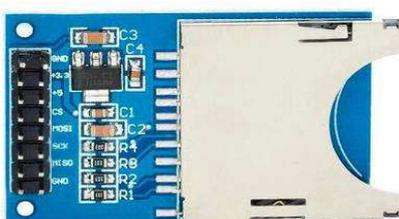
Características:

- Frequência de Trabalho: 420MHz até 450MHz.
- Tensão: 3.5 – 5.5V
- Corrente: $<25-35mA$.
- Temperatura de Trabalho: $-20C - 70C$.
- Faixa: 1.2Km.
- Interface: UART/TTL.
- **Baud Rate**(no Ar): 1200-19200bps.
- Buffer do Receptor: 256 bytes.

- Peso: 30g.

Uma das grandes vantagens da placa de aquisição de dados é versatilidade de gravação de dados tanto com o computador, APC 220, ou sem utilizando o dispositivo de cartão SD. O objetivo da utilização desse dispositivo é poder gravar informações caso não seja possível utilizar a comunicação em radio frequência devido à longa distância ou caso, deseje-se utilizar o quadricóptero de forma autônoma.

Figura 3.7 - Leitora de Cartão SD.



Fonte: (FELIPEFLOP, 2017)

Os divisores de tensão foram utilizados para adaptar os valores de tensão que alimentam os motores e a bateria, respectivamente 6.4V e 7.4, para o valor de tensão de leitura do canal ADC do Arduino Atmega328p que é 5V. Os resistores que compõem o divisor de tensão são ambos de 1K Ω .

O sonar HC-SRF04, Figura 3.8, é vastamente utilizado para aplicações de detecção de distância sua interface é bem simples para ser conectada ao Arduino contendo 4 pinos: Vcc de 5V, GND, Trigger e Echo. Os pinos Trigger e Echo é que são responsáveis pela leitura da distância que é na faixa de 2cm a 400cm. Ele é bastante robusto e simples de se realizar a interface com o Arduino bastando somente a utilização de uma biblioteca “Ultrasonic.h” e alguns comandos para leitura da distância em centímetros ou polegadas.

Figura 3.8 - SONAR HF-SRF04



Fonte: (FELIPEFLOP, 2017)

3.2 Métodos Para Identificação do Sistema SYMA X6

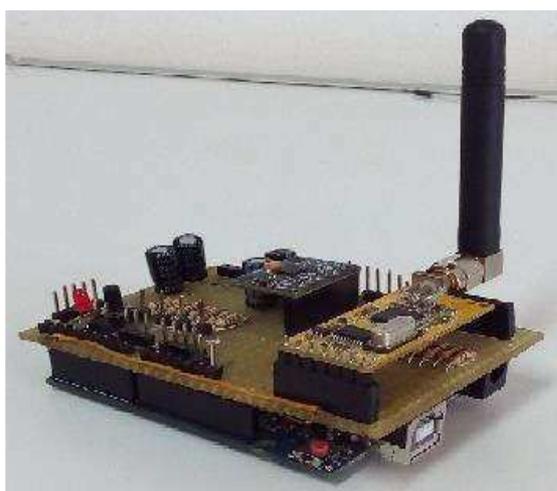
Este subcapítulo será dividido em cinco etapas que serão: Desenvolvimento da placa de aquisição de dados ACBOARD, o algoritmo e processo de aquisição de dados da ACBOARD, o projeto do experimento utilizado para identificação do sistema, análise e tratamento dos dados coletados, o algoritmo de identificação de sistema e a validação de dados. Também será descrito como foram os procedimentos para as realizações destas.

3.2.1 Desenvolvimento da Placa de Aquisição de Dados ACBOARD

Dado ao fato do quadricóptero SYMA X6 não ser capaz de levantar voo com cargas acima de 175 gramas, exemplo: a NI-Mydaq, também não possuía os sensores: sonar, acelerômetro e giroscópio. Portanto, a construção da placa de aquisição de dados (ACBOARD) buscou o baixo peso (aproximadamente 78 gramas com os sensores), de forma a não influenciar severamente o voo, e acoplar os três sensores.

Esta placa é constituída, como já foi mencionada no subcapítulo anterior, de um módulo de comunicação (APC 220) ou de gravação de dados (SD Card), o acelerômetro e giroscópio (MPU 6050) para realizar de leituras angulares e um sonar (HC SRF04) para medir a distância do quadricóptero em relação ao solo. Todos esses equipamentos são conectados em uma shield, a ACBOARD, que fica interconectada a uma placa física (Arduino UNO). Como exposto na Figura (3.9).

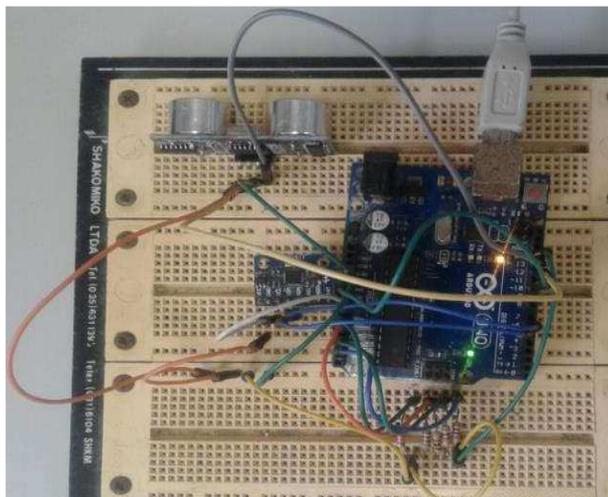
Figura 3.9 - ACBOARD acoplada à placa física.



Fonte: Daniel Trovão.

A construção da placa foi realizada após a montagem e conexão dos sensores e módulos de comunicação ao Arduino Uno em um protoboard.

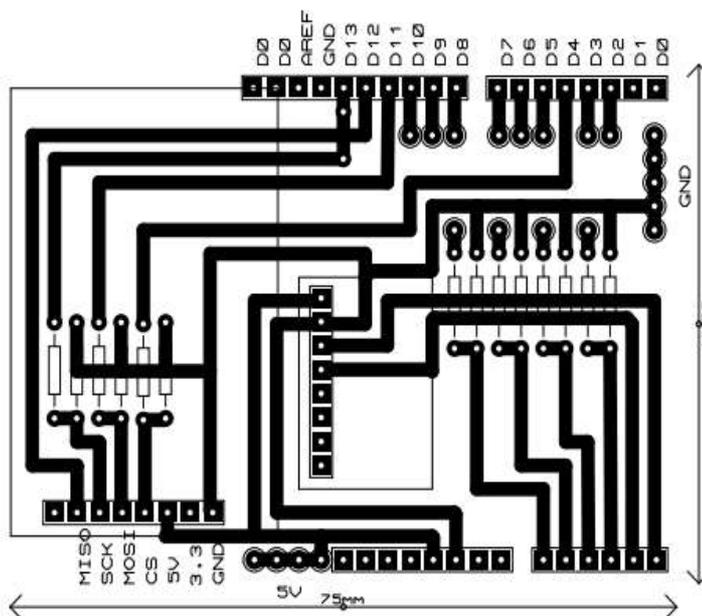
Figura 3.10 - Construção da ACBOARD de dados na protoboard.



Fonte: Daniel Trovão.

Logo após o sucesso da montagem foi realizada a prototipagem da placa de aquisição de dados no formato de **Shield** para conexão dos módulos e sensores com a placa física.

Figura 3.11: Esquema de prototipagem da placa de aquisição de dados.



Fonte: Daniel Trovão.

Finalmente, a construção da placa de aquisição acoplada ao Arduino Uno, como foi visto na Figura (3.9) e foi acoplada ao quadricóptero na FIGURA 3.12.

Figura 3.12 - ACBOARD com SYMA X6.



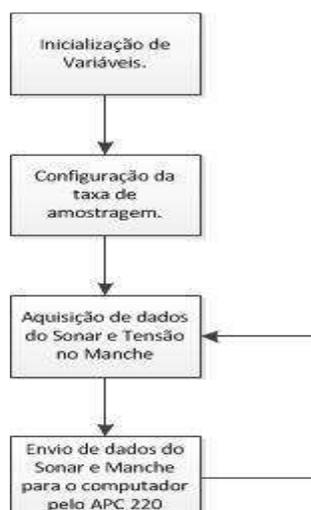
Fonte: Daniel Trovão.

3.2.2 O Algoritmo e Processo de Aquisição de Dados da ACBOARD

O algoritmo responsável por realizar a aquisição de dados foi criado na interface de desenvolvimento da placa física Arduino Uno, o ARDUINO IDE, e a linguagem de programação utilizada foi **Wiring**, que é baseada na linguagem C.

Outro ponto a se frisar, é que o programa foi construído somente para atender a necessidade de identificação do sistema do experimento. Este algoritmo é descrito no fluxograma abaixo e pode ser visto com mais detalhes no ANEXO 7.1 Código de Leitura de Tensão do Manche e Altura do Quadricóptero SYMA X6.

Figura 3.13 - Fluxograma do algoritmo de aquisição da ACBOARD.

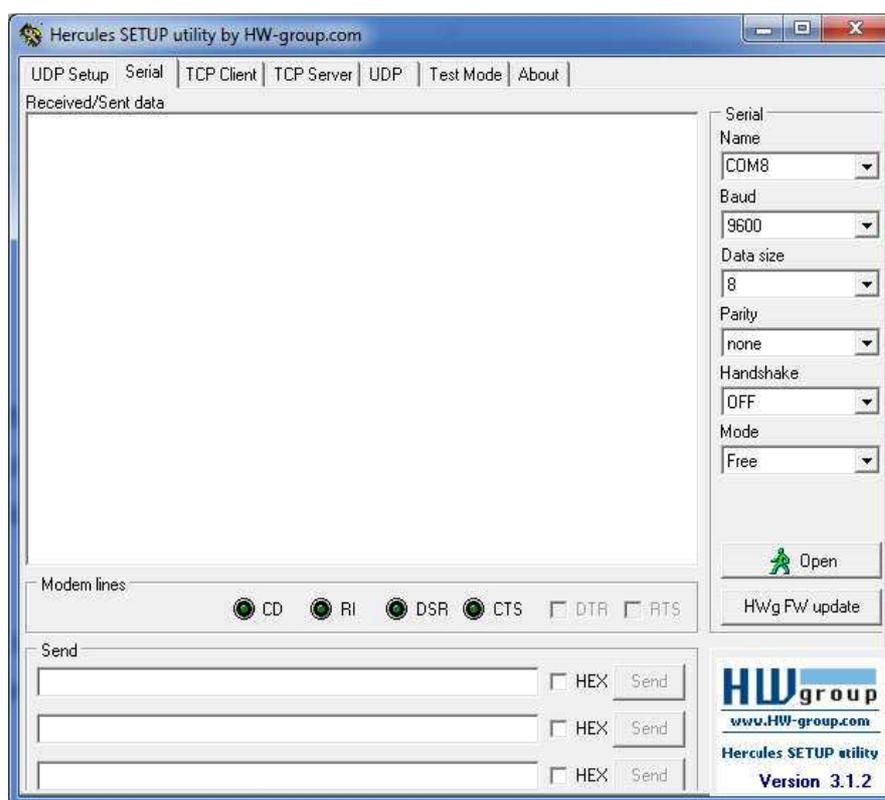


Fonte: Daniel Trovão.

Este algoritmo é utilizado para envio dos dados do sonar e tensão do manche da ACBOARD para o computador que os armazena. A aquisição do sonar ocorre a uma determinada taxa de amostragem configurada pelo programa.

Para receber os dados transmitidos pelo módulo de comunicação (APC220) da ACBOARD é necessário utilizar o programa Hercules, que é um software de leitura de porta de porta serial. Seu principal diferencial está na geração de um log de dados que pode ser lido pelo MATLAB.

Figura 3.14 - Tela do programa Hercules.

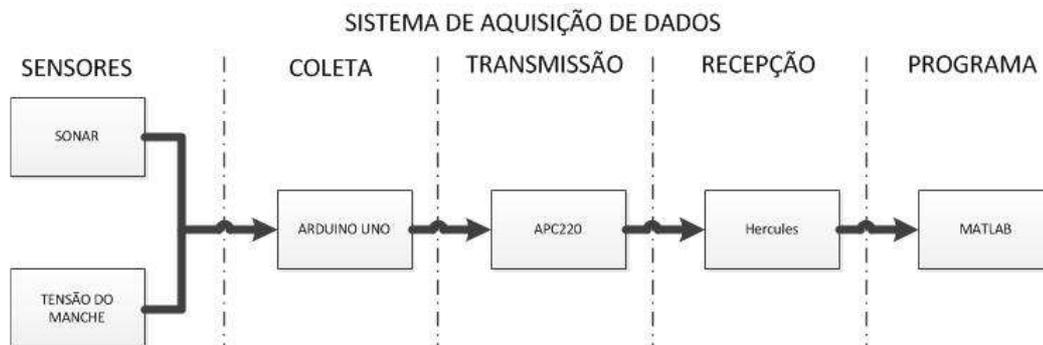


Fonte: Daniel Trovão.

Este software só foi utilizado devido à dificuldade integrar a plataforma MATLAB com a porta COM onde estava conectado o receptor APC 220.

Em geral, o processo de aquisição e transmissão da ACBOARD se dá ao seguinte diagrama de processo.

Figura 3.15 - Processo de aquisição e transmissão de dados ACBOARD.



Fonte: Daniel Trovão.

3.2.3 O Projeto do Experimento Utilizado para Identificação do Sistema.

Para projeto experimental, os seguintes procedimentos foram realizados:

1. A identificação seria realizada para somente uma variável de entrada e uma de saída tornando o sistema SISO.
2. A variável de entrada escolhida foi o valor referente à tensão manche no controle do Quadrimotor [V] e a variável de saída foi à altura adquirida pelo sonar [cm].
3. O drone foi ligado a um cabo guia para evitar qualquer desvio para eixo X e Y causado pelo vento ou qualquer outro fator de distúrbio.
4. A taxa de amostragem foi fixada em 5 milissegundo para justamente fazer a leitura mais precisa do transitório de subida do quadricóptero.

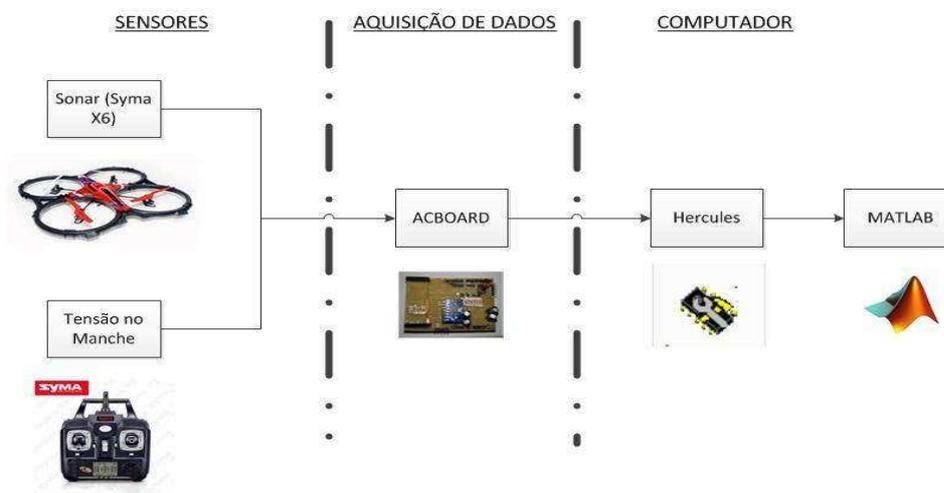
Esses procedimentos experimentais foram responsáveis por tornar o modelo mais simples. Além disso, o dado de saída, altura [cm], foi coletado durante a subida do drone, ou seja, o manche era variado a formar um sinal de entrada em rampa e a saída era analisada com o objetivo de se obter uma modelagem em malha aberta do Syma X6.

Para execução do experimento foram necessários os seguintes itens:

- Quadricóptero SYMA X6
- ACBOARD.
- O controle do Quadricóptero.
- Um cabo guia.
- Softwares: MATLAB e Hercules.

A seguinte ilustração descreve a execução do experimento:

Figura 3.16 - Diagrama do processo do procedimento experimental.



Fonte: Daniel Trovão.

O sonar está localizado em baixo da fuselagem do SYMA X6. No processo do procedimento experimental é possível ver duas coletas: a aquisição proveniente do sonar da ACBOARD, já discutida na etapa anterior, e o conversor analógico-digital da ACBOARD conectada aos terminais no manche de subida do controle.

3.2.4 Análise e Tratamento dos Dados Coletados

Após a análise dos dados coletados de entrada e saída, o sinal de entrada, o valor analógico-digital do Arduino UNO de variação entre 0 a 5,0V, observou que a entrada variou aproximadamente de 0 a 2,8V. Já o valor do sonar tinha a conversão apropriada em centímetros e seu valor máximo pelo próprio experimento não iria passar de 150 cm. O algoritmo de coleta de dados, que pode ser visto com mais detalhes no ANEXO 7.2 Análise de dados, tem como função:

- Limitar um valor máximo dos dados coletados de entrada e saída. O objetivo é evitar picos discrepantes com as demais amostras adquiridas.
- Selecionar os melhores intervalos de dados para identificação ou validação. O intuito é selecionar o melhor conjunto de dados de uma dada aquisição para identificação ou validação. Pontuando, os dados de validação e identificação podem ser extraídos de um único de evento de aquisição ou não.
- Análise da taxa de amostragem média do sinal. Esta função tem como mecanismo analisar os dados de temporização transmitidos pelo Arduino, calcular as taxas de amostragem do sinal, verificar se há variação entre elas e tirar uma média.

3.2.5 Filtragem dos Dados

Contudo, mesmo com a análise dos dados coletados eles apresentavam distúrbios, perdas em certos intervalos, deslocamentos e desregulações. Sendo assim, foi necessária aplicar uma das táticas discutidas em 2.4.3: Coleta de dados, que foi a utilização de um filtro. Nesse caso, o filtro de **Kalman** que está nos ANEXOS em Filtragem de Dados para tal tarefa. Este filtro foi desenvolvido com as seguintes etapas:

- Carregamento dos dados (Entrada e Saída).
- Tamanho das variáveis de entrada e Saída.
- Determinação dos valores iniciais como: variância calculada, variância do processo, covariância inicial, Ganho de **Kalman** inicial e Estado inicial estimado.
- O processo de execução do filtro de **Kalman**.

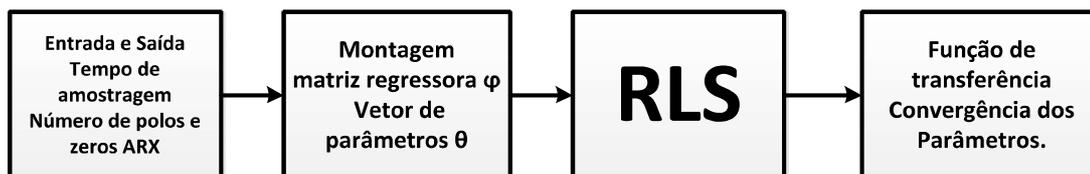
3.2.6 O Algoritmo de Identificação de Sistema

A escolha do modelo para representação do sistema foi à função de transferência em tempo discreto. Porque esta apresenta maior facilidade de criação de um algoritmo de identificação para o experimento realizado. Além disso, análise dos coeficientes do numerador e denominador poderia ser facilmente programada no modelo ARX que foi o tipo de modelo escolhido.

Esse algoritmo foi criado de forma recursiva, ou seja, seus parâmetros e sua saída são atualizados à medida que os valores são carregados, analisando os dados e estimando o próximo valor por valores anteriores. A preferência pelo algoritmo de forma recursiva em detrimento da forma batelada foi feita por: fatores de redução do custo computacional e a possibilidade de trabalhos em tempo real. Conseqüentemente, esse algoritmo poderá ter aplicações para controle adaptativo.

O algoritmo de identificação foi desenvolvido de forma que o usuário possa determinar o número de zeros, número de polos e o tempo de amostragem dando maior autonomia ao usuário de escolher como será sua função de transferência. Segue no diagrama de blocos o funcionamento do algoritmo.

Figura 3.17- Algoritmo ARX.



Fonte: Daniel Trovão.

O algoritmo ARX pode ser dividido em quatro etapas: a inicialização de dados que será preenchida pelo usuário, a verificação e adequação dos dados para algoritmo recursivo, o próprio algoritmo recursivo e por fim gráficos de análise de parâmetros, erros e a função de transferência. Este algoritmo está em mais detalhes em ANEXOS na Identificação ARX .

3.2.7 Validação do Sistema

O algoritmo de validação tem como objetivo pegar a função de transferência gerada pela identificação e avaliar se ela pode ou não representar o sistema real. Nesta etapa, primeiramente, coleta-se novos dados de entrada e saída. Os novos dados de entrada são usados na função transferência que gera uma série de valores de saída. Posteriormente, é feita a comparação entre os valores de saída da função de transferência e os valores de saída dos novos dados coletados e feita uma análise gráfica de erro entre eles. Este algoritmo pode ser visto nos ANEXOS em Validação do Sistema.

4 EXPERIMENTOS E RESULTADOS

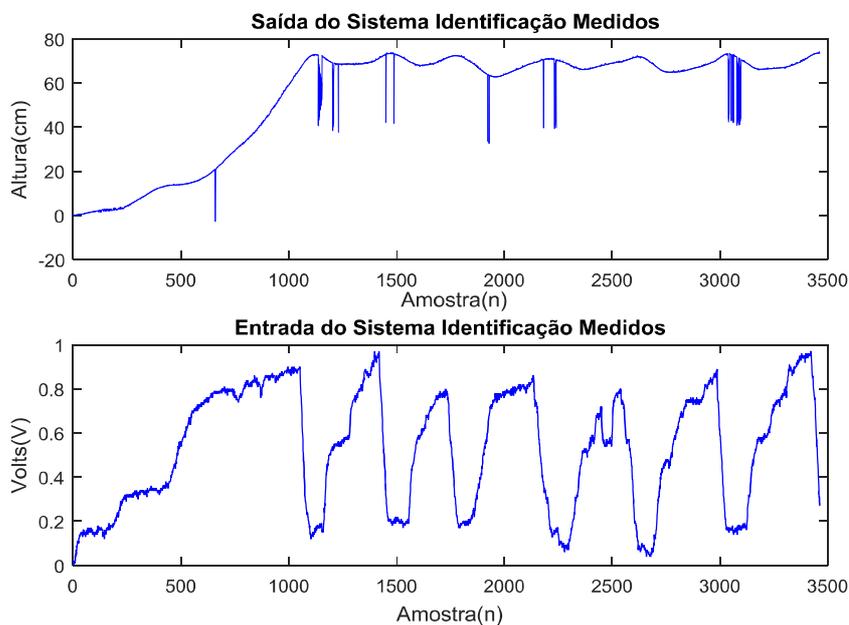
Neste capítulo serão apresentados os resultados aos métodos do processo de identificação do sistema SYMA X6. Esse capítulo será dividido em: Dados de entrada e saída coletados e filtrados, Análise do algoritmo ARX, Validação de Dados e comparação com a ferramenta “identification” do MATLAB.

4.1 Dados de Entrada e Saída Coletados e Filtrados

Nesta seção serão apresentados os dados coletados e após os processos de filtragem.

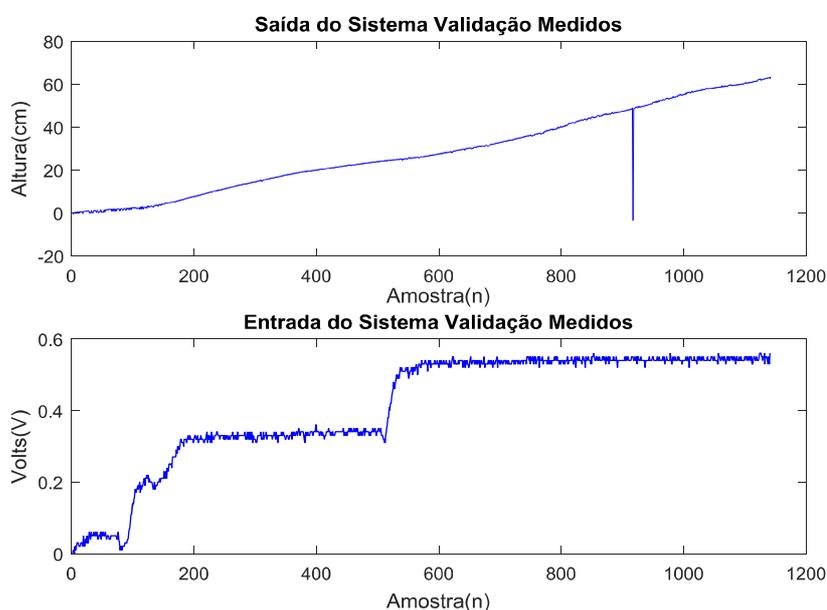
Quando foi realizada a coleta dos dados de entrada e saída para a identificação e validação o procedimento para identificação foi variar o sinal de entrada; conseqüentemente, variar a saída. Com a intenção de que se o sistema identifica-se um conjunto de dados oscilante seria mais fácil para validar um sinal de saída mais simples como uma rampa. Como será visto nas Figuras 4.1-2.

Figura 4.1 - Dados coletados para identificação.



Fonte: Daniel Trovão.

Figura 4.2 - Dados coletados para validação.



Fonte: Daniel Trovão.

A taxa de amostragem de ambos os conjuntos de dados foi de 5 milissegundos ou 0,005 segundos.

Depois de coletados os dados de Identificação e Validação foi realizada à filtragem dos dados utilizando o filtro de Kalman relatado no Capítulo 3. Nessa filtragem os valores dos parâmetros do filtro são apresentados na Tabela 4.1.

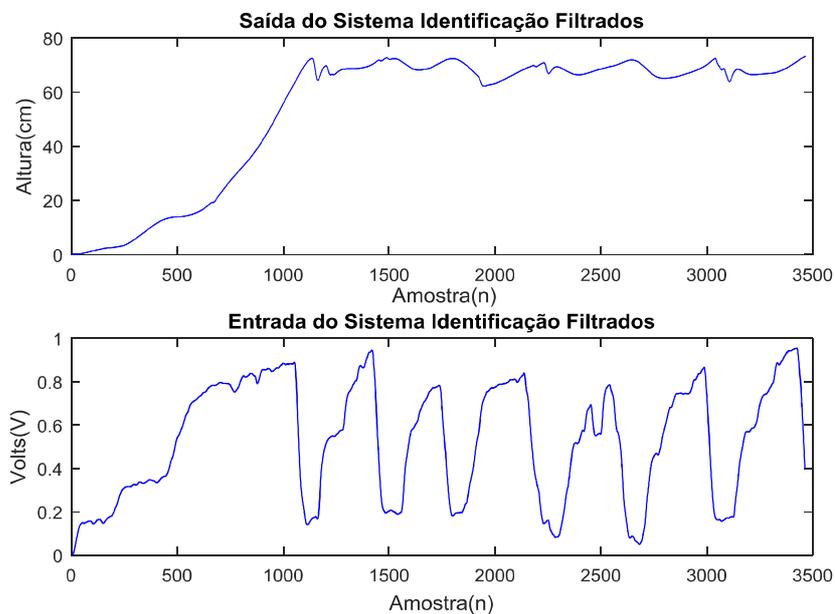
Valores configurados.	Valores
Variância Calculada (R)	0,00075
Variância do processo (Q)	0,0001
Covariância P inicial (P)	0,0000001
Ganho de Kalman Inicial (K)	0
Estado Estimado Inicial (Ei)	0

Tabela 4.1 - Valores pré-configurados do filtro de Kalman.

Estes valores já são os pré-configurados no próprio algoritmo de filtragem. Desta forma, não foi necessário nenhum tipo de alteração tanto para a filtragem dos sinais coletados de identificação quanto os de validação. Porque ambos mostraram uma resposta satisfatória.

Depois de configurada a filtragem, o processo pode ser realizado e os sinais tanto de validação quanto de identificação mostraram os seguintes resultados.

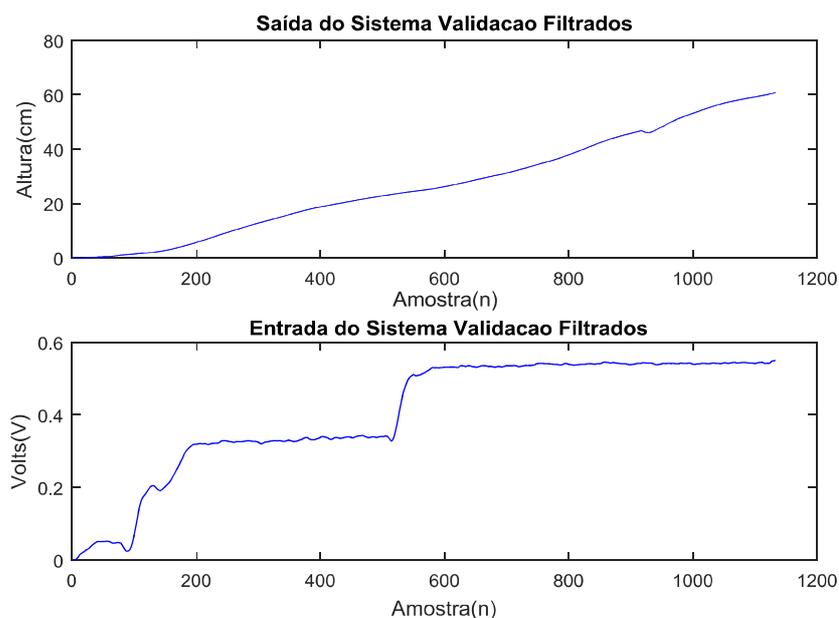
Figura 4.3 - Dados de identificação filtrados.



Fonte: Daniel Trovão.

Analisando os dados de entrada e saída da filtragem é possível perceber uma atenuação dos ruídos principalmente nas partes onde ocorrem as oscilações do conjunto dos sinais de identificação na Figura 4.1.

Figura 4.4 - Dados de validação filtrados.



Fonte: Daniel Trovão.

Com relação aos dados de validação é possível perceber uma grande atenuação de um pico em um intervalo de amostras entre 800 a 1000, Figura 4.2, como também uma filtragem de ruídos no sinal de entrada. A Figura 4.4 apresenta-se de forma mais suave em relação à Figura 4.2.

4.2 Algoritmo ARX

Logo após o processo de filtragem dos sinais foi iniciado o algoritmo ARX recursivo com o intuito encontrar uma função de transferência apropriada para os dados identificados. A relação da função de transferência utilizada foi de nenhum zero e dois polos com a taxa de amostragem de 5 milissegundos.

Outras relações de polos e zeros de funções de transferência também foram utilizadas; porém, todas apresentaram resultados muito próximos e por isso se deu a preferência pela função de menor grau para que possa ser mais fácil, futuramente, aplicar algum tipo de controlador.

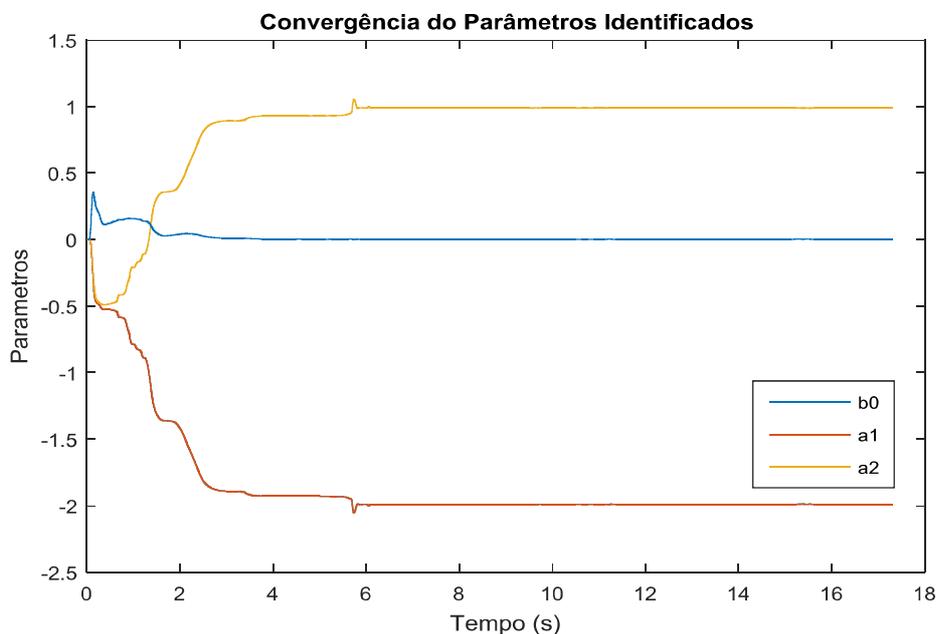
Valores de Entrada para o algoritmo ARX.	
Número de Zeros	0
Número de Polos	2
Taxa de Amostragem	0,005s

Tabela 4.2 - Valores de Entrada para o Algoritmo ARX.

Com esses valores o algoritmo de identificação ARX pode ser rodado e os seguintes gráficos foram apresentados com a função de transferência discreta é dada por

$$G(z) = \frac{0,002516}{1 - 1,988z^{-1} + 0,968z^{-2}} \quad (33)$$

Figura 4.5 - Convergência dos Parâmetros Identificados.



Fonte: Daniel Trovão.

O primeiro gráfico a ser mostrado é a Figura 4.5, onde é possível ver os coeficientes do numerador em roxo que está próximo de zero, pois seu valor é de 0,002516 e os dois coeficientes do denominador, na qual, o coeficiente do primeiro grau que está em laranja de valor -1,988 e o segundo grau em amarelo com 0,968. Estes valores são pertencentes ao vetor de parâmetros que é dado por

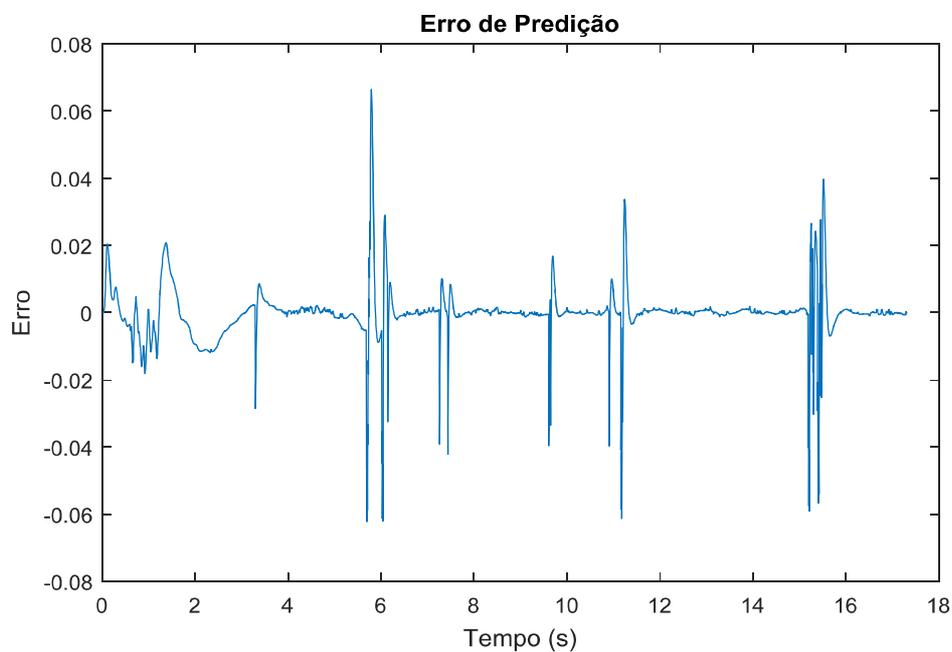
$$\vartheta = [0,002516 \quad 1,988 \quad 0,968] \quad (34)$$

Outro fator interessante de se observar é a convergência dos coeficientes se estabilizaram a partir dos seis segundos mostrando o sucesso no alcance dos valores dos parâmetros com o método de mínimos quadráticos.

Na Figura 4.6, o gráfico de Erro de predição é uma relação entre os dados da “Saída do sistema identificação filtrados” na Figura 4.3 com valor de saída estimada pelo método de mínimos quadráticos de acordo com a equação dado por

$$\text{Erro de Predição}(t) = y_{\text{real}}(t) - y_{\text{estimado}}(t) \quad (35)$$

Figura 4.6 - Erro de predição do estimador.



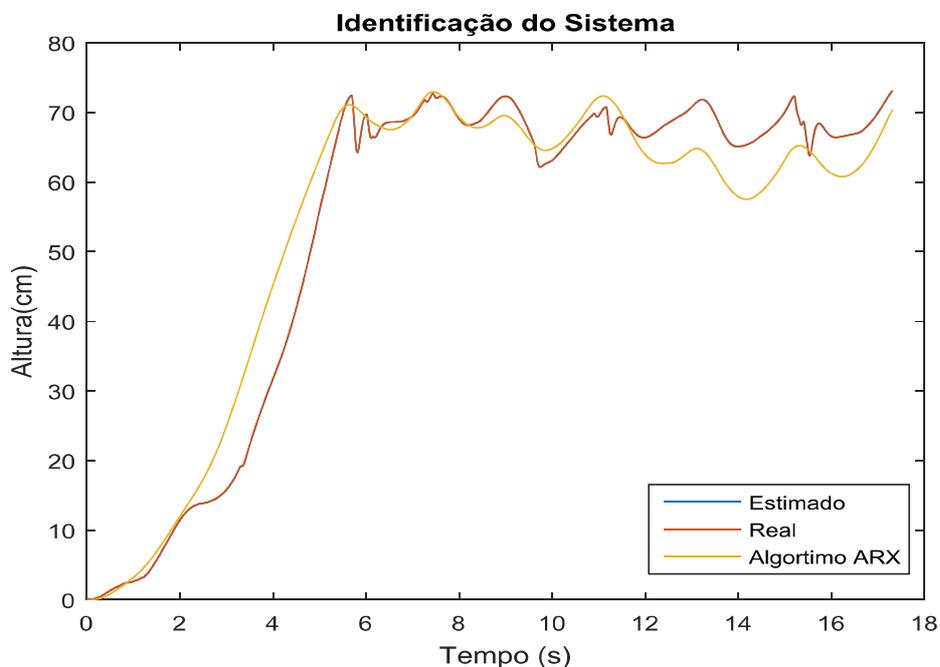
Fonte: Daniel Trovão.

Com o erro de predição é possível analisar se realmente os coeficientes estão atendendo função de transferência pretendida. Neste caso, o erro de predição está próximo de zero, o que é considerado ideal para o modelo. É também percebido que há pequenas oscilações devido à própria natureza do sinal de identificação que mesmo filtrado apresentava pequenas distorções.

O próximo gráfico é justamente utilizando a função de transferência $G(z)$ com o sinal de dados “Entrada do sistema identificação filtrado” da Figura 4.1 com o comando do MATLAB `lsim` (MATHWORKS, 2017). Este comando tem semelhanças o com comando `step` muito utilizado para aplicar um degrau em funções de transferência.

Na Figura 4.7 é possível perceber que o modelo gerado pelo algoritmo ARX tenta se aproximar ao valor de saída real da identificação, contudo por ser uma aproximação nem todos os pontos serão idênticos. No entanto, devido à natureza do sinal, o modelo ainda sim é válido para representar o sistema real.

Figura 4.7 - Identificação do SYMA X6.



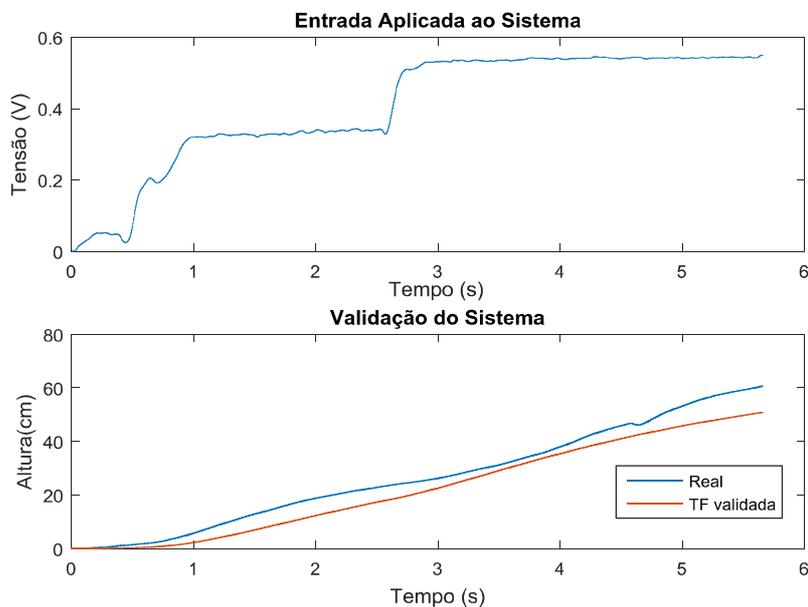
Fonte: Daniel Trovão.

4.3 Validação do SYMA X6

Observando na Figura 4.6 e 4.7, o processo de identificação é considerado concluído. Desta forma, a próxima etapa foi iniciada: a validação. Esta etapa é bastante similar ao gráfico da Figura 4.7, Identificação do Sistema, o mesmo comando do MATLAB é utilizado, o `lsim`. Contudo, a diferença está no sinal de entrada do `lsim` que será os dados “Entrada do sistema validação filtrado” e o resultado será comparado com o sinal saída do sistema validação filtrados vistos na Figura 4.2. Esta comparação é observada na figura a seguir.

Na Figura 4.8, a legenda “TF validada” é observada bem próxima da legenda “Real” mesmo possuindo uma distorção. Esta função de transferência pode ser utilizada para análise e controle de futuros trabalhos. Portanto, o sistema é considerado validado pelo processo de segunda leva de dados citado na seção Validação de Dados vistos no Capítulo 2.

Figura 4.8 – Validação do Sistema SYMA X6.



Fonte: Daniel Trovão.

4.4 Comparação com a Ferramenta “Identification” do MATLAB®

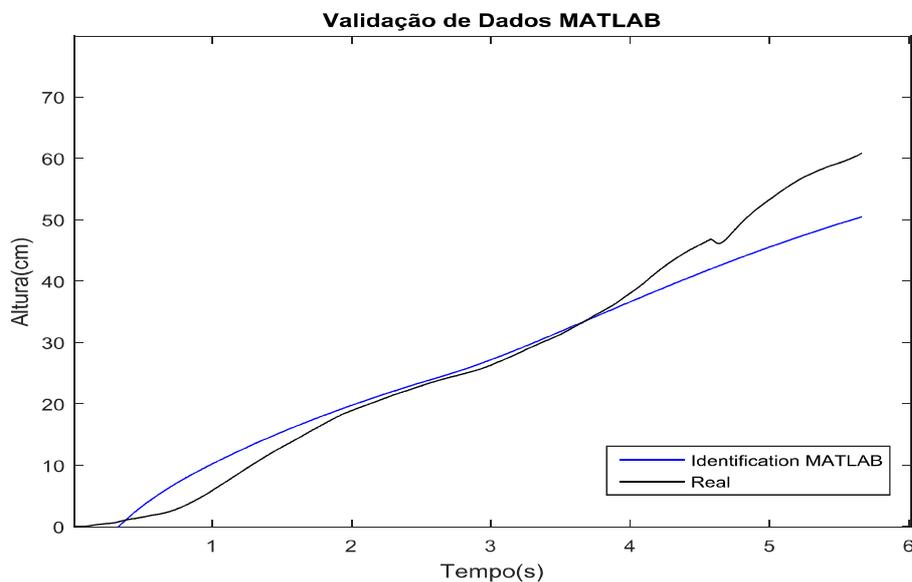
Nesta seção os dados de validação serão comparados com a ferramenta desenvolvida pelo MATLAB a “identification”. Aqui será somente comparada a validação gerada pelo algoritmo ARX e a validação gerada pelo MATLAB configurada também para utilizar o modelo ARX fazendo uma análise de desempenho do algoritmo de identificação criado neste trabalho.

A configuração da ferramenta do MATLAB foi idêntica à tabela 4.2 para que possa ser realizada a comparação de forma satisfatória. A função de transferência gerada pelo MATLAB é dada por:

$$S(z) = \frac{0,001292}{1 - 1,992z^{-1} + 0,9924z^{-2}} \quad (36)$$

É possível perceber que há diferenças entre os coeficientes do numerador e denominador da função de transferência $S(z)$ e a Equação (32) gerada pelo Algoritmo ARX a $G(z)$. Isto ocorre pelo fato que a ferramenta “identification” é mais avançada utilizando outros processos para gerar maior aproximação da função $S(z)$ com os dados de validação em comparação com algoritmo $G(z)$. A Figura 4.9 mostra o gráfico de validação da função de $S(z)$.

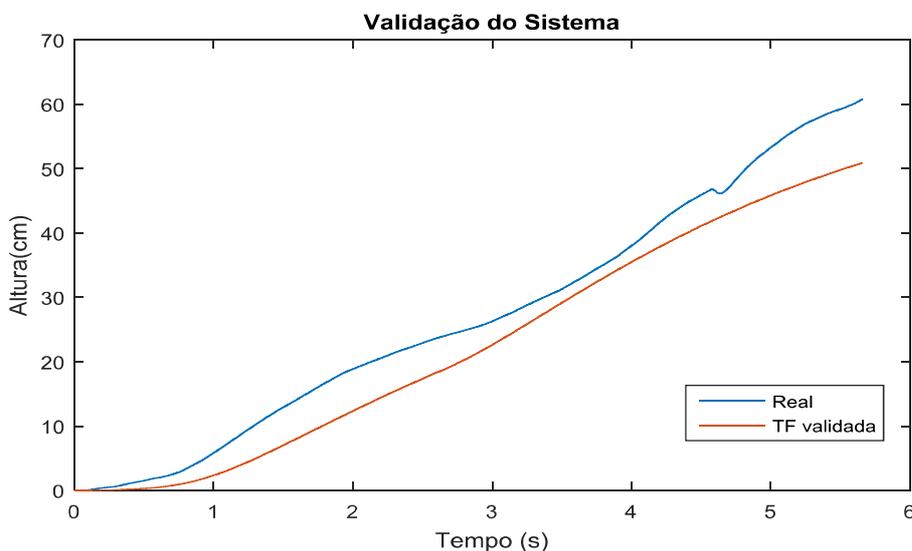
Figura 4.9 - Validação de dados com a ferramenta IDENTIFICATION.



Fonte: Daniel Trovão.

Na Figura 4.9, a linha em preto representa os dados de validação filtrados e a linha azul representa o gráfico gerado pela função de transferência $S(z)$.

Figura 4.10 - Validação de dados com Algoritmo ARX.



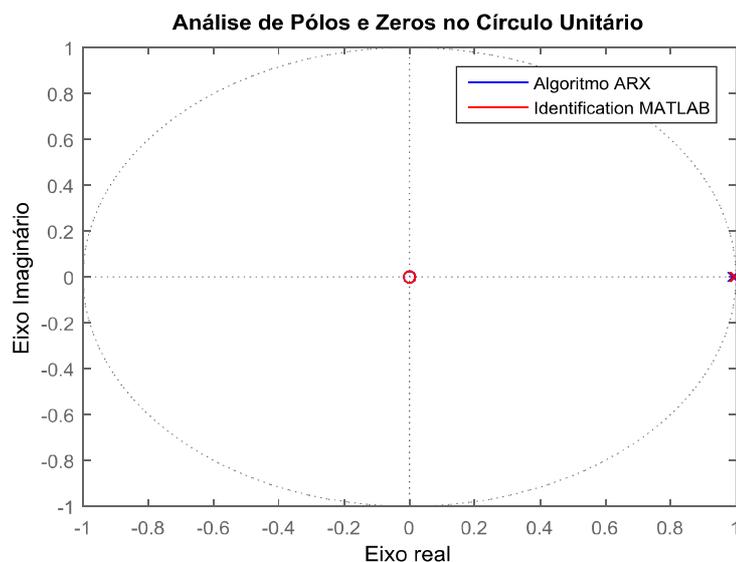
Fonte: Daniel Trovão.

Analisando as duas figuras acima a função de transferência gerada pela Identification possui uma precisão um pouco maior e uma resposta mais rápida. Porém, entre o tempo de 0 a 1 segundo a Figura 4.9 apresenta sinal negativo gerando uma discrepância o que não é vista na figura 4.10. No intervalo de 2 a 4 segundos, a função $S(z)$ se aproxima mais com os dados

de validação do que a $G(z)$; contudo, acima de 4 segundos ambas as funções se comportam de forma semelhante.

Outro tipo de análise realizada foi o gráfico de polos e zeros de ambas as funções de transferência.

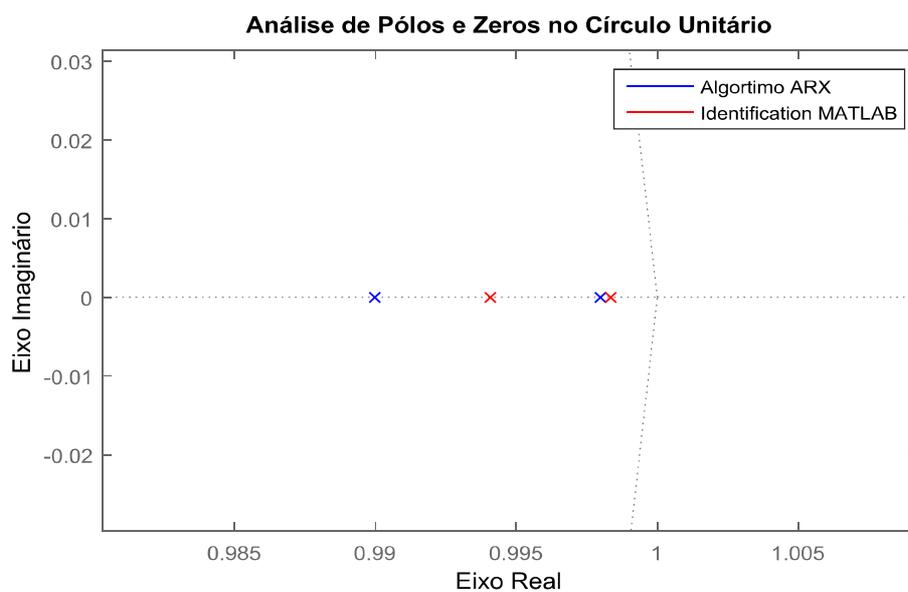
Figura 4.11 - Gráfico de polos e zeros no círculo unitário.



Fonte: Daniel Trovão.

Na Figura 4.11 observa-se que os polos tanto da função $G(z)$ e $S(z)$ são muito próximos da região do círculo unitário como pode ser visto na Figura 4.12 que é uma versão ampliada da figura anterior.

Figura 4.12 - Gráfico de polos e zeros no círculo unitário Ampliado.



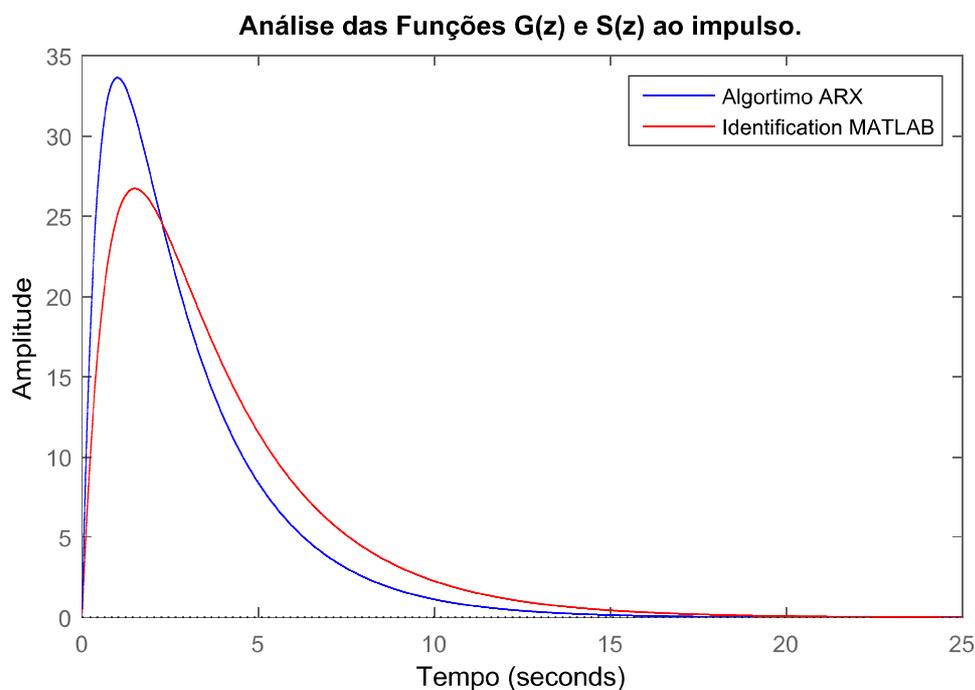
Fonte: Daniel Trovão.

Contudo, quando ampliada, a função identificada pelo algoritmo ARX em azul está mais afastada da região unitária que a `toolbox` do MATLAB em vermelho. Ambos os sistemas são marginalmente estáveis.

As próximas análises realizadas foram às respostas ao impulso e degrau na Figura 4.13, o algoritmo ARX apresentou um pico maior ao impulso o que pode significar um esforço maior para o controlador em atenuar este tipo de sinal. Entretanto, o sinal apresentou uma rápida queda em relação à função $S(z)$.

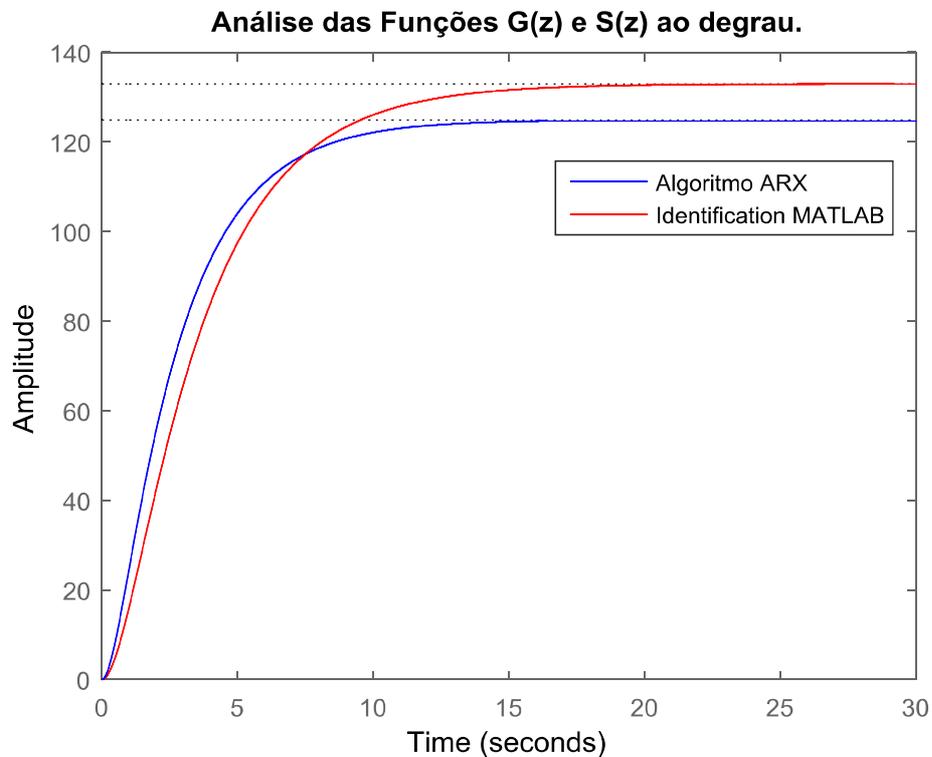
Portanto, ambas as funções podem ser usadas para controle ficando a critério do projetista a escolha da função que melhor lhe convém.

Figura 4.13 - Análise de $G(z)$ e $S(z)$ ao impulso.



Fonte: Daniel Trovão.

Para a análise em degrau, a Figura 4.14 mostra que a função $G(z)$ apresenta uma pequena rapidez comparada a função identificada no MATLAB, $S(z)$.

Figura 4.14 - Análise de $G(z)$ e $S(z)$ ao degrau.

Fonte: Daniel Trovão.

Portanto, ambas as funções podem ser utilizada para avaliação e controle, pois ambas possuem características muito semelhantes. Desta forma, dada à complexidade da ferramenta Identification do MATLAB, o algoritmo ARX pode ser utilizado para identificar o sistema o menor custo computacional e de forma mais simples.

5 CONCLUSÃO

A monografia foi desenvolvida com o intuito de conseguir realizar uma identificação do quadrimotor através da construção de uma placa de aquisição de dados com o microcontrolador Arduino. Depois de muitas tentativas e erros, o objetivo do trabalho foi alcançado.

O que pode ser ressaltado como ponto positivo nesta monografia foi às diversas soluções encontradas para lidar com os problemas. O que englobou diversos conhecimentos da graduação como: Eletrônica, Linguagem de Programação, Processos Estocásticos, Probabilidade e Controle.

Adicionalmente, conhecimentos mais aprofundados em controle tiveram que ser explorados como: Filtros de Kalman e Identificação de Sistemas que são somente vistas no mestrado.

Outro desafio foi como realizar a transmissão e lidar com os defeitos nos dados, exemplos: ruídos, atenuações, perdas, discrepâncias como também o controle da taxa de amostragem.

Portanto, esta monografia veio a contribuir como consolidador do conhecimento e também uma porta para trabalhos de outros estudantes que desejam seguir na área de controle e automação focados nos processos de modelagem matemática utilizando técnicas de identificação.

- Trabalhos futuros.

Muitos desafios ainda ficaram em aberto para serem alcançados exemplo: realizar a identificação completa do quadricóptero. Porque este sistema é MIMO (múltiplas entradas e múltiplas saídas) com característica de subatuação, pois este sistema possui somente quatro entradas: Empuxo, Arfagem, Rolagem e Guinada para seis saídas que são: os movimentos translacionais nos eixos X,Y,Z e movimentos rotacionais Arfagem, Rolagem e Guinada.

Outra proposta de trabalho futuro é lidar com a não linearidade do modelo, pois este modelo é altamente não linear quando as variações de entrada saem de uma determinada taxa. Portanto, realizar uma identificação não linear no modelo pode dá mais precisão a dinâmica do sistema.

Um trabalho futuro que pode ter bastante impacto no ramo acadêmico quanto profissional é o desenvolvimento de um controlador tanto para manter a estabilidade de voo

quanto para designar trajetórias a serem percorridas. Com relação à estabilidade de voo pode ser desenvolvido um trabalho na construção de um controlador digital não linear utilizando múltiplas taxas de amostragem ou desenvolver controladores adaptativos.

Em conclusão, este trabalho pode gerar vários artigos, dissertações e até teses se for bem explorado.

REFERÊNCIAS BIBLIOGRÁFICAS

AGUIRRE, L. A. *Introdução à identificação de sistemas*. 2º. ed. Belo Horizonte: UFMG, 2004.

AMAZON, C. Amazon: Syma X6. Amazon, 2017. Disponível em: <<https://www.amazon.com/Syma-Super-Channel-2-4GHz-Copter/dp/B00JKV88K4>>. Acesso em: 17 jun. 2017.

ASTRÖM, K.; WITTENMARK, B. *Adaptive Control*. New York: Dover, 2008.

BAPTISTA, M. *Sistema de Aquisição de dados*. Escola Superior de Tecnologia de Viseu. Viseu, p. 30. 2008.

BOUABDALLAH, S. *Design and Control of Quadrotors with application to autonomous flying*. ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE. Lausanne, p. 116. 2007.

BRESCIANI, T. *Modelling, Identification and Control of a Quadrotor Helicopter*. Lund University. Lund, Sweden, p. 184. 2008.

CORP, M. Microchip, 1998. Disponível em: <<http://www.microchip.com/wwwproducts/en/ATmega328P>>. Acesso em: 08 maio 2017.

DA SILVA, A. J. *Identificação de Sistemas: Teoria e prática em modelo polinomiais*. 1ª. ed. São Luís: EDUFMA, v. único, 2015.

DE CASTRO JORGE, L. A.; INAMASU, R. Y. *Uso de veículos aéreos não tripulados (VANT) em Agricultura de Precisão*. In: DE CAMPOS BERNARDI, A. C., et al. *Agricultura de precisão: um novo olhar*. São Carlos: Empraba, 2014. p. 600.

DE CASTRO RIBEIRO RESENDE, D. *Drones e Vant's: Conceitos Básicos e seu uso na engenharia de avaliações de perícias*., Fevereiro 2015.

EIRELI, F. C. E. FelipeFlop. *FilipeFlop*, 2017. Disponível em: <<http://www.filipeflop.com/pd-10fd9f-acelerometro-e-giroscopio-3-eixos-6-dof-mpu-6050.html>>. Acesso em: 09 maio 2017.

EMILIO DI PAOLO, M. *Data Acquisition System: from fundamental to Applied Design*. 1ª. ed. New York: Springer, 2013.

FELIPEFLOP. *FelipeFlop*, 2017. Disponível em: <<http://blog.filipeflop.com/wireless/modulo-rf-apc220-arduino.html>>. Acesso em: 09 maio 2017.

FELIPEFLOP, C. Módulo Cartão SD Card. FELIPEFLOP, 2016. Disponível em: <<http://www.filipeflop.com/pd-6b847-modulo-cartao-sd-card.html?ct=&p=1&s=1>>. Acesso em: 17 jun. 2017.

FURTADO, V. H.; VEIGA GIMENES, R. A.; CAMARGO JUNIOR, J. B. Aspectos de segurança na integração de veículos aéreos não tripulados (VANT) no espaço aéreo brasileiro. *Sitraer* 7, Rio de Janeiro, 26 à 28 novembro 2008. 506-517.

H. ROSEMARY, T. *Data Aquisition for sensors*. 1ª. ed. Devon: Chapman & Hall, 1997.

KEESMAN, J. K. *System Identification: An Introduction*. London: Springer, 2011.

LJUNG, L. *System Identification, Theory for the user*. 2ª. ed. New Jersey: Prentice Hall, 1999.

LUUKKONEN, T. *Modelling and control of quadcopter*. Aalto University. Espoo, p. 26. 2011.

MATHWORKS, I. MathWorks - Makers of MATLAB and Simulink. Mathwors, 2016. Disponível em: <<https://www.mathworks.com/>>. Acesso em: 17 jun. 2016.

MENDEL, J. M. *Lessons in Digital Estimation Theory*. New Jersey: Prentice Hall, 1987.

NATIONAL INSTRUMENTS, C. myDAQ - myDAQ - Student Data Acquisition Device Manual - National Instruments. National Instruments, 2017. Disponível em: <<http://www.ni.com/documentation/en/mydaq-student-data-acquisition-device/latest/mydaq/overview/>>. Acesso em: 17 jun. 2017.

NEWCOME, L. R. *Unmanned Aviation: A brief history of unmanned aerial vehicles*. Virginia.: American Institute of Aeronautics and Astronautics., 2004.

OLIVEIRA MARTINS, R. M. **Identification and Validation of a Quadrotor's Model Dynamics**. Instituto Superior Tecnico. Lisboa, p. 9. 2014.

PARK, J.; MACKAY, S. *Practical Data Acquisition for instrumentation and Control Systems*. 1ª. ed. Burlington: ELSEVIER, 2003.

RESENDIZ, V. M. A.; ARAIZA, E. R. *System Identification of a Quad-rotor in X Configuration from Experimental Data*. Universidad Autonoma de Queretaro. Queretaro, Mexico, p. 10. 2016. (ISSN 1870-4069).

RIBEIRO RESENDE, D. D. C.; IVES ROQUE PACHECO, N. *Controle de estabilidade de aeromodelo tipo quadcopter autômato por lei de controle PID*. PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS. Belo Horizonte, p. 88. 2014.

RICH, M. Model development, system identification, and control of a quadrotor helicopter. Iowa State University. Ames, p. 144. 2012.

SALAMEH, I. M.; AMMAR, E. M.; TUTUNJI, T. A. Identification of Quadcopter Hovering Using Experimental Data. 2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies, Amman, 15 março 2015. 6.

SANTOS TORRES, L. H. MODELAGEM, IDENTIFICAÇÃO E CONTROLE ADAPTATIVO DE SISTEMA DE BOMBEIO MECÂNICO PARA POÇOS DE PETRÓLEO. Universidade Federal da Bahia. Salvador, p. 142. 2012.

SARıOđLU, A.; KURAL, A. Modeling and ARX Identification of a Quadrotor MiniUAV. Istanbul Technical University. Istanbul , p. 6. 2015.

SODERSTROM, T.; STOICA, P. System Identification. London: Prentice Hall, 1989.

STANCULEANU, I.; BORANGIU, T. Quadrotor Black-Box System Identification. International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering, v. 5, 2011. ISSN 6.

VALAVANIS, K. P.; VACHTSEVANOS, G. J. Handbook of Unmanned Aerial Vehicles. New York: SpringerReference, v. I, 2014.

ANEXOS

AQUISIÇÃO DA TENSÃO DO MANCHE E ALTURA DO SYMA X6

```

//Nome:
//Criado por Daniel Trovao Simoes
//Data 16/12/2016

//1.Bibliotecas.-----//-----//-----

#include <Ultrasonic.h>
//2.Inicializacao de variaveis.-----//-----

#define pino_trigger 8
#define pino_echo 7
Ultrasonic ultrasonic(pino_trigger, pino_echo);
float MedidaZ=0;
float ControleManche=0;
double timer;

void setup() {
  Serial.begin(57600);
}
void loop() {
  // Tempo de amostragem.
  timer = millis();
  long microsec = ultrasonic.timing();

  // Leitura de Altura do sonar em cm.
  MedidaZ = ultrasonic.convert(microsec, Ultrasonic::CM);

  // Leitura da tensao do mache em volts.
  float ControleManche = analogRead(A0)*2*0.005;

  // Enviando Dados pela porta Serial.
  Serial.print(timer);
  Serial.print(' ');
  Serial.print(',');
  Serial.print(MedidaZ);
  Serial.print(' ');
  Serial.print(',');
  Serial.print(' ');
  Serial.println(ControleManche);
}

```

ANÁLISE DA COLETA DE DADOS

```

% -----
%                               Análise da Coleta de Dados
% -----
%                               Drone
% -----
%   Desenvolvidor: Daniel Trovão, Evandro Martins
%   Supervisão: Prof. João Viana
%   Data: 23/06/2017
% -----
%                               Entrada de dados
%                               COLETA DE DADOS
% -----
%                               Analise do Dados
% -----
TamanhoU=length(Entradaident);
TamanhoY=length(Saidaident);
if(TamanhoY==TamanhoU)
    disp('Tamanho de amostras igual!');
    N=TamanhoY;
else
    disp('Erro: Tamanho nao batem!');
end
% -----
%                               Analise Temporal
% -----

for i=2:length(milis)
    dt(i)=milis(i)-milis(i-1);
end
tabulate(dt)
analiseDT=tabulate(dt); % analise as frequencias de intervalos de tempo
j=1;
porcent=input('O valor minimo de % para ser reconhecido como amostragem:');
for i=1:length(analiseDT(:,1))
    if (analiseDT(i,3)>porcent) %quando a frequencia do intervalor for >
porcent%
        valoresDT(j)=analiseDT(i,1); %guarda no valoresDT os intervalos >1%
        j=j+1;
    end
end
for i=1:length(Saidaident);
    if (min(valoresDT) <= dt(i) <= max(valoresDT)) % dt(i) estiver entre o
min e max dos valoresDT
        Saidaident(i)=Saidaident(i);           %Atribui Saida a propria saida
        Entradaident(i)=Entradaident(i);       %Atribui Entrada a propria
    entrada
    else
        Saidaident(i)=Saidaident(i-1);         % se nao atribui a anterior
        Entradaident(i)=Entradaident(i-1);
    end
end
disp('O intervalo de amostragem eh (S):');
ts=mean(valoresDT)/1000;
% -----
%                               SAIDA e ENTRADA GRAFICOS
% -----
figure(1)
subplot(2,1,1)
plot(Saidaident, '-b');

```

```

legend('Saida')
title('Saida do Sistema')
xlabel('Amostra(n)')
ylabel('Altura(cm)')
subplot(2,1,2)
plot(Entradaident, '-b');
title('Entrada do Sistema')
legend('Entrada')
xlabel('Amostra(n)')
ylabel('Volts(V)')

%valor da maximo da altura da porta 220cm
%valor da maximo da tensao manche 3.3v
MaxEntrada=input('Entre com valor maximo de Entrada:');
MaxSaida=input('Entre com valor maximo de saida:');
for i=1:N
    if isnan(Saidaident(1))
        Saidaident(1)=0;
    end
    if isnan(Entradaident(1))
        Entradaident(1)=0;
    end
    if (isnan(Saidaident(i)) || (Saidaident(i)>MaxSaida))
        Saidaident(i)= Saidaident(i-1);
    end
    if (isnan(Entradaident(i)) || (Entradaident(i)>MaxEntrada))
        Entradaident(i)= Entradaident(i-1);
    end
end
end
%-----
%                               SAIDA e ENTRADA GRAFICOS
%-----
figure(1)
subplot(2,1,1)
plot(Saidaident, '-b');
legend('Saida')
title('Saida do Sistema')
xlabel('Amostra(n)')
ylabel('Altura(cm)')
subplot(2,1,2)
plot(Entradaident, '-b');
title('Entrada do Sistema')
legend('Entrada')
xlabel('Amostra(n)')
ylabel('Volts(V)')
disp(' ');

Corte=input('Deseja fazer algum corte nos dados? (Y/N):','s');
if Corte=='Y'
    Min=input('Voce desejou cortar os dados. Digite o intervalo menor:');
    Max=input('Voce desejou cortar os dados. Digite o intervalo Maior:');
    Saidaident=Saidaident(Min:Max);
    Entradaident=Entradaident(Min:Max);
    figure(1)
    subplot(2,1,1)
    plot(Saidaident, '-b');
    legend('Saida')
    title('Saida do Sistema')
    xlabel('Amostra(n)')
    ylabel('Altura(cm)')

```

```
subplot(2,1,2)
plot(Entradaident, '-b');
title('Entrada do Sistema')
legend('Entrada')
xlabel('Amostra(n)')
ylabel('Volts(V)')
disp(' ');
end
```

FILTRAGEM DE DADOS

```

% -----
%                               Filtragem de Dados
% -----
%                               Drone
% -----
%       Desenvolvidor: Daniel Trovão, Evandro Martins
%       Supervisão: Prof. João Viana
%       Data: 24/05/2017
% -----
% ----- Load Data -----
% ----- Time Constant -----
Tf=length(Saidaident);
% ----- System Data Load -----
Y=Saidaident;U=Entradaident;
% ----- Initial Conditions -----
Uetest='Y';
while(Uetest=='Y')
KALMAN=input('Deseja ajustar o sinal de entrada do filtro de Kalman
(Y/N):','s');
disp(' ');
disp('Relacao dos Valores Preconfigurados');
disp('Variancia calculada:      R = 0.00075');
disp('Variancia do processo:    Q = 0.1e-3; ');
disp('Covariancia P inicial:    P = 1e-7;');
disp('Ganho de Kalman Inicial:   K = 0;');
disp('Estado estimado inicial:  Ue = 0;');
disp(' ');
if KALMAN=='Y'
R = input('Variancia calculada      R:');           % Variancia
calculada;
Q = input('Variancia do processo    Q:');           % Variancia do
processo;
P = input('Covariancia inicial     P:');           % Covariancia P
inicial;
K = input('Ganho de Kalman Inicial K:');           % Ganho de Kalman
Inicial.
Ue = input('Estado estimado inicial Ue');           % Estado estimado
inicial.
end
if KALMAN=='N'
R = 0.00075;           % Variancia calculada;
Q = 0.1e-3;           % Variancia do processo;
P = 1e-7;             % Covariancia P inicial;
K = 0;                % Ganho de Kalman Inicial.
Ue = 0;               % Estado estimado.
end
% ----- Kalman Filtering -----
for k=2:Tf
K(k) = (P(k-1) + Q)/((P(k-1) + Q) + R); % Ganho de Kalman.
P(k) = (1 - K(k))*((P(k-1) + Q));      % Calculo da covariancia.
Ue(k) = Ue(k-1) + K(k)*(U(k)-Ue(k-1)); % Estado estimado pelo filtro.
end
% ----- Plot Area -----
%SAIDA e ENTRADA ESTIMADAS
figure(1)
plot(U,'Color',[0 0 0.7] );
hold on
title('Entrada do Sistema')

```

```

legend('Entrada')
plot(Ue, '-r');
legend('Medido', 'Estimado')

%COVARIANCIA DE KALMAN ENTRADA E SAIDA
figure(2)
plot(K, '-b')
title('Ganho de Kalman em Ue')

%COVNARIANCIA DE KALMAN ENTRADA E SAIDA
figure(3)
plot(P, '-b')
title('Covariancia de Kalman Ue')

Uetest=input('Voce deseja ainda alterar a filtragem dos dados de entrada
?(Y/N):', 's');
end

Yetest='Y';
while(Yetest=='Y')
KALMAN=input('Deseja ajustar o sinal de saida do filtro de Kalman
(Y/N):', 's');
disp(' ');
disp('Relacao dos Valores Preconfigurados');
disp('Variancia calculada:      R1 = 0.00075');
disp('Variancia do processo:    Q1 = 0.1e-3; ');
disp('Covariancia P inicial:    P1 = 1e-7;');
disp('Ganho de Kalman Inicial:  K1 = 0;');
disp('Estado estimado inicial:  Yel= 0;');
disp(' ');
if KALMAN=='Y'
R1 = input('Variancia calculada      R1:');           % Variancia calculada;
Q1 = input('Variancia do processo    Q1:');           % Variancia do processo;
P1 = input('Covariancia inicial      P1:');           % Covariancia P inicial;
K1 = input('Ganho de Kalman Inicial  K1:');           % Ganho de Kalman
Inicial.
Ye= input('Estado estimado inicial  Ye:');           % Estado estimado
inicial.
end
if KALMAN=='N'
R1 = 0.00075;           % Variancia calculada;
Q1 = 0.1e-4;           % Variancia do processo;
P1 = 1e-7;             % Covariancia P inicial;
K1 = 0;                % Ganho de Kalman Inicial.
Ye = 0;                % Estado estimado.
end
% ----- Initial Conditions ----- %
for k=2:Tf
K1(k) = (P1(k-1) + Q1)/((P1(k-1) + Q1) + R1);   % Ganho de Kalman.
P1(k) = (1 - K1(k))*((P1(k-1) + Q1));           % Calculo da
covariancia.
Ye(k) = Ye(k-1) + K1(k)*(Y(k)-Ye(k-1));         % Estado estimado pelo
filtro.
end
% ----- Plot Area ----- %
%SAIDA e ENTRADA ESTIMADAS
figure(4)
plot(Y, 'Color', [0 0 0.7] );
hold on
title('Saida do Sistema')
legend('Saida')

```

```
plot(Ye, '-r');  
legend('Medido', 'Estimado')  
  
%COVARIANCIA DE KALMAN ENTRADA E SAIDA  
figure(5)  
plot(K1, '-b')  
title('Ganho de Kalman em Ye')  
  
%COVNARIANCIA DE KALMAN ENTRADA E SAIDA  
figure(6)  
plot(P1, '-b')  
title('Covariancia de Kalman Ye')  
  
Yetest=input('Voce deseja ainda alterar a filtragem dos dados de Saida  
?(Y/N):', 's');  
end
```

IDENTIFICAÇÃO ARX

```

% -----
%                               Identification of ARX Model
% -----
%                               Drone
% -----
%   Desenvolvidor: Daniel Trovão, Evandro Martins
%   Supervisão: Prof. João Viana
%   Data: 17/05/2017
% -----
% ----- Load Data -----
y=Ye;                               %Saida
U=Ue;                               %Entrada
Ts=input('Entre com o intervalo de amostragem em segundos:');%amostragem.
NP=input('Entre com o numero de polos:'); %Coeficientes de A
NZ=input('Entre com o numero de zeros:'); %Coeficientes de B
%----- Initial Conditions -----
NZ=NZ+1; % Numero de zeros na TF
teta=zeros(NP+NZ,NP); % Parametros Iniciais
K=teta;
P=100*eye(NP+NZ); % Matriz de Covariancia inicial
Y=-y; %Saida p/ vetor de regressores
% ----- Modelo ARX -----
%Analise do Tamanho de Dados.
if (NZ>NP)
    disp('Erro:Numero Zeros maior que o de polos!');
end
TamanhoU=length(U);
TamanhoY=length(Y);
if(TamanhoY==TamanhoU)
    disp('Tamanho de amostras igual!');
    N=TamanhoY;
else
    disp('Erro: Tamanho nao batem!');
end

for t=NP+1:N
    %(OK)Vetor de Regressores phi=[-y(t-1) ... -y(t-Np) U(t-1) ... U(t-Nz)];
    for k=1:NP
        phi(k,1)=Y(t-k);
    end
    for l=1:NZ
        phi(NP+1,1)=U(t-1);
    end

    %Algoritmo de Estimacao.
    r(t) = 0; %gerador de ruido
    y_est(t) = phi'*teta(:,t-1) + r(t); %saida estimada
    erEst(t) = y(t) - y_est(t); %Erro de estimacao
    K(:,t) = (P * phi)/(1 + (phi'*P*phi)); %Ganho do estimador
    teta(:,t) = teta(:,t-1) + K(:,t)*erEst(t); %Parametros Estimados
    P = (P-K(:,t)*phi'*P); %Calculo da covariancia.
end
% ----- Transfer Function Identified -----
num = [teta(NP+1:NP+NZ,N)'];
den = [1 teta(1:NP,N)'];
G = tf(num,den,Ts,'variable','z^-1') % Função tranferencia
t=0:Ts:N*Ts-Ts;
dd=lsim(G,U,t);
%----- GRAFICOS DE CONVERGENCIA -----

```

```

%Valores de Entrada e Saida Identificacao
figure(1)
subplot(2,1,1)
plot(Ye, '-r');
legend('Medida')
title('Saida do Sistema')
ylabel('Altura(cm)')
xlabel('Tempo (s)')
subplot(2,1,2)
hold on
title('Entrada do Sistema')
ylabel('Volts(V)')
xlabel('Tempo (s)')
plot(Ue, '-r');
legend('Medida')

%Analise do PARAMETROS.
figure(3)
stairs(t,teta(1,:))
hold on
for i=1:(NP+NZ)
stairs(t,teta(i,:))
end
title('Convergencia do Parametros Identificados')
ylabel('Parametros')
xlabel('Tempo (s)')
hold off

%ANALISE do Y estimado e Y real
figure(4)
plot(t,y_est)
hold on
plot(t,y)
plot(t,dd)
legend('Estimado','Real')
title('Identificação do Sistema')
ylabel('Altura(cm)')
xlabel('Tempo (s)')
hold off

%Analise do Erro de Estimacao
figure(5)
plot(t,erEst)
title('Erro de Predicao')
ylabel('Erro')
xlabel('Tempo (s)')

erValidacao=Ye-dd';
figure(6)
plot(t,erValidacao)
title('Erro da funcao de transferencia de identificacao')
ylabel('Erro')
xlabel('Tempo (s)')

```

VALIDAÇÃO DO SISTEMA

```

% ----- Validação de sistema -----
%                                     %
%                                     %
%                                     %
%                                     %
%                                     %
%                                     %
%                                     %
% ----- Validacao -----
%                                     %
%                                     %
%                                     %
%                                     %
%                                     %
%                                     %
% ----- Validação do Modelo -----
%                                     %
TamanhoU=length(Ue);
TamanhoY=length(Ye);
if(TamanhoY==TamanhoU)
    disp('Tamanho de amostras igual!');
    N=TamanhoY;
else
    disp('Erro: Tamanho nao batem!');
end
t=0:Ts:N*Ts-Ts;
dd2=lsim(G,Ue,t);

figure(1)
subplot(2,1,1)
stairs(t,Ue)
title('Entrada Aplicada ao Sistema')
ylabel('Tensão (V)')
xlabel('Tempo (s)')
subplot(2,1,2)
stairs(t,Ye)
hold on
stairs(t,dd2);
legend('Real','TF validada')
title('Validação do Sistema')
ylabel('Altura(cm)')
xlabel('Tempo (s)')

```