

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
ENG. DE CONTROLE E AUTOMAÇÃO

NICKOLAS PESSIM OLIVEIRA DA
FONSECA - 00302582

**PROJETO DE CONTROLADOR
PARA QUADRICÓPTEROS EM
ENTREGAS AUTÔNOMAS**

Porto Alegre
2024

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
ENG. DE CONTROLE E AUTOMAÇÃO

NICKOLAS PESSIM OLIVEIRA DA
FONSECA - 00302582

**PROJETO DE CONTROLADOR
PARA QUADRICÓPTEROS EM
ENTREGAS AUTÔNOMAS**

Trabalho de Conclusão de Curso (TCC-CCA)
apresentado à COMGRAD-CCA da Universi-
dade Federal do Rio Grande do Sul como parte
dos requisitos para a obtenção do título de *Ba-
charel em Eng. de Controle e Automação* .

ORIENTADOR:

Prof. Dr. Mário R. Sobczyk Sobrinho

Porto Alegre
2024

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
ENG. DE CONTROLE E AUTOMAÇÃO

NICKOLAS PESSIM OLIVEIRA DA
FONSECA - 00302582

**PROJETO DE CONTROLADOR
PARA QUADRICÓPTEROS EM
ENTREGAS AUTÔNOMAS**

Este Trabalho de Conclusão de Curso foi julgado adequado para a obtenção dos créditos da Disciplina de TCC do curso *Eng. de Controle e Automação* e aprovado em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: _____
Prof. Dr. Mário R. Sobczyk Sobrinho, UFRGS
Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Banca Examinadora:

Prof. Dr. Mário R. Sobczyk Sobrinho, UFRGS
Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Prof. Dr. Eduardo André Perondi, UFRGS
Doutor pela Universidade Federal de Santa Catarina – Florianópolis, Brasil

Prof. Dr. Walter Fetter Lages, UFRGS
Doutor pelo Instituto Tecnológico de Aeronáutica – São José dos Campos, Brasil

Alceu Heinke Frigeri
Coordenador de Curso
Eng. de Controle e Automação

Porto Alegre, Agosto 2024

AGRADECIMENTOS

Aos meus pais, Patrício e Eliane, e minha irmã Nicolle pelo apoio incondicional e sustentação emocional proporcionados durante todo o processo de graduação, mesmo apesar da distância, amo vocês.

A toda minha família incluindo tios, primos, avós, e em especial Luciano e Bruno, que também passaram pela experiência de se formar no estado gaúcho e foram grande fonte de companhia ao longo dos anos.

A todos os meus amigos e companheiros de curso, que formaram um verdadeiro grupo de apoio, proporcionando diversão e alívio em momentos difíceis.

Ao meu orientador e professor Mário pela alta preocupação e dedicação ao auxílio neste trabalho e na minha formação.

A todos os meus colegas de bolsa de pesquisa e de estágio, que proporcionaram um conhecimento externo e adicional ao ensinado pela universidade.

Essa conquista não seria possível sem vocês, muito obrigado.

RESUMO

No âmbito deste trabalho, busca-se o desenvolvimento de um sistema embarcado destinado a quadricópteros, visando a controlar sua dinâmica com base em trajetórias predefinidas para operações autônomas de entrega. O sistema é projetado empregando malhas de controle digital para controlar os estados que influenciam os motores da aeronave não tripulada (UAV, *unmanned aerial vehicle*), utilizando as medições provenientes de uma unidade de medição inercial (IMU, *inertial measurement unit*), um sistema de posicionamento global (GPS, *global positioning system*), um barômetro, um magnetômetro e sensores de distância. Para o controle, são avaliadas duas abordagens distintas, uma baseada em um controlador PID e outra baseada em uma estratégia de linearização via realimentação de estados. Também é desenvolvido um algoritmo de cálculo de trajetórias de referência para os voos autônomos com uma análise de seu desempenho. Ambas as técnicas são implementadas em um ambiente de simulação desenvolvido pelo autor, programado na linguagem C++. Os resultados mostram que ambas as técnicas fornecem um comportamento considerado adequado para a aplicação, tanto para a lei de controle quanto para o algoritmo de geração de trajetórias.

Palavras-chave: Projeto, UAV, Entrega Autônoma, Automação e Controle, Dinâmica de voo, Quadricópteros, Linearização por realimentação de estados.

ABSTRACT

This work aims to develop an embedded system for quadcopters, designed to control their dynamics based on predefined trajectories for autonomous delivery operations. The system is developed using digital control loops to control the states influencing the unmanned aerial vehicle (UAV) motors, utilizing measurements from an inertial measurement unit (IMU), a global positioning system (GPS), a barometer, a magnetometer, and distance sensors. Two distinct control approaches are evaluated: one based on a PID controller and the other on a state feedback linearization strategy. Additionally, a reference trajectory calculation algorithm for autonomous flights is developed, with a performance analysis. Both techniques are implemented in a simulation environment developed by the author, programmed in C++. The results demonstrate that both techniques provide satisfactory behavior for the application, for both the control law and the trajectory generation algorithm.

Keywords: Project, UAV, Autonomous Delivery, Automation and Control, Flight Dynamics, Quadcopters, Feedback linearization.

LISTA DE ILUSTRAÇÕES

1	Quadricópteros em diferentes configurações.	15
2	Referenciais inercial e não-inercial indicados respectivamente por i e η	18
3	Diagrama de corpo livre da seção um quadricóptero em diferentes sistemas de coordenadas.	19
4	Diagrama de corpo livre da atitude de um quadricóptero no referencial não-inercial.	20
5	Diagrama de corpo livre da hélice do rotor com v_{rv} representando a velocidade relativa com relação ao ar, T a força propulsora, S a força de sustentação e D a componente de arrasto.	22
6	Diagrama de blocos do sistema que evidencia as entradas e saídas de cada bloco com unidades no SI.	24
7	Diagrama de blocos do sinal de controle para o quadricóptero com bloco de conversão ε especificado.	28
8	Lugar geométrico das raízes do sistema realimentado com um controlador PID, PI, PD e P.	28
9	Diagrama de blocos da malha fechada do quadricóptero com dois controladores PID em cascata, C_o para a odometria e C_v para a velocidade.	29
10	Lugar geométrico das raízes para o laço de velocidade com um controlador PID, PI, PD e P.	29
11	Diagrama de blocos da linearização por realimentação de estados para um sistema na forma canônica controlável.	31
12	Diagrama de blocos da relação entre o sinal aplicado e o sinal aerodinâmico.	32
13	Diagrama de blocos da relação entre o sinal aplicado e o sinal aerodinâmico.	32
14	Orientação da guinada para seguimento da rota planejada.	35
15	Distância entre o quadricóptero e a rota planejada para um desvio anômalo.	36
16	Máquina de estados para o algoritmo de entregas autônomas.	39
17	Ensaio de modelagem para variações de estado independentes.	41
18	Histograma dos erros relativos do modelo proposto com relação ao modelo de referência.	42
19	Ensaio de controle para controlador PD em cascata.	43
20	Ensaio de controle para controlador de linearização por realimentação de estados.	44
21	Relação do tempo de acomodação teórico com o tempo de acomodação simulado.	45

22	Posição do quadricóptero para o ensaio do algoritmo de voo.....	46
23	Orientação do quadricóptero para o ensaio do algoritmo de voo.....	47
24	Diagrama UML para a código desenvolvido.....	54
25	Diagrama de fluxo de dados para a código desenvolvido.	55
26	Visualização gráfica da simulação do quadricóptero no ambiente <i>Unreal Engine</i>	56

LISTA DE TABELAS

1	Simbologia da atitude angular do sistema	16
2	Simbologia da atitude linear do sistema.	16
3	Ganhos para os controladores do tipo PD unidimensionais equivalentes em cascata.	30
4	Ganhos para os controladores unidimensionais genéricos em cascata. . .	33
5	Referências do controlador para a decolagem.	35
6	Referências do controlador para o transporte.	36
7	Referências do controlador para a prevenção de colisões.	37
8	Referências do controlador para o pouso.	38
9	Referências do controlador para todos os estados.	38
10	Parâmetros do modelo adotados para a simulação de dados.	40
11	Degraus de tensão aplicados aos ensaios de modelagem independentes. .	41
12	Parâmetros para o controlador PD em cascata.	42
13	Parâmetros para o controlador linear em cascata da linearização por realimentação de estados.	44

LISTA DE ABREVIATURAS

UAV	Unmanned Aerial Vehicle (Veículo Aéreo Não-tripulado)
IMU	Inertial Measurement Unit (Unidade de Mensuração Inercial)
GPS	Global Positioning System (Sistema de Posicionamento Global)
VTOL	Vertical Take-Off and Landing (Decolagem e Aterrisagem Verticais)
PID	Controlador proporcional, integral e derivativo
LRE	Linearização por Realimentação de Estados

LISTA DE SÍMBOLOS

θ	Ângulo de arfagem
ϕ	Ângulo de rolamento
ψ	Ângulo de guinada
p	Velocidade angular de rolamento
q	Velocidade angular de arfagem
r	Velocidade angular de guinada
x	Latitude
y	Longitude
z	Altitude
v_x	Velocidade linear de latitude
v_y	Velocidade linear de longitude
v_z	Velocidade linear de altitude
i	Subscrito indica o referencial inercial
η	Subscrito indica o referencial não-inercial
m	Massa do quadricóptero
g	Gravidade
\vec{F}_η	Força de atuação no referencial não-inercial
j	Indexador
T_j	Força de propulsão do rotor j
\vec{D}_i	Força de arrasto
C_d	Coefficiente de arrasto
ρ	Massa específica do ar
A_d	Área do <i>drone</i> da seção perpendicular ao movimento

\vec{v}_r	Velocidade relativa entre o veículo e o ar
L	Distância do centro do quadricóptero à posição do eixo dos rotores
\mathbf{I}	Tensor de inércia do quadricóptero
I_{xx}	Momento de inércia no plano YZ do quadricóptero
I_{yy}	Momento de inércia no plano XZ do quadricóptero
I_{zz}	Momento de inércia no plano XY do quadricóptero
\vec{M}_R^η	Momento atuante resultante no referencial não-inercial
\vec{M}_a	Momento aparente
\vec{M}_g	Momento giroscópico
τ_j	Momento tursor do rotor j
I_r	Momento de inércia da hélice do quadricóptero
ω_j	Velocidade angular do rotor j
Ω	Somatório das velocidades angulares dos rotores
ω_{jRP}	Velocidade angular do rotor j em regime permanente
V_j	Tensão (elétrica) do rotor j
k_v	Constante de relação entre a tensão e a velocidade angular
k_f	Constante de relação entre a velocidade angular e a propulsão
k_t	Constante de relação entre a velocidade angular e o momento tursor
T_m	Constante de tempo do motor
S	Força de sustentação
σ	Matriz de conversão das velocidades angulares ao quadrado para variáveis auxiliares
U_j	Variável auxiliar do rotor j relacionada à propulsão
\mathbf{A}	Matriz de estados
\mathbf{B}	Matriz de entradas
s	Variável de domínio da frequência
Γ	Ação de controle no domínio da frequência
E	Erro de seguimento no domínio da frequência
K_p	Ganho proporcional do controlador PID

K_i	Ganho integral do controlador PID
K_d	Ganho derivativo do controlador PID
d	Frequência do filtro derivativo
ε	Conversão do sinal de controle para tensão no controlador PID
δ	Conversão do sinal de controle para tensão no controlador por linearização por realimentação de estados
α	Parâmetro arbitrário que pode representar a massa ou algum dos momentos de inércia
T_a	Tempo de assentamento
γ	Ação de controle
h	Função de saturação
x_d	Latitude do destino
y_d	Longitude do destino
x_o	Latitude da origem
y_o	Longitude da origem
z_c	Altitude de cruzeiro
y_p	Longitude prevista na rota
d_t	Distância do veículo à rota
d_d	Distância do veículo ao destino
T_{as}	Tempo de assentamento obtido na simulação
T_{at}	Tempo de assentamento teórico

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Objetivo Geral	14
1.2	Objetivos Específicos	14
2	DESCRIÇÃO DO SISTEMA	15
2.1	Movimentação	15
2.2	Sensoriamento	17
3	MODELAGEM MATEMÁTICA DO QUADRICÓPTERO	18
3.1	Referenciais	18
3.2	Dinâmica de translação	19
3.3	Dinâmica de rotação	20
3.4	Dinâmica dos atuadores	22
3.5	Sistema completo	23
3.6	Linearização	25
4	CONTROLE DA ATITUDE E ALTITUDE	26
4.1	Controle PID	26
4.1.1	Aplicação no Quadricóptero	27
4.2	Linearização por realimentação de estados	30
4.2.1	Aplicação no Quadricóptero	31
5	ALGORITMO DE VOO	34
5.1	Decolagem	34
5.2	Transporte	35
5.3	Prevenção de colisões	37
5.4	Pouso	37
5.5	Entrega	38
5.6	Retorno	38
5.7	Máquina de estados	38
6	ANÁLISE	40
6.1	Simulação	40
6.2	Modelagem	40
6.3	Avaliação do controlador	42
6.3.1	Controlador PD em cascata	42
6.3.2	Linearização por Realimentação de Estados	43
6.3.3	Comparação dos resultados dos controladores	44

6.4	Avaliação do algoritmo	45
7	CONCLUSÃO	48
	REFERÊNCIAS	49
	APÊNDICES	53
	APÊNDICE A - AMBIENTE DE SIMULAÇÃO	54
A.1	Desenvolvimento	54
A.2	Visualização tridimensional	56

1 INTRODUÇÃO

O avanço tecnológico tem transformado significativamente a maneira como são realizadas as atividades cotidianas, incluindo as operações de entrega de mercadorias. Nesse contexto, os veículos aéreos não tripulados (UAV's - *unmanned aerial vehicles*), popularmente conhecidos como *drones*, têm emergido como uma promissora alternativa para a entrega autônoma de produtos em áreas urbanas e rurais (JONES, 2017). Entre as diversas configurações de UAV's, os quadricópteros se destacam por sua versatilidade e capacidade de operar em espaços confinados, tornando-os ideais para aplicações em entregas autônomas (GATTESCHI et al., 2015).

Esse cenário tem gerado uma crescente necessidade de aprimorar os sistemas de logística e transporte, impulsionando a busca por soluções inovadoras. Nesse contexto, a adoção de quadricópteros emerge como uma resposta engenhosa, capaz de transpor obstáculos de infraestrutura e enfrentar congestionamentos urbanos, enquanto simultaneamente reduz de forma substancial os prazos de entrega. Dados atuais exemplificam a demanda expansiva no setor de entregas, como no caso da empresa Amazon que registrou um acréscimo de 45% (AMAZON, 2022b) no número de novos vendedores em 2020, em comparação a 2019. Da mesma forma, a empresa iFood apresentou um aumento de 200% (IFOOD, 2023) nas entregas mensais, totalizando 40 milhões, em 2021, em relação ao ano anterior. Esses dados demonstram que o expressivo crescimento nas entregas e colaborações decorre da disseminação de tecnologias avançadas, que permitem uma comunicação instantânea entre o vendedor e o cliente, além da influência da pandemia de COVID-19, que restringiu as opções dos usuários, incentivando-os a obter produtos sem sair de casa.

Alguns empreendimentos, como a Zipline, já são exemplos práticos dessa nova aplicação de UAV's, com aeronaves VTOL (*Vertical Take-Off and Landing*, ou, decolagem e aterrissagem verticais) que já efetuaram mais de 5 milhões de entregas (ZIPLINE, 2023). Além disso, a Amazon e o iFood já possuem planos de implementação de aeronaves não tripuladas, como o octacóptero MK4 (AMAZON, 2022a) e um hexacóptero em colaboração com a empresa Speedbird (IFOOD, 2022), respectivamente.

Tendo em vista o crescimento expressivo do uso de UAVs em serviços de entrega, este trabalho tem como objetivo projetar e desenvolver um controlador eficiente e seguro para quadricópteros utilizados em entregas autônomas. A proposta é explorar as principais técnicas de controle e navegação, aliadas às mais recentes tecnologias em sensores e sistemas de posicionamento, para garantir a eficácia, precisão e confiabilidade nas operações de entrega. A implementação bem-sucedida do controlador exige um estudo aprofundado da dinâmica de voo dessas aeronaves, bem como a integração inteligente de algoritmos de controle e sistemas de navegação. A segurança é uma preocupação primordial, considerando-se a interação dos quadricópteros com o espaço aéreo e com ambientes urbanos densamente povoados. Portanto, este trabalho também se compromete a abordar as estratégias necessárias para evitar colisões e garantir que as entregas ocorram sem incidentes.

1.1 OBJETIVO GERAL

O objetivo geral deste trabalho consiste em desenvolver um controlador para um quadricóptero que utilize um gerador de trajetórias para efetuar entregas autônomas. Para isso deve-se considerar suas diferentes configurações e implementações e determinar a mais promissora. Procura-se, ao fim deste trabalho de conclusão de curso em Engenharia de Controle e Automação, desenvolver um controlador para quadricópteros de entregas, e testá-lo em uma simulação.

1.2 OBJETIVOS ESPECÍFICOS

Com a finalidade geral definida, deve-se determinar as etapas necessárias para alcançá-la e quais os objetivos específicos que auxiliarão nesta trajetória. Assim, as etapas intermediárias contam com:

- Modelagem da dinâmica do quadricóptero;
- Pesquisa de controladores comumente utilizados em quadricópteros;
- Seleção do controlador baseado em características desejadas;
- Desenvolvimento de um gerador de trajetórias que gerará as referências para o controlador;
- Desenvolvimento de código computacional de integração do sistema gerador de trajetórias com o controlador;
- Simulação do comportamento do quadricóptero controlado;

Ao final deste trabalho, espera-se oferecer uma base sólida de conhecimento e uma proposta viável de controlador para quadricópteros em entregas autônomas, a fim de impulsionar o avanço e a aplicação dessa tecnologia inovadora no campo logístico, considerando suas potencialidades e as demandas cada vez mais presentes na sociedade contemporânea.

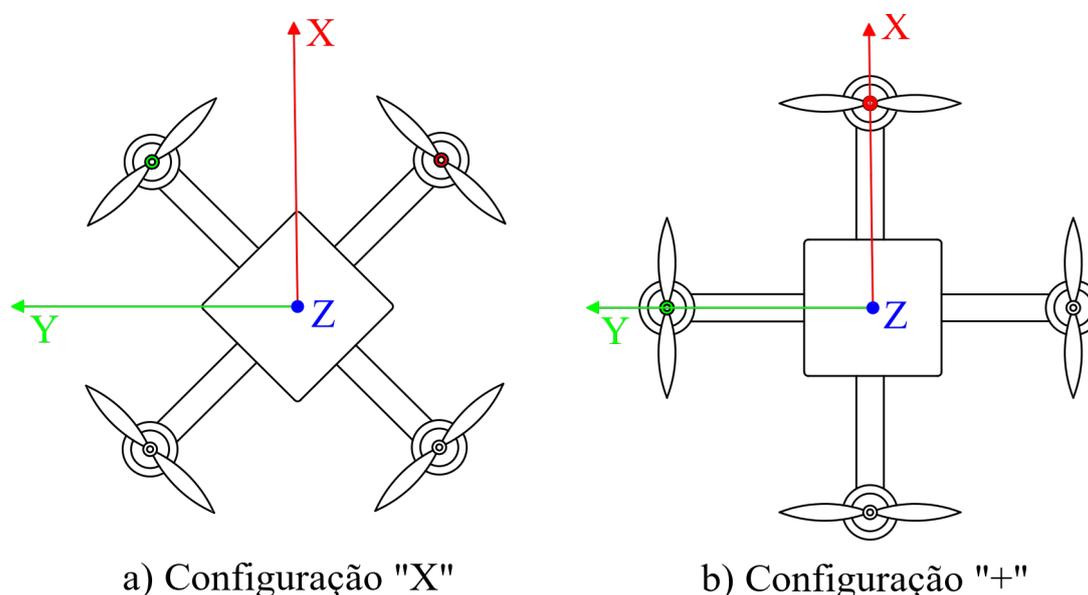
2 DESCRIÇÃO DO SISTEMA

Este capítulo do trabalho visa a fornecer uma visão geral da dinâmica de voo de um quadricóptero, destacando os principais conceitos e fatores que influenciam seu comportamento durante o voo.

2.1 MOVIMENTAÇÃO

Os quadricópteros são um tipo específico de veículo aéreo de múltiplos rotores, compostos por quatro motores dispostos em configuração cruzada, cada um acoplado a uma hélice, conforme exemplificado pela Figura 1.

Figura 1: Quadricópteros em diferentes configurações.



A modelagem geral é feita em duas configurações possíveis, como demonstrado na figura, denominadas da seguinte forma:

- **Configuração X:** o veículo avança em um ângulo de 45° com relação a um de seus braços, em forma de um "X" em relação à trajetória percorrida, conforme a Figura 1.a). Dois motores estão na parte dianteira e dois na parte traseira. Essa configuração é a mais comum e oferece uma boa combinação de estabilidade, manobrabilidade e resistência a falhas dos motores (NIEMIEC; GANDHI, 2016).

- **Configuração +**: o veículo avança colinear a um de seus braços, formando uma cruz em relação à trajetória, conforme a Figura 1.b). Um motor está na parte dianteira; outro, na traseira; os outros dois, nos lados esquerdo e direito. Essa configuração também é popular e oferece boa estabilidade.

Ambas as configurações têm suas próprias vantagens e desvantagens, e a escolha entre elas depende do uso pretendido do *drone* e das preferências do operador. Neste trabalho, será utilizada a configuração "mais" (+) pois essa permite simplificar o cálculo, isolar obstáculos em apenas um eixo de avanço e ainda possuir uma boa estabilidade.

O quadricóptero possui controle independente das rotações de cada uma de suas hélices, gerando forças que regulam tanto sua translação no espaço tridimensional, como sua atitude, que se refere à manutenção de uma posição e uma orientação definidas em relação a um sistema de coordenadas fixo. Tais regulagens são executadas por meio de quatro manobras básicas (CROCKER, 2010):

- **Ascendência/Descendência**: alteração da altitude do quadricóptero. O controle desse movimento é obtido variando as velocidades de rotação de todos rotores.
- **Arfagem (*pitch*)**: o ato de inclinar para frente ou para trás, que, conseqüentemente, faz o quadricóptero avançar ou recuar. Seu controle é obtido variando as velocidades de rotação entre os rotores traseiro e dianteiro.
- **Rolamento (*roll*)**: o ato de inclinar para a esquerda ou para a direita, que, conseqüentemente, desloca o quadricóptero lateralmente. Seu controle é obtido variando as velocidades de rotação entre os rotores esquerdo e direito.
- **Guinada (*yaw*)**: movimento ao longo do eixo vertical do quadricóptero, mudando sua orientação no plano horizontal. O controle desse movimento é obtido variando as velocidades de rotação entre rotores adjacentes.

A estabilidade desses movimentos é mantida através de sensores como giroscópios e acelerômetros, cujas informações alimentam um controlador de voo que ajusta automaticamente as velocidades dos rotores para responder às mudanças nas condições de voo. A Tabela 1 apresenta a convenção dos movimentos angulares utilizados neste trabalho, e a Tabela 2 a convenção da translação.

Tabela 1: Simbologia da atitude angular do sistema

Nome	Posição Angular	Velocidade Angular	Aceleração Angular
Arfagem	θ	q	\dot{q}
Rolamento	ϕ	p	\dot{p}
Guinada	ψ	r	\dot{r}

Tabela 2: Simbologia da atitude linear do sistema

Nome	Posição Linear	Velocidade Linear	Aceleração Linear
Latitude	x	\dot{x}	\ddot{x}
Longitude	y	\dot{y}	\ddot{y}
Altitude	z	\dot{z}	\ddot{z}

2.2 SENSORIAMENTO

A composição de sensores pode variar quanto à aplicação, mas o mínimo necessário consiste em (FLOREANO; WOOD, 2015):

- **Acelerômetro:** detecta variações em forças lineares por meio de micro-mecanismos internos. Quando parado, também consegue identificar a força da gravidade atuando sobre o dispositivo.
- **Giroscópio:** percebe variações em momentos angulares por meio de micro-mecanismos internos. Em alguns casos, o giroscópio está integrado ao acelerômetro como parte do *chip*.
- **Magnetômetro:** detecta o campo magnético da Terra da mesma forma que uma bússola mecânica, utilizando a direção norte como ponto de referência.
- **Barômetro:** lê a pressão atmosférica e, a partir de princípios físicos, converte na altitude atual.
- **GPS:** coleta a localização geodética do veículo na forma de latitude e longitude. Essas informações são interpretadas a partir do conhecimento da posição de três satélites orbitais no mínimo, de forma a obter a posição do objeto nos três eixos. No caso de um excedente de satélites, eles podem ser usados para melhorar a precisão de medição.

Além dos sensores essenciais, a utilização de sensores que detectem obstáculos durante a execução da trajetória é fundamental para garantir a segurança do voo. Uma solução eficaz consiste em equipar o veículo com dois sensores de distância: um posicionado na parte frontal, responsável por identificar obstáculos e informar o algoritmo de voo para evitar colisões, e outro na parte inferior, dedicado a monitorar a distância em relação ao solo. A escolha do princípio físico de cada sensor dependerá das características específicas da aplicação, como o ambiente de voo e o tipo de obstáculos a serem detectados.

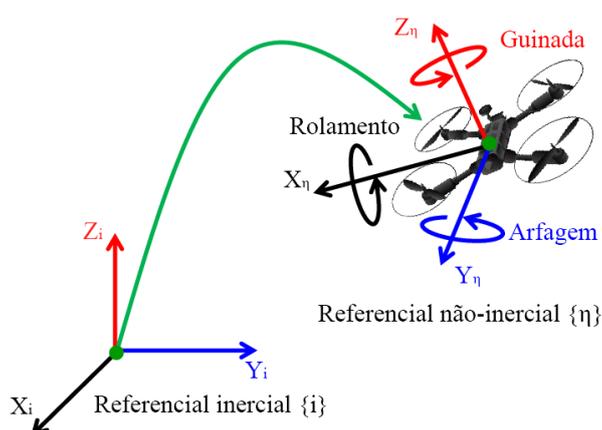
3 MODELAGEM MATEMÁTICA DO QUADRICÓPTERO

A fim de controlar um quadricóptero, é usualmente necessário conhecer sua dinâmica e ser capaz de descrevê-la em termos matemáticos. Assim, nesta seção são apresentados os equacionamentos para esse sistema a partir dos diagramas de corpo livre do veículo.

3.1 REFERENCIAIS

Com o objetivo de obter um conjunto de equações que descreva a dinâmica dos sistemas em foco, é necessário estabelecer as relações entre as forças e momentos dos atuadores com os deslocamentos e velocidades correspondentes do veículo. Essas relações podem ser interpretadas em diferentes sistemas de coordenadas para facilitar a modelagem, os quais serão explorados nesta seção. Na Figura 2 são apresentados os sistemas de coordenadas utilizados neste trabalho.

Figura 2: Referenciais inercial e não-inercial indicados respectivamente por i e η .



Fonte: (HASHIM; BROWN; MCISAAC, 2019)

Os referenciais podem ser descritos como:

1. **Referencial inercial:** possui orientação e posição fixas. É normalmente considerado fixo a um ponto na superfície da Terra, seguindo a orientação baseada no referencial magnético global, onde X , Y e Z correspondem respectivamente às direções Leste, Norte e Cima. Sua representação na simbologia é feita por meio do índice i subscrito

ou sobrescrito, de acordo com a indicação pretendida, ou fica subentendida em caso de ausência de referência explícita a um referencial (coordenadas ou orientações sem qualquer subscrito ou sobrescrito).

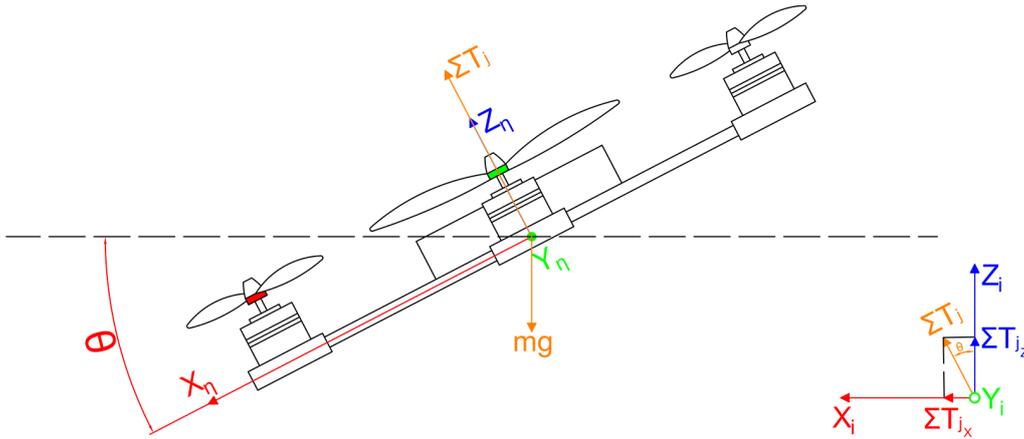
2. **Referencial não-inercial:** sua orientação e posição acompanham o veículo, podendo ainda serem relacionados ao referencial inercial. Convencionou-se a simbologia η (subscrito ou sobre-escrito) como um referencial não-inercial.

Ambos os referenciais são importantes no contexto do presente projeto e serão utilizados no cálculo da dinâmica de translação e de rotação, apresentado a seguir.

3.2 DINÂMICA DE TRANSLAÇÃO

A dinâmica de translação é descrita a partir de um referencial não-inercial, conforme apresentado na Figura 3, onde m representa a massa do quadricóptero, e g a gravidade.

Figura 3: Diagrama de corpo livre da seção um quadricóptero em diferentes sistemas de coordenadas.



A atuação dos motores é dada pela Equação 1, onde T_j representa a força de propulsão individual na direção Z_η exercida por cada rotor j , e F_η a força resultante dessas atuações.

$$\vec{F}_\eta = \begin{bmatrix} 0 \\ 0 \\ \sum T_j \end{bmatrix} \quad (1)$$

Para compreender o efeito dessa força na translação, é necessário considerá-lo no referencial inercial do veículo. Este procedimento é realizado por meio da matriz de rotação ${}^i\mathbf{R}_\eta$ (CRAIG, 2014), conforme apresentado na Equação 2, em que os ϕ , θ e ψ representam os ângulos de rolamento, arfagem e guinada respectivamente.

$${}^i\mathbf{R}_\eta = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}_z \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}_y \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}_x \quad (2)$$

Além da atuação e força gravitacional, existe ainda a componente de força de arrasto (BATCHELOR, 1967) \vec{D}_i , que aparece sempre na direção contrária ao movimento. Seu equacionamento é apresentado na Equação 3, com C_d representando o coeficiente de arrasto, ρ a massa específica do fluido, A_d a área da seção perpendicular ao movimento e \vec{v}_r a velocidade relativa entre o movimento e o ar, que pode ou não estar em movimento.

$$\vec{D}_i = -\frac{1}{2}C_d\rho A_d v_r^2 \hat{v}_r \quad (3)$$

A força de arrasto, será desconsiderada para o sistema completo e para o cálculo dos controladores por ser de difícil medição. Ao invés disso, ela será incorporada no controle como uma perturbação não modelada. Assim, realiza-se o somatório de forças do sistema inercial conforme a Equação 4, considerando as atuações rotacionadas e o peso do *drone* que está na direção Z_i .

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = {}^i \mathbf{R}_\eta \vec{F}_\eta + m \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \quad (4)$$

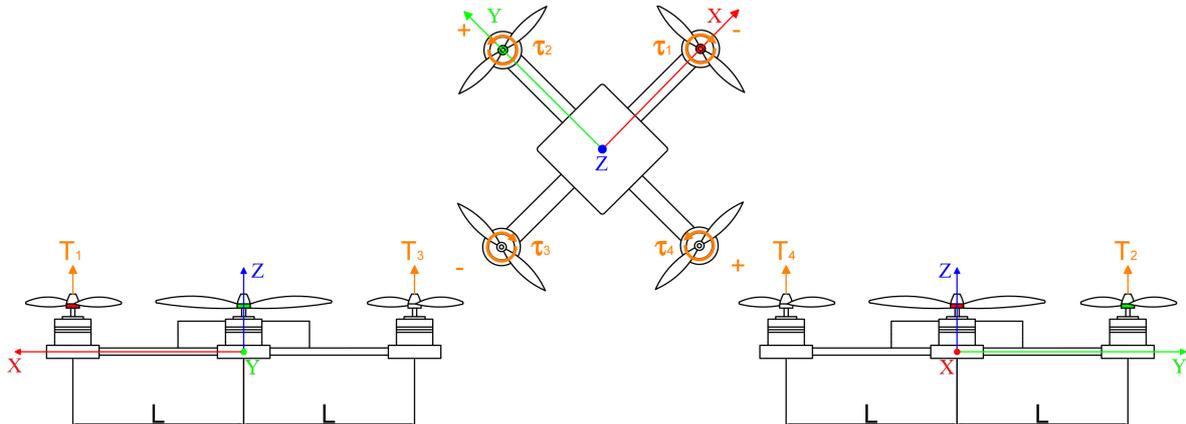
Por fim, a dinâmica de translação no referencial inercial é descrita por meio da Equação 5.

$$\begin{aligned} \ddot{x} &= [\sin(\psi) \sin(\phi) + \cos(\psi) \sin(\theta) \cos(\phi)] m^{-1} \Sigma T_j \\ \ddot{y} &= [-\cos(\psi) \sin(\phi) + \sin(\psi) \sin(\theta) \cos(\phi)] m^{-1} \Sigma T_j \\ \ddot{z} &= -g + m^{-1} \cos(\theta) \cos(\phi) \Sigma T_j \end{aligned} \quad (5)$$

3.3 DINÂMICA DE ROTAÇÃO

Já, para a dinâmica de rotação, parte-se do momento resultante no corpo rígido a partir de um referencial não-inercial, conforme o diagrama de corpo livre que é apresentado na Figura 4, onde L representa a distância do centro à posição do eixo dos rotores do quadricóptero.

Figura 4: Diagrama de corpo livre da atitude de um quadricóptero no referencial não-inercial.



A fim de descrever suas propriedades inerciais, é necessário inicialmente conhecer o tensor de inércia \mathbf{I} do quadricóptero, apresentado na Equação 6, em que I_{xx} , I_{yy} e I_{zz} representam os momentos de inércia em cada um dos planos do quadricóptero. Os produtos de inércia, nesse caso, são desprezados devido à simetria geométrica do quadricóptero (WANG et al., 2016).

$$\mathbf{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \approx \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (6)$$

Assim, o somatório de momentos no referencial não-inercial é apresentado na Equação 7, onde $\vec{\mathbf{M}}_R^n$ representa o momento atuante resultante no sistema não inercial, $\vec{\mathbf{M}}_a$ o momento aparente e $\vec{\mathbf{M}}_g$ o momento giroscópico.

$$\vec{\mathbf{M}}_R^n = \mathbf{I} \begin{bmatrix} \dot{p}_\eta \\ \dot{q}_\eta \\ \dot{r}_\eta \end{bmatrix} = \Sigma \vec{\mathbf{M}}_\eta = \vec{\mathbf{M}}_a + \vec{\mathbf{M}}_g \quad (7)$$

O momento aparente pode ser calculado a partir apenas do diagrama de corpo livre apresentado na Figura 4, com seu resultado descrito na Equação 8, onde τ_j representa o momento tursor exercido pelo rotor j .

$$\vec{\mathbf{M}}_a = \begin{bmatrix} L(T_2 - T_4) \\ L(T_3 - T_1) \\ -\tau_1 + \tau_2 - \tau_3 + \tau_4 \end{bmatrix} \quad (8)$$

Porém, existe ainda um efeito inercial contrário ao movimento angular, denominado momento de giroscópico, o qual ocorre por fato de os rotores possuírem uma direção fixa em seu eixo de rotação, no referencial não-inercial, apesar de existir uma velocidade angular não nula (HIBBELER, 2011). O equacionamento do momento giroscópico é apresentado na Equação 9, onde I_r representa o momento de inércia da hélice do veículo, ω_j representa a velocidade angular do rotor j , Ω é a velocidade angular resultante dos rotores, e p , q e r representam as velocidades angulares do veículo referentes ao rolamento, à arfagem e à guinada respectivamente.

$$\vec{\mathbf{M}}_g = \begin{bmatrix} 0 \\ 0 \\ I_r \Omega \end{bmatrix} \times \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} -I_r \Omega q \\ I_r \Omega p \\ 0 \end{bmatrix} \quad \Omega = -\omega_1 + \omega_2 - \omega_3 + \omega_4 \quad (9)$$

Logo, o momento resultante no referencial não-inercial é apresentado na Equação 10 ao combinar as equações 8 e 9 na Equação 7.

$$\vec{\mathbf{M}}_R^n = \mathbf{I} \begin{bmatrix} \dot{p}_\eta \\ \dot{q}_\eta \\ \dot{r}_\eta \end{bmatrix} = \begin{bmatrix} -I_r \Omega q + L(T_2 - T_4) \\ I_r \Omega p + L(T_3 - T_1) \\ -\tau_1 + \tau_2 - \tau_3 + \tau_4 \end{bmatrix} \quad (10)$$

Por fim, para considerar a dinâmica no referencial inercial, é necessário considerar o efeito que a rotação do sistema de coordenadas causa a partir de um observador fixo, o qual ocasiona as pseudoforças de Coriolis e centrífuga (THORNTON; MARION, 2012) apresentado na Equação 11.

$$\mathbf{I} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \left(\mathbf{I} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \right) = \vec{\mathbf{M}}_R^n \quad (11)$$

Portanto, ao substituir a Equação 10 na Equação 11 e isolar as acelerações, é obtida a dinâmica de rotação em um referencial inercial, apresentada na Equação 12.

$$\begin{aligned}\dot{p} &= [(I_{zz} - I_{yy})qr - I_r\Omega q + L(T_2 - T_4)] I_{xx}^{-1} \\ \dot{q} &= [(I_{xx} - I_{zz})pr + I_r\Omega p + L(T_3 - T_1)] I_{yy}^{-1} \\ \dot{r} &= [(I_{yy} - I_{xx})pq - \tau_1 + \tau_2 - \tau_3 + \tau_4] I_{zz}^{-1}\end{aligned}\quad (12)$$

3.4 DINÂMICA DOS ATUADORES

Depois de obtidos os equacionamentos que descrevem a aerodinâmica do veículo, para controlá-lo é ainda necessário caracterizar uma grandeza diretamente controlável. Portanto, é necessário compreender como um sinal elétrico enviado aos motores se transforma em forças e momentos de propulsão.

Os motores usualmente utilizados na construção de um quadricóptero são de corrente contínua sem escovas, que, a partir de um controlador intermediário chamado de ESC (*eletronic speed controller*, ou controlador eletrônico de velocidade) (SHIN; TEH, 2018), podem ser modelados como motores de corrente contínua tradicionais (POPOOLA; OLADEJO; ODEYEMI, 2015).

Para um motor de corrente contínua, a relação entre um valor de tensão aplicado e uma velocidade angular é linear em seu regime permanente e conforme a Equação 13. Na equação, ω_{jRP} representa a velocidade angular do rotor j em regime permanente, k_v a constante que relaciona um valor de tensão à velocidade angular, V a tensão no rotor j .

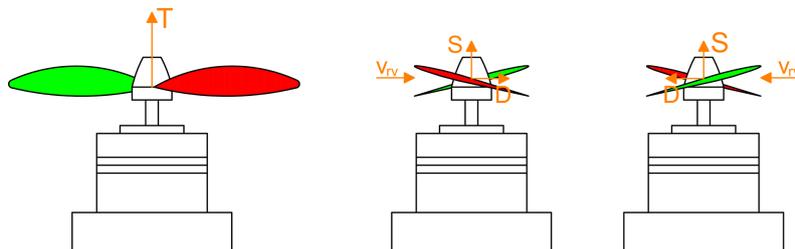
$$\omega_{jRP} = k_v V_j \quad (13)$$

Já, ao considerar a dinâmica completa do *drone*, considera-se apenas a dinâmica do subsistema mecânico do motor, pois a dinâmica da parte elétrica é usualmente muito mais rápida que a do subsistema mecânico. É apresentada na Equação 14 a relação entre um sinal elétrico e a velocidade angular no rotor, na qual ω_j representa a velocidade angular do rotor j e T_m a constante de tempo dessa dinâmica.

$$\dot{\omega}_j = T_m^{-1}(\omega_{jRP} - \omega_j) = T_m^{-1}(k_v V - \omega_j) \quad (14)$$

Obtida a velocidade angular de uma hélice acoplada a um motor, deve-se então descrever a dinâmica que ocasiona a força de propulsão, conforme ilustrado na Figura 5.

Figura 5: Diagrama de corpo livre da hélice do rotor com v_{rv} representando a velocidade relativa com relação ao ar, T a força propulsora, S a força de sustentação e D a componente de arrasto.



Observa-se que o movimento angular da hélice ocasiona uma velocidade relativa com o ar, o qual, em consequência, é propulsionado verticalmente com relação ao eixo do rotor. A força que impulsiona o ar para baixo reage na hélice, o que causa uma propulsão apenas para cima e nenhum movimento horizontal, devido ao cancelamento das componentes laterais de arraste.

Esse comportamento é denominado Sustentação Aerodinâmica (JR., 2010) e é representado por meio da Equação 15, onde S representa a força de sustentação, C_L o coeficiente de sustentação, A a área projetada sobre o plano perpendicular à direção do vento e v a velocidade relativa entre a pá e o ar e r representa o raio da hélice.

$$S = \frac{1}{2}C_L\rho Av^2 = \frac{1}{2}C_L\rho Ar^2\omega_j^2 \quad (15)$$

Ainda, os coeficientes associadas à sustentação podem ser negligenciadas para o cálculo da propulsão, visto que na literatura esses coeficientes são obtidos como apenas uma constante a partir de ensaios mecânicos realizados nos rotores (MUZAR; LANTEIGNE, 2016). Assim, considera-se a força de propulsão T_j conforme apresentada na Equação 16, na qual k_f representa a constante de propulsão.

$$T_j = k_f\omega_j^2 \quad (16)$$

O momento tursor da hélice é realizado de maneira similar por meio da grandeza k_t (BRYANT, 2020), denominada constante de torque, de forma que é descrito conforme a Equação 17.

$$\tau_j = k_t\omega_j^2 \quad (17)$$

Logo, ao combinar as equações elétricas e mecânicas, obtêm-se as equações 18 e 19, para as forças e momentos respectivamente, que descrevem como o sinal elétrico implica em uma ação mecânica.

$$T_j = k_f(k_v V_j - T_m\dot{\omega}_j)^2 \quad (18)$$

$$\tau_j = k_t(k_v V_j - T_m\dot{\omega}_j)^2 \quad (19)$$

É importante destacar que os atuadores reais apresentam uma região de saturação na qual o controle é aplicável. Os valores de tensão V_j são, portanto, descritos pela Equação 20, onde V_{max} representa o valor máximo de tensão possível, e V o valor de tensão aplicado sem saturação.

$$V_j(V, 0, V_{max}) = \begin{cases} V_{max} & V > V_{max} \\ V & 0 \geq V \geq V_{max} \\ 0 & V < 0 \end{cases} \quad (20)$$

3.5 SISTEMA COMPLETO

As relações entre as dinâmicas rotacionais e translacionais apresentam interdependência entre os diferentes motores, o que pode tornar a metodologia de controle mais complexa. Assim, os efeitos individuais dos motores devem ser substituídos por variáveis

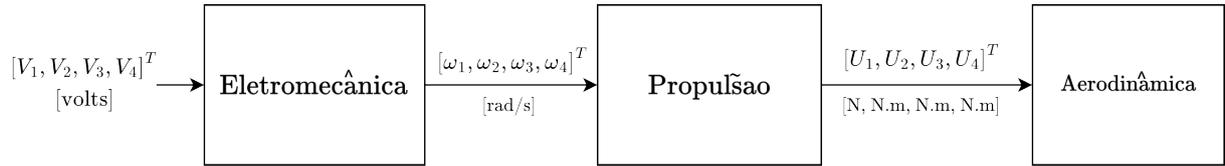
auxiliares linearmente dependentes da velocidade angular ao quadrado. Essa relação é apresentada na Equação 21, na qual σ representa a matriz de conversão das velocidades angulares ao quadrado para as variáveis auxiliares.

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \sigma \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}, \quad \sigma = \begin{bmatrix} k_f & k_f & k_f & k_f \\ 0 & L \cdot k_f & 0 & -L \cdot k_f \\ -L \cdot k_f & 0 & L \cdot k_f & 0 \\ -k_t & k_t & -k_t & k_t \end{bmatrix} \quad (21)$$

O modelo do quadricóptero pode ser separado em três blocos distintos, os quais são ilustrados na Figura 6 e representam as partes:

1. **Eletromecânica:** efetua a conversão de sinais elétricos em velocidades angulares.
2. **Propulsão:** traduz as velocidades angulares em forças e momentos propulsores, que são combinados em variáveis mais simples de serem manipuladas.
3. **Aerodinâmica:** representa a dinâmica de voo do quadricóptero.

Figura 6: Diagrama de blocos do sistema que evidencia as entradas e saídas de cada bloco com unidades no SI.



O conjunto completo de equações pode ser representado como a combinação do efeito de propulsão por parte dos rotores (conforme equações 18 e 19), combinação linear dos efeitos de propulsão (na Equação 21) e realização do voo (Equação 22). Na equação, v_x, v_y, v_z representam as velocidades do veículo nas direções x, y e z respectivamente.

$$\begin{aligned} \dot{v}_x &= \ddot{x} = [\sin(\psi) \sin(\phi) + \cos(\psi) \sin(\theta) \cos(\phi)] m^{-1} U_1 \\ \dot{v}_y &= \ddot{y} = [-\cos(\psi) \sin(\phi) + \sin(\psi) \sin(\theta) \cos(\phi)] m^{-1} U_1 \\ \dot{v}_z &= \ddot{z} = -g + m^{-1} \cos(\theta) \cos(\phi) U_1 \\ \dot{p} &= [(I_{zz} - I_{yy})qr - I_r \Omega q + U_2] I_{xx}^{-1} \\ \dot{q} &= [(I_{xx} - I_{zz})pr + I_r \Omega p + U_3] I_{yy}^{-1} \\ \dot{r} &= [(I_{yy} - I_{xx})pq + U_4] I_{zz}^{-1} \\ \dot{x} &= v_x \\ \dot{y} &= v_y \\ \dot{z} &= v_z \\ \dot{\phi} &= p \\ \dot{\theta} &= q \\ \dot{\psi} &= r \end{aligned} \quad (22)$$

3.6 LINEARIZAÇÃO

Como o modelo proposto é não-linear, é necessário obter-se um modelo linearizado para utilização de técnicas lineares. O cálculo deste modelo é realizado a partir da Linearização Jacobiana do sistema (D. K. ARROWSMITH BSC PHD, 1992), conforme a Equação 23, na qual \vec{x} representa o vetor de estados $[\dot{z}, p, q, r]^T$, \vec{U} o vetor de entradas $[U_1, U_2, U_3, U_4]^T$, \mathbf{A} representa a matriz de estados linearizados, \mathbf{B} representa a matriz de entradas linearizadas e o subscrito 0 representa a aplicação em torno do ponto de equilíbrio. Ainda, são linearizados apenas as dinâmicas necessárias para o controle do veículo (que possuem U_j diferentes entre si).

$$\dot{\vec{x}} = \mathbf{A}\Delta\vec{x} + \mathbf{B}\Delta\vec{U} = \begin{bmatrix} \frac{\partial \dot{z}}{\partial z} & \frac{\partial \dot{z}}{\partial p} & \frac{\partial \dot{z}}{\partial q} & \frac{\partial \dot{z}}{\partial r} \\ \frac{\partial \dot{p}}{\partial z} & \frac{\partial \dot{p}}{\partial p} & \frac{\partial \dot{p}}{\partial q} & \frac{\partial \dot{p}}{\partial r} \\ \frac{\partial \dot{q}}{\partial z} & \frac{\partial \dot{q}}{\partial p} & \frac{\partial \dot{q}}{\partial q} & \frac{\partial \dot{q}}{\partial r} \\ \frac{\partial \dot{r}}{\partial z} & \frac{\partial \dot{r}}{\partial p} & \frac{\partial \dot{r}}{\partial q} & \frac{\partial \dot{r}}{\partial r} \end{bmatrix}_0 \Delta\vec{x} + \begin{bmatrix} \frac{\partial \dot{z}}{\partial U_1} & \frac{\partial \dot{z}}{\partial U_2} & \frac{\partial \dot{z}}{\partial U_3} & \frac{\partial \dot{z}}{\partial U_4} \\ \frac{\partial \dot{p}}{\partial U_1} & \frac{\partial \dot{p}}{\partial U_2} & \frac{\partial \dot{p}}{\partial U_3} & \frac{\partial \dot{p}}{\partial U_4} \\ \frac{\partial \dot{q}}{\partial U_1} & \frac{\partial \dot{q}}{\partial U_2} & \frac{\partial \dot{q}}{\partial U_3} & \frac{\partial \dot{q}}{\partial U_4} \\ \frac{\partial \dot{r}}{\partial U_1} & \frac{\partial \dot{r}}{\partial U_2} & \frac{\partial \dot{r}}{\partial U_3} & \frac{\partial \dot{r}}{\partial U_4} \end{bmatrix}_0 \Delta\vec{U} \quad (23)$$

Assim, a Linearização Jacobiana para a aerodinâmica do quadricóptero implicam no cálculo das matrizes \mathbf{A} e \mathbf{B} , conforme apresentado nas equações 24 e 25.

$$\mathbf{A} = \begin{bmatrix} 0 & -m^{-1} \cos(\theta_0) \sin(\phi_0) p_0 U_{10} & -m^{-1} \sin(\theta_0) \cos(\phi_0) q_0 U_{10} & 0 \\ 0 & 0 & ((I_{zz} - I_{yy})r_0 - I_r \Omega_0) I_{xx}^{-1} & ((I_{zz} - I_{yy})q_0) I_{xx}^{-1} \\ 0 & ((I_{xx} - I_{zz})r_0 + I_r \Omega_0) I_{yy}^{-1} & 0 & ((I_{xx} - I_{zz})p_0) I_{yy}^{-1} \\ 0 & ((I_{yy} - I_{xx})q_0) I_{zz}^{-1} & ((I_{yy} - I_{xx})p_0) I_{zz}^{-1} & 0 \end{bmatrix} \quad (24)$$

$$\mathbf{B} = \begin{bmatrix} m^{-1} \cos(\theta_0) \cos(\phi_0) & 0 & 0 & 0 \\ 0 & I_{xx}^{-1} & 0 & 0 \\ 0 & 0 & I_{yy}^{-1} & 0 \\ 0 & 0 & 0 & I_{zz}^{-1} \end{bmatrix} \quad (25)$$

Ainda, os termos da matriz de transição \mathbf{A} podem ser considerados nulos com o emprego de algumas premissas:

1. as velocidades angulares iniciais são nulas
2. a soma das velocidades dos rotores iniciais é nula

Como essas afirmações são comuns à inicialização do veículo nas aplicações genéricas, elas serão aplicadas neste trabalho. Assim, o novo modelo linearizado da aerodinâmica é apresentado na Equação 26. Observa-se que, dessa forma, o sistema deixa de ser interdependente, pois cada estado depende exclusivamente de uma entrada.

$$\begin{bmatrix} \ddot{z} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} m^{-1} & 0 & 0 & 0 \\ 0 & I_{xx}^{-1} & 0 & 0 \\ 0 & 0 & I_{yy}^{-1} & 0 \\ 0 & 0 & 0 & I_{zz}^{-1} \end{bmatrix} \Delta\vec{U} \quad (26)$$

4 CONTROLE DA ATITUDE E ALTITUDE

Neste capítulo, são abordadas duas metodologias de controle para o quadricóptero: o controle PID e a técnica de linearização por realimentação de estados. Estas estratégias foram escolhidas não só por representar duas alternativas plausíveis para o objetivo, mas também para comparar a utilização de um método linear com um método não linear. Nas seções seguintes são apresentadas revisões bibliográficas de cada estratégia, seguida da sua aplicação neste trabalho.

4.1 CONTROLE PID

O controlador proporcional, integral e derivativo (PID) é caracterizado por operações lineares aplicadas ao erro da variável de processo (ANG; CHONG; LI, 2005), equivalentes a um desvio no valor medido com relação à referência. A determinação dos parâmetros pode ser efetuado a partir de métodos heurísticos, como no método de Ziegler-Nichols (ZIEGLER; NICHOLS, 1942), ou por métodos analíticos, como por exemplo a alocação de polos (OGATA, 2009). Como neste trabalho foi efetuada a modelagem teórica da planta, será utilizado um método por alocação de polos (LEONELLO et al., 2021).

O controlador PID, ainda em sua forma monovariável, é apresentado na Equação 27 no domínio da frequência, com o objetivo de explicitar os polos e zeros do controlador. Na equação, s representa a frequência atuante, d a frequência de corte da ação derivativa, Γ o sinal de controle, E o desvio da referência com relação à saída, K_p o ganho proporcional, K_i o ganho integral e K_d o ganho derivativo.

$$C_{PID}(s) = \frac{\Gamma(s)}{E(s)} = K_p + K_i \frac{1}{s} + K_d s \frac{d}{s + d} \quad (27)$$

Além disso, se for escolhido uma frequência para o filtro derivativo muito maior que as frequências dos polos e zeros presentes em malha fechada, seu efeito pode ser desconsiderado para o cálculo. Logo, na Equação 28, é apresentado o controlador PID considerando esta observação.

$$C_{PID}(s) \approx \frac{\Gamma(s)}{E(s)} = \frac{1}{s} (K_d s^2 + K_p s + K_i) \quad , \quad s \ll d \quad (28)$$

Já a abordagem multivariável (para este trabalho) consiste em um conjunto de controladores do tipo PID monovariáveis atuando de maneira simultânea (GARRIDO et al., 2024). Sua forma geral é apresentada na Equação 29, na qual $C_{j,k}$ representa um controlador PID com subscritos j e k indicando o erro de seguimento da referência j com relação ao sinal de controle k para o respectivo controlador monovariável, e N o número total de referências e sinais de controle.

$$\mathbf{C}_{PID}(s) = \begin{bmatrix} C_{1,1} & 0 & 0 & \dots & 0 \\ 0 & C_{2,2} & 0 & \dots & 0 \\ 0 & 0 & C_{3,3} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & C_{N,N} \end{bmatrix} \quad (29)$$

Essa topologia foi implementada em diversos outros trabalhos no âmbito do controle de quadricópteros, como no apresentado por Nengsheng em 2017 (NENGSHENG, 2017) e por Fernando H.C.T.E em (FERNANDO et al., 2013) em 2013. Ainda, para melhorar os resultados, foram introduzidos esquemas baseados em aprendizado de máquina, conforme Koch (KOCH et al., 2019) em 2019, para manipulação dinâmica dos parâmetros do PID que pode vir a atenuar trepidações e melhorar a dinâmica de voo durante diferentes condições.

4.1.1 Aplicação no Quadricóptero

Para o cálculo dos parâmetros do controlador PID de maneira analítica, parte-se da aerodinâmica linearizada (mostrada na Equação 26), por se tratar de um método de controle linear. Sua representação no domínio da frequência é apresentada na Equação 30.

$$\begin{bmatrix} \Delta Z(s) \\ \Delta \Phi(s) \\ \Delta \Theta(s) \\ \Delta \Psi(s) \end{bmatrix} = \frac{1}{s^2} \begin{bmatrix} m^{-1} & 0 & 0 & 0 \\ 0 & I_{xx}^{-1} & 0 & 0 \\ 0 & 0 & I_{yy}^{-1} & 0 \\ 0 & 0 & 0 & I_{zz}^{-1} \end{bmatrix} \begin{bmatrix} \Delta U_1(s) \\ \Delta U_2(s) \\ \Delta U_3(s) \\ \Delta U_4(s) \end{bmatrix} \quad (30)$$

Entretanto, como o sinal de controle aplicado depende da dinâmica de atuação dos rotores, essa dinâmica deve ser considerada no modelo linear. A planta do sistema completo linearizado é obtida a partir da utilização das equações 18, 19 e 21 na Equação 31.

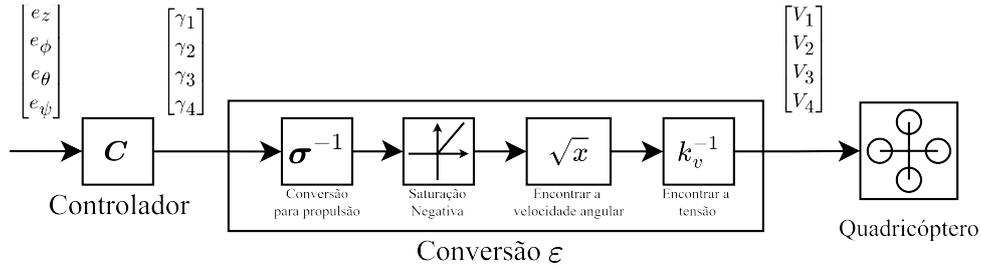
$$\begin{bmatrix} \Delta Z(s) \\ \Delta \Phi(s) \\ \Delta \Theta(s) \\ \Delta \Psi(s) \end{bmatrix} = \frac{T_m^{-1}}{s + T_m^{-1}} \frac{1}{s^2} \begin{bmatrix} m^{-1} & 0 & 0 & 0 \\ 0 & I_{xx}^{-1} & 0 & 0 \\ 0 & 0 & I_{yy}^{-1} & 0 \\ 0 & 0 & 0 & I_{zz}^{-1} \end{bmatrix} \begin{bmatrix} \Delta \Gamma_1(s) \\ \Delta \Gamma_2(s) \\ \Delta \Gamma_3(s) \\ \Delta \Gamma_4(s) \end{bmatrix} \quad (31)$$

Ainda, como o cálculo do controlador é baseado em forças e momentos, e como esse sinal também deve ser reestruturado para a propulsão em cada rotor; deve ser introduzida uma operação de conversão que execute essa tarefa. A função de conversão ε deve ser tal que: o ganho elétrico seja compensado; a característica quadrática da propulsão seja levada em conta; e a propulsão seja reconstruída a partir das variáveis auxiliares determinadas na Equação 21. Como os rotores giram em um sentido fixo após a instalação dos motores, a propulsão negativa não é fisicamente possível. Para incorporar essa restrição ao modelo, a função L , definida na Equação 33, é utilizada para saturar os sinais de propulsão negativos no valor nulo. Assim, essa função deve possuir a forma apresentada na Equação 32 e representação em diagrama de blocos conforme a Figura 7.

$$V_j = \varepsilon(\vec{\gamma}) = k_v^{-1} \sqrt{L(\zeta_j, 0, +\infty)} \quad , \quad \vec{\zeta} = \sigma^{-1} \vec{\gamma} \quad (32)$$

$$L(x, l_+, l_-) = \begin{cases} l_+ & x > l_+ \\ x & l_- \geq x \geq l_+ \\ l_- & x < l_- \end{cases} \quad (33)$$

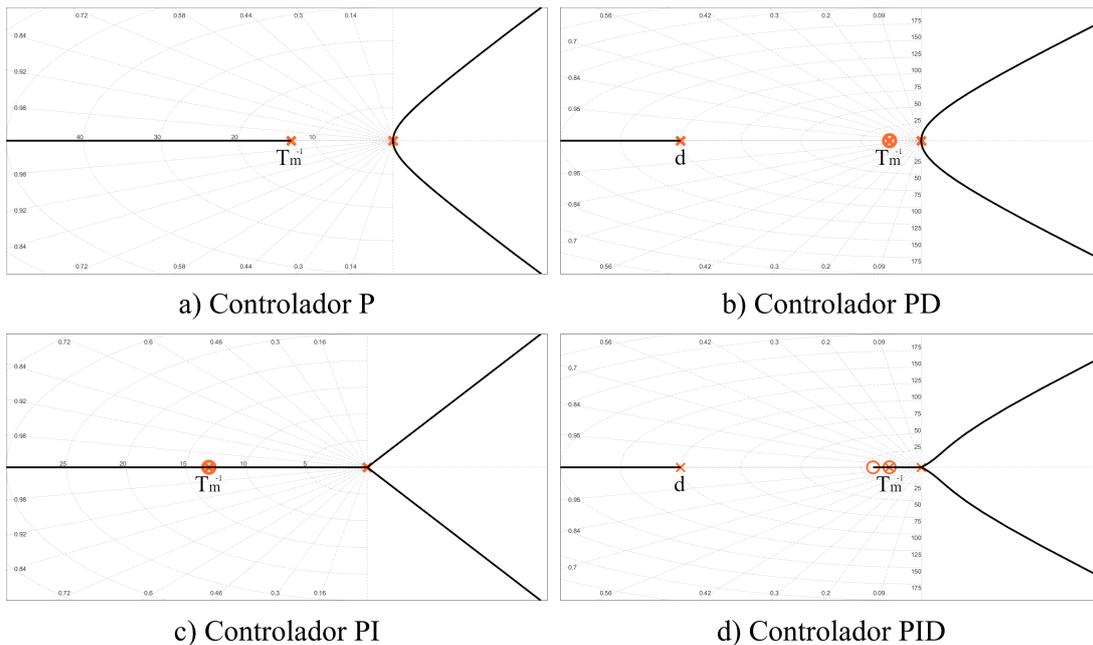
Figura 7: Diagrama de blocos do sinal de controle para o quadricóptero com bloco de conversão ε especificado.



A partir da nova equação da planta, observa-se que as dinâmicas são todas semelhantes entre si, portanto, para calcular o controlador, basta considerar o cálculo para uma versão equivalente a esta dinâmica e substituí-la posteriormente. Considera-se então um termo de ganho α , um sinal de controle Γ_e e saída Y_e para a planta genérica unidimensional representada na Equação 34 no domínio frequência.

$$\Delta Y_e(s) = \alpha^{-1} \frac{1}{s^2} \frac{T_m^{-1}}{s + T_m^{-1}} \Delta \Gamma_e(s) \quad , \quad \begin{aligned} Y_e &\in [Z, \Phi, \Theta, \Psi] \\ \alpha &\in [m, I_{xx}, I_{yy}, I_{zz}] \\ \Gamma_e &\in [\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4] \end{aligned} \quad (34)$$

Figura 8: Lugar geométrico das raízes do sistema realimentado com um controlador PID, PI, PD e P.

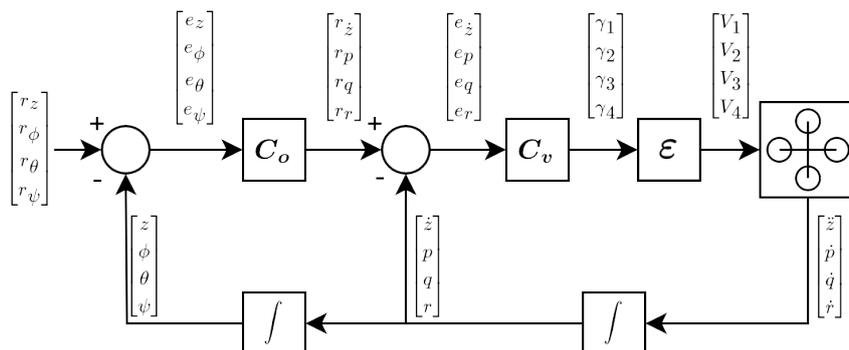


A aplicação direta, dos controladores PID, PI, PD ou mesmo P, não é encorajada, visto que o lugar das raízes dessas malhas fechadas tende à instabilização. Conforme

mostra a Figura 8, caso o controlador possua um zero, se o mesmo puder cancelar o polo eletromecânico será possível remover essa limitação.

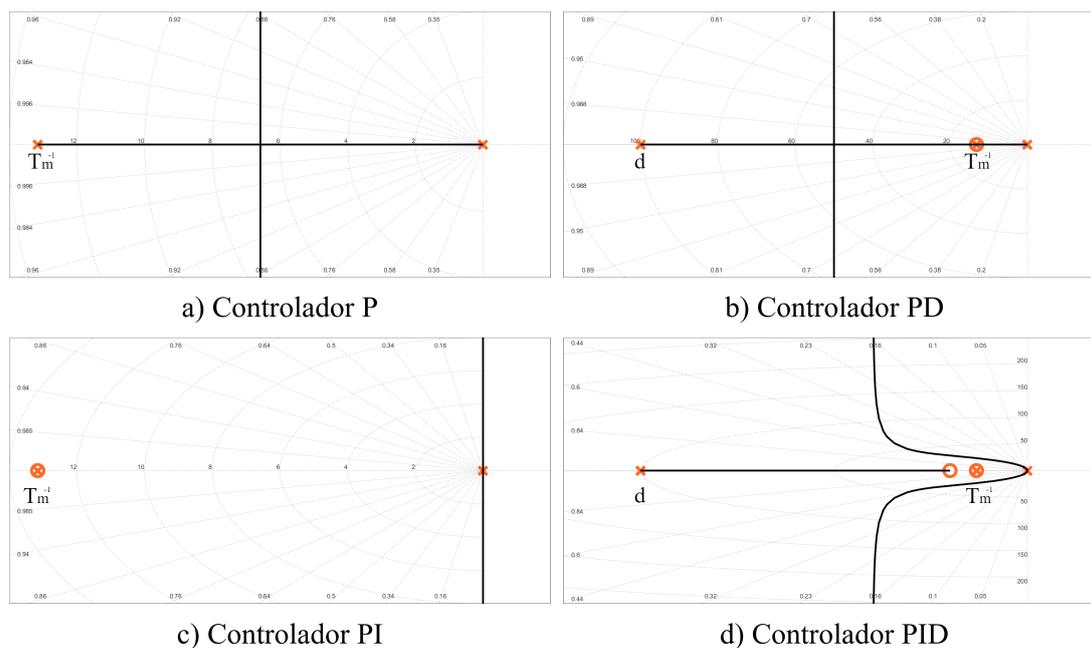
Esse comportamento em grande parte está ligado ao fato da planta possuir um integrador duplo. Assim, ao invés de considerar apenas um controlador PID diretamente associado ao erro das variáveis odométricas, deve-se considerar dois controladores do tipo PID em cascata que objetivam controlar a velocidade e a posição em dois laços de controle, representados na Figura 9.

Figura 9: Diagrama de blocos da malha fechada do quadricóptero com dois controladores PID em cascata, C_o para a odometria e C_v para a velocidade.



Com essa nova configuração, pondera-se novamente sobre qual configuração de PID deve ser considerada, entre PID, PI, PD e P. Logo, novamente, são traçados os lugares das raízes de cada configuração com a mesma premissa para ambas as malhas e o resultado é ilustrado na Figura 10.

Figura 10: Lugar geométrico das raízes para o laço de velocidade com um controlador PID, PI, PD e P.



A aplicação de dois controladores do tipo PD em cascata é a opção considerada adequada, por não apresentar sobressano e ter um tempo de acomodação T_a limitado

apenas pelo filtro derivativo. É importante destacar que, no laço de posição, a característica do controlador PD é análoga à do laço de velocidade.

Os ganhos podem ser determinados utilizando a estratégia baseado no lugar das raízes (A. S. BAZANELLA, 2005) em função do tempo de acomodação desejado. No geral, o tempo de acomodação é aproximado por quatro vezes a constante de tempo da função de laço para alcançar a referência com uma tolerância de 2%. Na Tabela 3 são apresentados os ganhos e com a premissa de que o zero de cada controlador PD cancelará o polo de seu respectivo laço.

Tabela 3: *Ganhos para os controladores do tipo PD unidimensionais equivalentes em cascata.*

	K_p	K_d
C_{ve}	$4\alpha T_a^{-1}$	$4\alpha T_a^{-1} T_m$
C_{oe}	$4T_a^{-1}$	1

Por fim, a escolha de T_a fica apenas limitada à saturação dos atuadores, visto que, em um caso real, o tempo de acomodação não é ilimitado. O controlador final é apresentado na Equação 35. Ainda, impõe-se que a frequência do filtro da ação derivativa seja alta o suficiente para que seu comportamento seja considerado desprezível.

$$\begin{aligned}
 \mathbf{C}_v(s) &= \begin{bmatrix} C_{vZ} & 0 & 0 & 0 \\ 0 & C_{v\Phi} & 0 & 0 \\ 0 & 0 & C_{v\Theta} & 0 \\ 0 & 0 & 0 & C_{v\Psi} \end{bmatrix} & \begin{aligned} C_{vZ} &= 4mT_a^{-1}T_m(s + T_m^{-1}) \\ C_{v\Phi} &= 4I_{xx}T_a^{-1}T_m(s + T_m^{-1}) \\ C_{v\Theta} &= 4I_{yy}T_a^{-1}T_m(s + T_m^{-1}) \\ C_{v\Psi} &= 4I_{zz}T_a^{-1}T_m(s + T_m^{-1}) \end{aligned} \\
 \mathbf{C}_o(s) &= \begin{bmatrix} C_{oe} & 0 & 0 & 0 \\ 0 & C_{oe} & 0 & 0 \\ 0 & 0 & C_{oe} & 0 \\ 0 & 0 & 0 & C_{oe} \end{bmatrix} & C_{oe} = s + 4T_a^{-1}
 \end{aligned} \tag{35}$$

4.2 LINEARIZAÇÃO POR REALIMENTAÇÃO DE ESTADOS

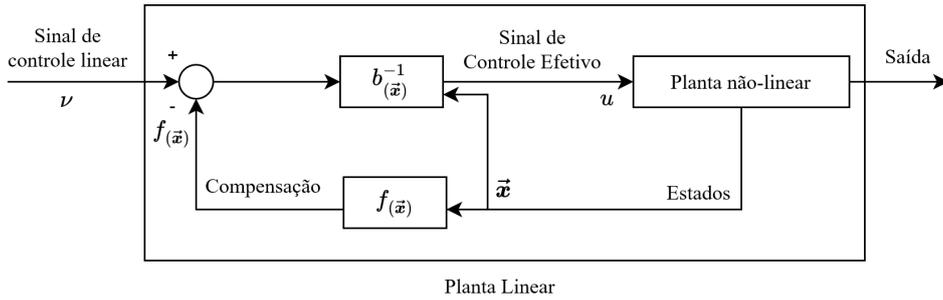
Apesar da técnica de aproximação do modelo pela matriz Jacobiana ser amplamente utilizada pela simplicidade de implementação e projeto, explora-se também uma segunda alternativa para possibilitar a comparação dos métodos. Assim, considera-se um novo controlador baseado na Linearização por Realimentação de Estados (LI, 1991). Essa técnica é baseada no cancelamento das parcelas não lineares do sistema algebricamente, a partir de suas variáveis de estado, de forma a obter um macro-sistema que apresente características lineares.

De maneira usual, seu cálculo é efetuado a partir das derivadas de Lie (YANO, 1957). Porém, caso o sistema seja afim com relação a controle/entrada (ou seja, caso o termo da ação de controle multiplique a função dos estados no modelo do sistema), como demonstrado na Equação 36, a função de controle pode ser determinada apenas isolando o valor do controle efetivamente aplicado (SOBCZYK, 2009), conforme a Equação 37. Nas equações, $x_j^{(n)}$ representa a n -ésima derivação do estado j , $f(\vec{x})$ e $b(\vec{x})$ são funções não lineares dos estados, $b^{-1}(\vec{x})$ o inverso da função de estado $b(\vec{x})$, ν é o sinal de controle linear desejado e u é o sinal de controle efetivamente aplicado. Ainda, na Figura 11, é ilustrado o diagrama de blocos com esta condição.

$$x_j^{(n)} = f(\vec{x}) + b(\vec{x})u \quad (36)$$

$$u = b^{-1}(\vec{x})(\nu - f(\vec{x})) \quad (37)$$

Figura 11: Diagrama de blocos da linearização por realimentação de estados para um sistema na forma canônica controlável.



A eficácia da técnica de cancelamento de dinâmica depende fortemente da fidelidade do modelo em relação à planta real. Em outras palavras, desvios entre o modelo utilizado para o projeto do controlador e o comportamento real do sistema podem comprometer o desempenho e até mesmo a estabilidade da malha de controle. Para contornar essa limitação e melhorar a robustez do sistema, é comum o uso de técnicas adicionais de realimentação, lineares ou não lineares, que visam minimizar os efeitos das incertezas e perturbações. Como exemplos de técnicas para aumentar a robustez, podemos citar o controle adaptativo, explorado por Sobczyk em 2016 (SOBCZYK et al., 2016), e o uso de redes neurais, apresentado por Borges em 2021 (BORGES et al., 2021). Em ambos os casos, a robustez do controlador é aprimorada através do ajuste contínuo dos parâmetros utilizados nas parcelas de realimentação linearizante, adaptando-se dinamicamente às variações e incertezas da planta.

Essa categoria de controladores é bastante utilizada quando trabalhando com sistemas não lineares no geral, como demonstrado por Benchouche em 2021 (BENCHOUCHE; MELLAH; BENNOUNA, 2021). No âmbito dos quadricópteros, também existem estudos como os efetuados por Voos em 2009 (VOOS, 2009), e também por Bonna em 2015 (BONNA; CAMINO, 2015) e, ainda, uma comparação com outro método não linear pelo *Jornal Internacional de Controle, Automação e Sistemas* (LEE; JIN KIM; SASTRY, 2009).

4.2.1 Aplicação no Quadricóptero

O sistema do quadricóptero como um todo não é considerado afim com relação a controle/entrada, porém a sua aerodinâmica é, conforme apresentada na Equação 22 para os estados \ddot{z} , \dot{p} , \dot{q} e \dot{r} . Assim, para aplicar o método ao quadricóptero é necessário compreender quais são as operações não-lineares $f(\vec{x})$ e $b(\vec{x})$ presentes na aerodinâmica, conforme denotado na Equação 38.

$$\begin{aligned}
\ddot{z} &= -g & + & m^{-1} \cos(\theta) \cos(\phi) \cdot U_1 \\
\dot{p} &= [(I_{zz} - I_{yy})qr - I_r \Omega \cdot q] I_{xx}^{-1} & + & I_{xx}^{-1} \cdot U_2 \\
\dot{q} &= [(I_{xx} - I_{zz})pr + I_r \Omega \cdot p] I_{yy}^{-1} & + & I_{yy}^{-1} \cdot U_3 \\
\dot{r} &= [(I_{yy} - I_{xx})pq] I_{zz}^{-1} & + & I_{zz}^{-1} \cdot U_4
\end{aligned} \tag{38}$$

$f(\vec{x})$

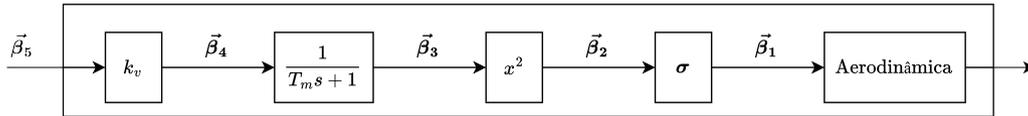
$b(\vec{x})$

Portanto, o resultado dessas operações não lineares na forma multivariável e com os valores de $b(\vec{x})$ já invertidos (termo a termo), é apresentado na Equação 39. Cabe observar que, diferente do controlador linear calculado pela matriz Jacobiana, o modelo da planta opera em dimensões das acelerações lineares e angulares do ponto de vista de $\vec{\gamma}$, ao invés das forças e momentos.

$$f(\vec{x}) = \begin{bmatrix} -g \\ [(I_{zz} - I_{yy})qr - I_r \Omega \cdot q] I_{xx}^{-1} \\ [(I_{xx} - I_{zz})pr + I_r \Omega \cdot p] I_{yy}^{-1} \\ [(I_{yy} - I_{xx})pq] I_{zz}^{-1} \end{bmatrix}, \quad b^{-1}(\vec{x}) = \begin{bmatrix} m \cos(\theta)^{-1} \cos(\phi)^{-1} \\ I_{xx} \\ I_{yy} \\ I_{zz} \end{bmatrix} \tag{39}$$

O cálculo efetuado até então considera um sistema capaz de atuar diretamente na aerodinâmica do quadricóptero. A fim de controlar o sistema completo, deve-se trazer este sinal diretamente aplicado para onde seja possível manipular a entrada e, por consequência, afetar a aerodinâmica. Esse processo, é ilustrado na Figura 12 e descrito na Equação 40, com variáveis auxiliares β .

Figura 12: Diagrama de blocos da relação entre o sinal aplicado e o sinal aerodinâmico.

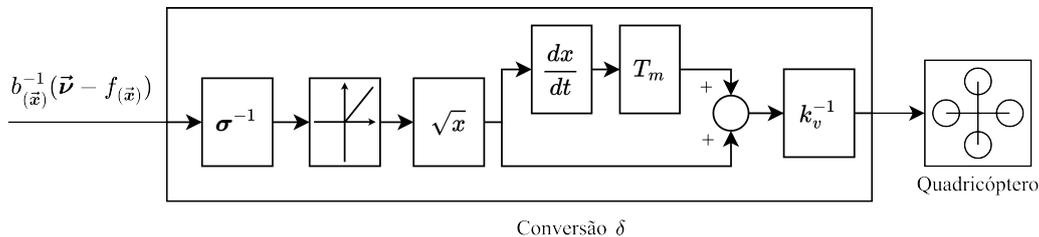


$$\vec{\beta}_1 = b_{(\vec{x})}^{-1}(\vec{\nu} - f(\vec{x})) \quad , \quad \vec{\beta}_2 = \sigma^{-1} \vec{\beta}_1 \tag{40}$$

$$\beta_{3_j} = \sqrt{L(\beta_{2_j}, 0, +\infty)} \quad , \quad \beta_{4_j} = \beta_{3_j} + T_m \dot{\beta}_{3_j} \quad , \quad \beta_{5_j} = k_v^{-1} \beta_{4_j}$$

Assim, de maneira equivalente ao realizado no controle linear pela aproximação Jacobiana, obtém-se uma relação de conversão entre o sinal aplicado e o sinal aerodinâmico. A conversão do sinal é simbolizada pela operação δ , que é representada na Figura 13.

Figura 13: Diagrama de blocos da relação entre o sinal aplicado e o sinal aerodinâmico.



É importante observar que a operação de compensação da dinâmica do rotor, inicialmente não causal, requer a introdução de um filtro derivativo para garantir a causalidade, a exemplo do filtro utilizado na ação derivativa de um controlador PID.

Por fim, com uma malha linearizada pela realimentação de estados, deve-se calcular a nova configuração PID que atende este caso, visto que a dinâmica do motor foi agora incluída no bloco de conversão. A nova planta linear apresenta uma dinâmica equivalente à aerodinâmica apenas, conforme apresentada na Equação 41.

$$\begin{bmatrix} \Delta Z(s) \\ \Delta \Phi(s) \\ \Delta \Theta(s) \\ \Delta \Psi(s) \end{bmatrix} = \frac{1}{s^2} \begin{bmatrix} \Delta \Gamma_1(s) \\ \Delta \Gamma_2(s) \\ \Delta \Gamma_3(s) \\ \Delta \Gamma_4(s) \end{bmatrix} \quad (41)$$

De maneira análoga ao cálculo para o controlador linear, percebe-se que a melhor configuração para a planta compensada é a de um controlador proporcional para a velocidade e o mesmo PD do controlador linear para a posição. Ao aplicar novamente a teoria do lugar das raízes, determina-se que o controlador deve possuir os parâmetros apresentados na Tabela 4 e na Equação 42.

Tabela 4: *Ganhos para os controladores unidimensionais genéricos em cascata.*

	K_p	K_d
C_{ve}	$4 \cdot T_a^{-1}$	-
C_{oe}	$4 \cdot T_a^{-1}$	1

$$\begin{aligned} C_v(s) &= \begin{bmatrix} 4T_a^{-1} & 0 & 0 & 0 \\ 0 & 4T_a^{-1} & 0 & 0 \\ 0 & 0 & 4T_a^{-1} & 0 \\ 0 & 0 & 0 & 4T_a^{-1} \end{bmatrix} \\ C_o(s) &= \begin{bmatrix} s + 4T_a^{-1} & 0 & 0 & 0 \\ 0 & s + 4T_a^{-1} & 0 & 0 \\ 0 & 0 & s + 4T_a^{-1} & 0 \\ 0 & 0 & 0 & s + 4T_a^{-1} \end{bmatrix} \end{aligned} \quad (42)$$

Ainda, como evidentemente o quadricóptero possui uma limitação física de atuação, nem todas as ações de controle podem ser realizadas de forma ideal. Esse fator é ainda mais crítico por se tratar de um sistema multivariável, com interdependências entre suas variáveis de estado. Durante manobras bruscas, como em mudanças rápidas de altitude, a limitação da referência de velocidade se torna crucial para evitar a saturação dos atuadores. Assim, tanto no controlador linear quanto no não linear, as referências de velocidade calculadas são limitadas por um valor máximo para cada estado, garantindo que o quadricóptero opere dentro de seus limites físicos. Essa estratégia garante um comportamento mais estável e previsível do quadricóptero, evitando oscilações e até mesmo a perda de controle.

5 ALGORITMO DE VOO

Este capítulo detalha o algoritmo desenvolvido, responsável por gerar as referências para o controlador, garantindo que o quadricóptero siga a rota planejada e execute as tarefas de entrega com sucesso. As regras de voo e os métodos para o cumprimento de cada etapa da missão serão abordados em detalhes nas seções subsequentes. Os desenvolvimentos foram baseados em condições geométricas presentes na rota e no trabalho do autor Kareem (KAREEM; OLEIWI; MOHAMED, 2023).

5.1 DECOLAGEM

A decolagem deve acontecer logo após a inicialização e imposição do destino no veículo. As características do estado de decolagem devem ser tais que:

- a altitude seja incrementada até a sua altitude de cruzeiro z_c , considerada para um voo seguro;
- o quadricóptero permaneça na mesma posição planar (em x e y), de forma que seu deslocamento de translação seja nulo para evitar possíveis colisões com estruturas próximas, pelo fato de o *drone* ainda não estar voando numa altitude considerada segura;
- o veículo se oriente precisamente para a posição de destino;

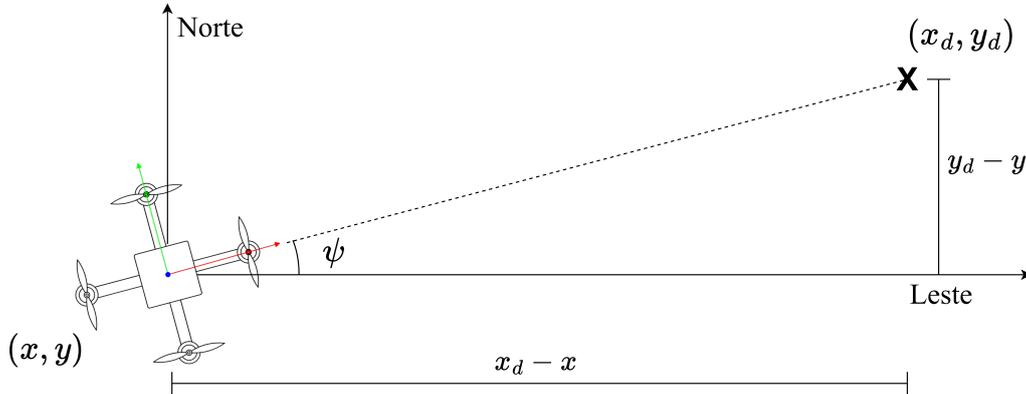
Tanto a altitude quanto a orientação da guinada são diretamente controláveis por meio das referências atribuídas ao controlador. No entanto, o deslocamento nulo deve ser alcançado a partir de uma função que equilibre as referências de arfagem e rolamento, considerando a influência de ventos. Esta função deve possuir a ação relativa ao desvio aparente do seu ponto inicial para evitar seu crescimento, porém de forma limitada. Uma função simples que possui essa característica é a própria função afim, quando limitada em uma região, conforme representado na Equação 43. Na equação, λ representa o limite de operação superior e inferior da referência em módulo, e k_l representa o ganho entre o desvio e a referência aplicada.

$$h(d) = \begin{cases} -\lambda & d < -k_l^{-1}\lambda \\ k_l d & -k_l^{-1}\lambda \leq d \leq k_l^{-1}\lambda \\ \lambda & d > k_l^{-1}\lambda \end{cases} \quad (43)$$

Para que a guinada esteja corretamente orientada, é necessário calcular o ângulo de abertura entre a latitude e o trajeto, conforme ilustrado na Figura 14. O trajeto escolhido é uma reta entre a origem e o destino, representando a menor distância entre esses dois

pontos. Assim, basta utilizar a função do arco-tangente para encontrar essa orientação. Detalha-se que essa operação deve ser acompanhada por uma avaliação dos sinais da variação latitudinal e longitudinal, para determinar corretamente em qual quadrante o ângulo estará.

Figura 14: Orientação da guinada para seguimento da rota planejada.



Logo, este estado deve possuir as referências apresentadas na Tabela 5, onde Δx_η e Δy_η representam os desvios com relação à posição global, fixada no início do estado, e orientada com o *drone*, que devem ser equilibrados para manter a posição planar do veículo constante. A diferença de sinal é relativo à referência que cada estado deve aplicar para corrigir o erro, de forma que um ϕ positivo decrementa a posição em y_η , e um θ positivo incrementa a posição em x_η .

Tabela 5: Referências do controlador para a decolagem.

	z	ϕ	θ	ψ
Decolagem	z_c	$h(-\Delta y_\eta)$	$h(\Delta x_\eta)$	$\text{atan}\left(\frac{y_d - y}{x_d - x}\right)$

5.2 TRANSPORTE

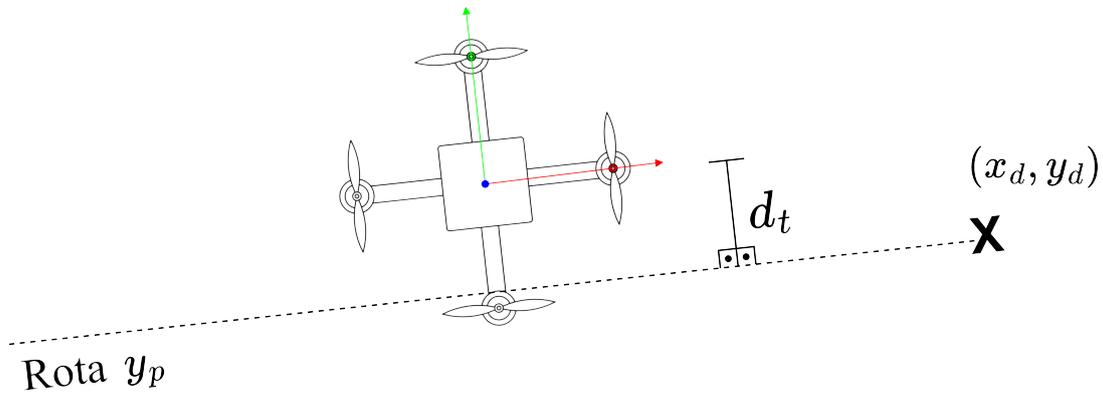
O transporte deve ocorrer logo após o veículo alcançar uma altitude viável para o deslocamento planar, entendida como um percentual da altitude de cruzeiro pretendida. As características para o estado do transporte devem ser tais que:

- a altitude mantenha-se na altitude de cruzeiro z_c ;
- a arfagem seja constante e cause o deslocamento ao destino;
- o rolamento equilibre o veículo em sua trajetória, a fim de evitar algum desvio de rota por ação de ventos mais fortes;
- a guinada mantenha a direção x_η do quadricóptero apontando para o destino geodeticamente;

A fim de minimizar o gasto energético do veículo, a trajetória deve ser calculada com a menor distância possível. No deslocamento aéreo para longas distâncias, a menor distância é representada pelo arco de círculo máximo entre os dois pontos pretendidos, por considerar a curvatura da Terra. Contudo, para esta aplicação, as distâncias de voo serão inferiores a 111 km, o que corresponde a um arco de aproximadamente um grau de curvatura terrestre. Dessa forma, a trajetória pode ser aproximada por uma reta sem perda significativa de precisão. Logo, a trajetória do veículo deve ser retilínea e representada conforme a Equação 44. Na equação, x_o e y_o representam respectivamente a latitude e longitude de origem, e x_d e y_d representam a latitude e longitude de destino.

$$y_p(x) = \frac{y_d - y_o}{x_d - x_o}x + y_o \quad (44)$$

Figura 15: Distância entre o quadricóptero e a rota planejada para um desvio anômalo.



O equilíbrio do rolamento deve ser tal que a função h receba o desvio da posição global do quadricóptero com relação à trajetória, conforme ilustrado na Figura 15. Essa determinação é equivalente à distância de um ponto a uma reta e possui a forma apresentada na Equação 45.

$$d_t(x, y) = \text{sgn}(y - y_p(x)) \left| \frac{y_d - y_o}{x_d - x_o}x_o + 2 \cdot y_o \right| \left[\left(\frac{y_d - y_o}{x_d - x_o} \right) + 1 \right]^{-0.5} \quad (45)$$

Por fim, os estados para o transporte são apresentados na Tabela 6.

Tabela 6: Referências do controlador para o transporte.

	z	ϕ	θ	ψ
Transporte	z_c	$h(-d_t(x, y))$	θ_{max}	$\text{atan}\left(\frac{y_d - y}{x_d - x}\right)$

5.3 PREVENÇÃO DE COLISÕES

Enquanto em um estado de transporte, o sensor frontal do quadricóptero pode detectar um empecilho em sua rota, neste caso o estado mudará para o estado de prevenção de colisão. Este estado deve possuir as características de:

- elevar o valor de z gradualmente até que o sensor frontal não detecte mais nenhum obstáculo, atualizando o valor de z_c no fim do estado;
- manter o deslocamento nulo;
- manter a guinada;

Logo, para este estado, o importante é manter o veículo parado e gradualmente encontrar a altitude de cruzeiro ideal para a rota. Na tabela, z_c^* representa a altitude de cruzeiro que está sendo constantemente incrementada.

Tabela 7: Referências do controlador para a prevenção de colisões.

	z	ϕ	θ	ψ
Colisão	z_c^*	$h(-\Delta y_\eta)$	$h(\Delta x_\eta)$	$\text{atan}\left(\frac{y_d - y}{x_d - x}\right)$

5.4 POUSO

O pouso deve ocorrer após o transporte chegar na posição pretendida, entendido como um valor de distância de tolerância determinado pelo projetista. Este estado possui as seguintes características:

- descer até que o sensor de distância na parte inferior do veículo detecte o solo;
- manter o quadricóptero na posição de destino;
- manter a guinada;

De maneira análoga à situação de prevenção em colisões, este estado deve receber o sinal do sensor de distância para concluir a sua ação. Nesse caso, a altitude de referência é representada por z_d e deve ser, no mínimo, inferior à altitude de destino, para que o veículo consiga chegar próximo do solo e detectá-lo por meio do sensor de distância.

Para manter o veículo na posição de destino, a sua referência de arfagem deve estar relacionada à distância restante, enquanto o rolamento controla possíveis desvios equivalente ao transporte. A distância com relação ao destino é representada por d_d e é apresentada na Equação 46.

$$d_d = \text{sgn}(y_d - y) \sqrt{(y_d - y)^2 + (x_d - x)^2} \quad (46)$$

Ainda, para evitar que a guinada instabilize, seu cálculo agora pode ser aproximado pelos valores da origem, de forma a evitar a divisão por zero. Na Tabela 8, são representados os estados para o pouso.

Tabela 8: Referências do controlador para o pouso.

	z	ϕ	θ	ψ
Pouso	z_d	$h(-d_t(x, y))$	$h(d_d)$	$\text{atan}\left(\frac{y_d - y_o}{x_d - x_o}\right)$

5.5 ENTREGA

Esta etapa deve ocorrer logo após a detecção do solo pelo sistema de pouso, se a entrega ainda não tiver sido efetuada. Durante a entrega, o veículo deve manter sua altitude e aplicar as demais referências como aplicado durante o estado de pouso. Neste caso, o que importa para o algoritmo é o ato de liberar a entrega e possuir a informação de que a entrega foi efetuada. Após isso, a massa m e as inércias I_{xx} , I_{yy} e I_{zz} devem ser atualizadas.

5.6 RETORNO

Na etapa de retorno, o processo de geração de trajetória é reutilizada, mas no sentido contrário. Logo, o que antes era o destino vira a origem e vice-versa. Na finalização do pouso de retorno, como a entrega já foi efetuada, não existirá liberação de um pacote, mas sim a conclusão da missão e desligamento da máquina.

5.7 MÁQUINA DE ESTADOS

Por fim, a sequência de passos e suas referências são mostrados na Tabela 9 e representados na Figura 16 por uma máquina de estados de Moore (MOORE et al., 1956). Na tabela, $d_t(x, y)^*$ representa a função $d_t(x, y)$ invertendo destino e origem, assim como d_o , que representa a distância à origem.

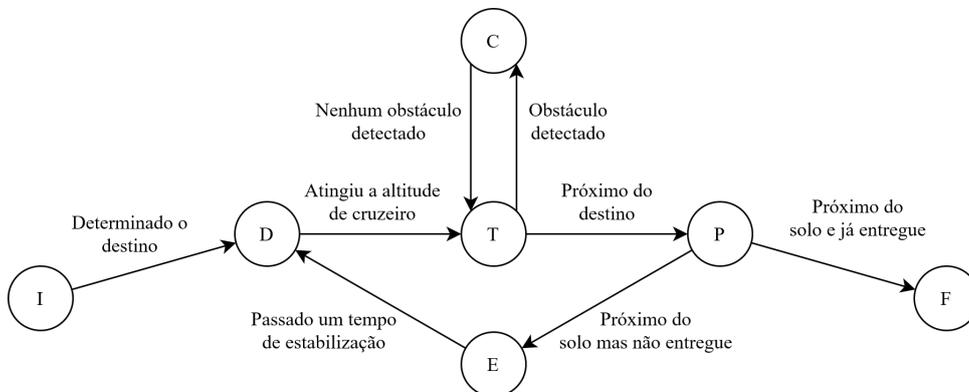
Tabela 9: Referências do controlador para todos os estados.

		Símbolo	z	ϕ	θ	ψ
Entregando	Decolagem	D	z_c	$h(-\Delta y_\eta)$	$h(\Delta x_\eta)$	$\text{atan}\left(\frac{y_d - y}{x_d - x}\right)$
	Transporte	T	z_c	$h(-d_t(x, y))$	θ_{max}	$\text{atan}\left(\frac{y_d - y}{x_d - x}\right)$
	Colisão	C	z_c^*	$h(-\Delta y_\eta)$	$h(\Delta x_\eta)$	$\text{atan}\left(\frac{y_d - y}{x_d - x}\right)$
	Pouso	P	z_d	$h(-d_t(x, y))$	$h(d_d)$	$\text{atan}\left(\frac{y_d - y_o}{x_d - x_o}\right)$
Retornando	Decolagem	D	z_c	$h(-\Delta y_\eta)$	$h(\Delta x_\eta)$	$\text{atan}\left(\frac{y_o - y}{x_o - x}\right)$
	Transporte	T	z_c	$h(-d_t^*(x, y))$	θ_{max}	$\text{atan}\left(\frac{y_o - y}{x_o - x}\right)$
	Colisão	C	z_c^*	$h(-\Delta y_\eta)$	$h(\Delta x_\eta)$	$\text{atan}\left(\frac{y_o - y}{x_o - x}\right)$
	Pouso	P	z_d	$h(-d_t^*(x, y))$	$h(d_o)$	$\text{atan}\left(\frac{y_o - y_d}{x_o - x_d}\right)$

As informações listadas na tabela já compõem o algoritmo, porém, usualmente, são utilizados diagramas para representar cada um dos possíveis estados de maneira clara. Uma forma de representação é a partir de uma máquina de estados, que informa, principalmente, como cada transição se relaciona com outros estados. Assim, com essas

informações coletadas, monta-se um diagrama de máquina de estados que representa o algoritmo em ação, apresentado na Figura 16.

Figura 16: Máquina de estados para o algoritmo de entregas autônomas.



Neste diagrama em específico, não foram detalhadas as formas de transição para melhor clareza. A máquina em questão é do tipo Moore (MOORE et al., 1956), pois a saída é determinada pelo estado, e não pela transição. Os estados I, F e E representam, respectivamente, os estados de inicialização, finalização e entrega.

6 ANÁLISE

Este capítulo apresenta uma análise do sistema de controle desenvolvido para um quadricóptero em aplicações de entregas autônomas. Inicialmente, a modelagem dinâmica do quadricóptero será comparada com um modelo de referência existente na literatura, elucidando as semelhanças ou diferenças. Em seguida, os controladores propostos são analisados em detalhes, considerando diferentes tipos de referências de atitude e avaliando o desempenho do sistema em termos de precisão e estabilidade. Por fim, o algoritmo de controle de trajetória será explorado, destacando suas características e limitações.

6.1 SIMULAÇÃO

O ambiente de simulação desenvolvido foi programado em linguagem C++ e está documentado no Apêndice A. Nele, consta a implementação do modelo em sua forma não linear, conforme as definições da Seção 3.5 e consideração do arrasto aerodinâmico, e todas as constatações de saturação determinadas. Essa simulação será utilizada tanto para analisar os algoritmos de controle desenvolvidos, quanto para analisar o algoritmo de voo em si. Para garantir que o modelo desenvolvido neste trabalho está adequado, além de garantir que a implementação em código também está condizente com as expectativas, será feita uma comparação com o modelo proposto na Seção 6.2.

6.2 MODELAGEM

A fim de assegurar a confiabilidade da simulação, faz-se imprescindível a validação dos resultados obtidos. Para tanto, será realizada uma comparação dos resultados com os de um trabalho de referência, buscando verificar a correspondência entre os resultados e validar a precisão do modelo desenvolvido. Os dados serão comparados com aqueles obtidos por David C. H. realizado durante o evento *IEEE CSS Video Clip Contest 2014* (ALLGOWER, 2014), o qual possui uma implementação em MATLAB utilizada como referência e está disponível em um repositório Github (DAVID, 2014). Os parâmetros adotados para essa comparação e demais simulações são apresentados na Tabela 10. Ainda,

Tabela 10: Parâmetros do modelo adotados para a simulação de dados.

m	1,023 kg	L	$2,223 \times 10^{-1}$ m	g	9,810 m/s ²
I_{xx}	$9,500 \times 10^{-3}$ kg.m ²	I_{yy}	$9,500 \times 10^{-3}$ kg.m ²	I_{zz}	$1,858 \times 10^{-2}$ kg.m ²
I_r	$3,788 \times 10^{-6}$ kg.m ²	T_m	$7,600 \times 10^{-2}$ s	V_{max}	$1,200 \times 10^1$ V
k_v	$7,884 \times 10^1$ rad/s.V	k_f	$1,355 \times 10^{-5}$ N.s ² /rad ²	k_t	$2,667 \times 10^{-7}$ N.m.s ² /rad ²

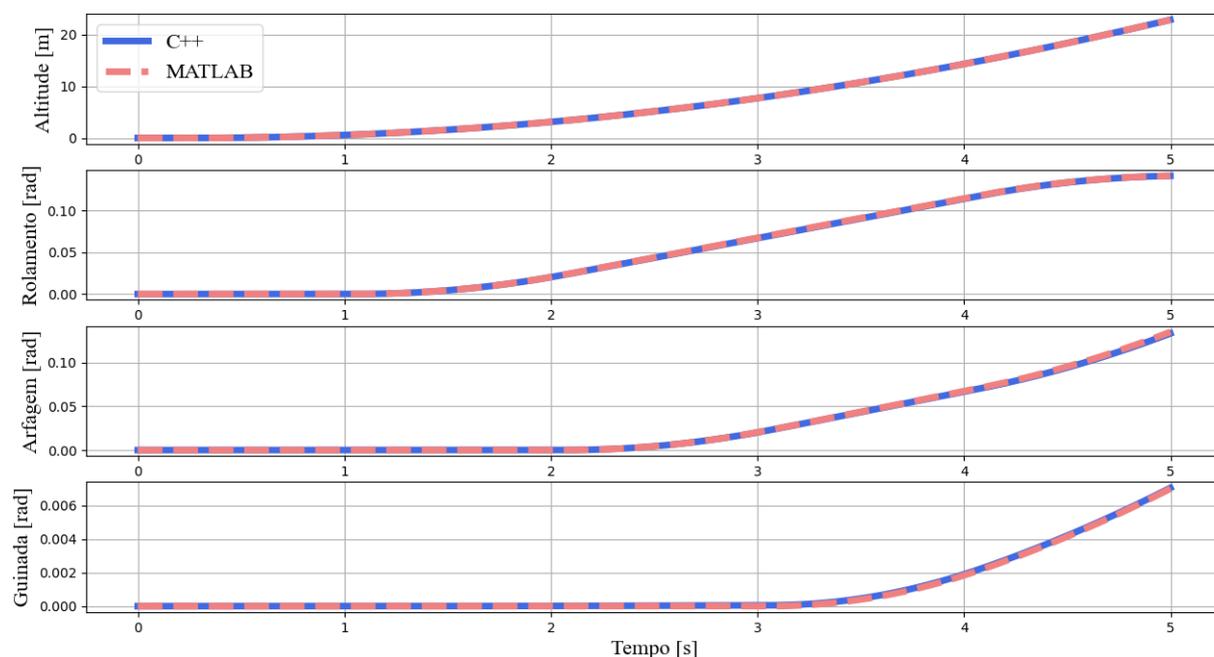
como o modelo de referência não propõe termo de arrasto, para este ensaio a força de arrasto será considerada nula.

Para validar a modelagem dinâmica do quadricóptero, foram realizados cinco ensaios em sequência variando os valores de tensão aplicados aos rotores. Os ensaios são separados por efeitos observados nas variáveis de estado e foram realizados com um passo de integração de 1 milissegundo. Para garantir a consistência da comparação, o método de integração utilizado no modelo de referência foi o de Runge-Kutta de quarta ordem, o mesmo utilizado no modelo proposto. No ensaio, foram aplicados sinais de tensão em degrau por 1 segundo cada, conforme apresentado na Tabela 11, e as respostas obtidas no ambiente de simulação proposto (em azul na Figura 17) foram comparadas com os das simulações realizadas usando o modelo de referência (em vermelho).

Tabela 11: Degraus de tensão aplicados aos ensaios de modelagem independentes.

	Instante [s]	Motor 1 [V]	Motor 2 [V]	Motor 3 [V]	Motor 4 [V]
Altitude	0,000	6,000	6,000	6,000	6,000
Rolamento	1,000	6,000	6,001	6,000	5,999
Arfagem	2,000	5,999	6,000	6,001	6,000
Guinada	3,000	5,999	6,001	5,999	6,001
Múltiplo	4,000	5,998	5,999	6,000	6,001

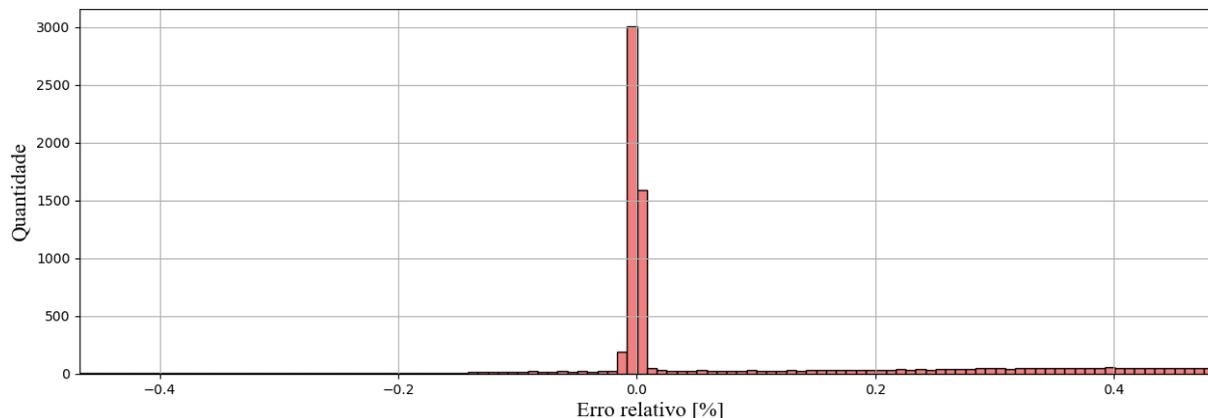
Figura 17: Ensaios de modelagem para variações de estado independentes.



A fim de quantificar a discrepância entre o modelo desenvolvido e o ambiente de referência, foi calculada a diferença entre as respostas obtidas em cada simulação. Os erros relativos obtidos foram então plotados no histograma apresentado na Figura 18.

Observa-se que a maior parte dos dados possui erros muito próximos de zero, representando um erro relativo de $1,114 \times 10^{-2} \% \pm 4,554 \times 10^{-1} \%$, indicando uma

Figura 18: Histograma dos erros relativos do modelo proposto com relação ao modelo de referência.



concordância considerada adequada entre o modelo proposto e o ambiente de referência.

6.3 AVALIAÇÃO DO CONTROLADOR

Para a avaliação do controlador, similarmente à validação do modelo, foram considerados cenários de aplicação de referências independentes e simultâneas. Denota-se, também, a necessidade de evitar sinais de saturação positiva (a negativa ainda deve ser considerada pelos rotores não proporcionarem propulsão negativa) no controlador, com o objetivo de minimizar potenciais não linearidades durante a análise. O tempo de acomodação escolhido deve representar o tempo suficiente para que o quadricóptero consiga reagir a um obstáculo detectado evitando sua colisão. Assim, considera-se o valor de 500 milissegundos de acomodação suficiente. O coeficiente de arrasto para este ensaio também será negligenciado, dado que o objetivo é avaliar apenas o modelo formulado, e não as perturbações envolvidas.

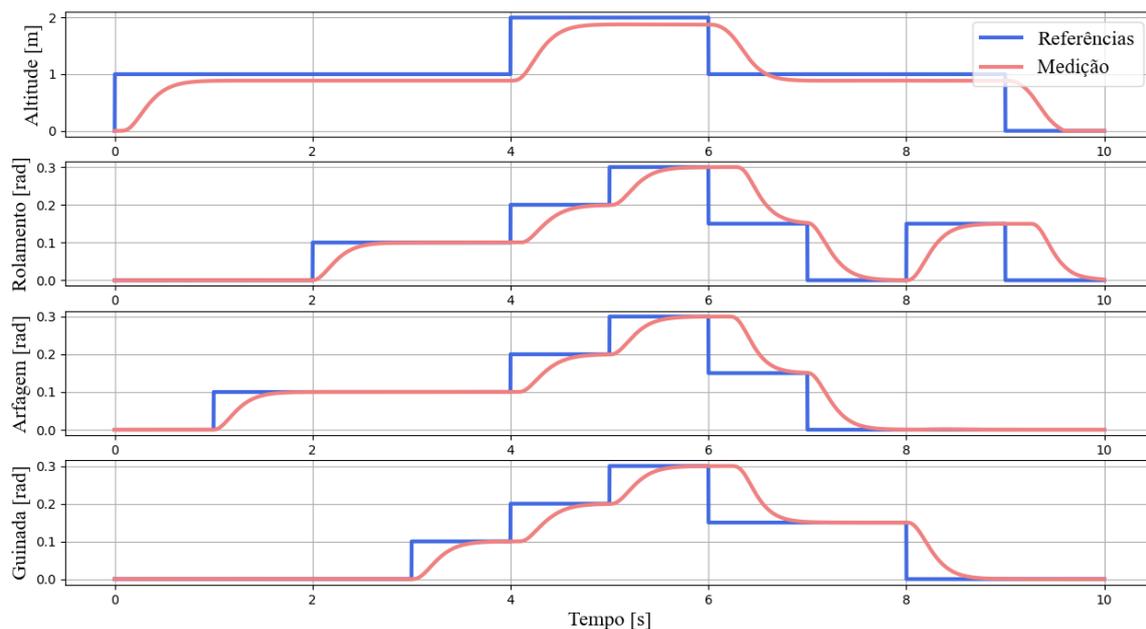
6.3.1 Controlador PD em cascata

Os valores resultantes do cálculo dos parâmetros do controlador PD em cascata, realizado por meio da Tabela 3, são exibidos na Tabela 12 para a aplicação de 500 milissegundos. Na Figura 19, são apresentados os resultados dos ensaios.

Tabela 12: Parâmetros para o controlador PD em cascata.

	Posição		Velocidade	
	K_p	K_d	K_p	K_d
Altitude	$8,000 \text{ s}^{-1}$	1,000	$8,184 \times 10^1 \text{ kg/s}$	$6,220 \times 10^{-1} \text{ kg}$
Rolamento	$8,000 \text{ s}^{-1}$	1,000	$7,600 \times 10^{-2} \text{ kg.m}^2/\text{s}$	$5,776 \times 10^{-3} \text{ kg.m}^2$
Arfagem	$8,000 \text{ s}^{-1}$	1,000	$7,600 \times 10^{-2} \text{ kg.m}^2/\text{s}$	$5,776 \times 10^{-3} \text{ kg.m}^2$
Guinada	$8,000 \text{ s}^{-1}$	1,000	$1,486 \times 10^{-1} \text{ kg.m}^2/\text{s}$	$1,129 \times 10^{-2} \text{ kg.m}^2$

Figura 19: Ensaio de controle para controlador PD em cascata.



Observou-se no ensaio que houve uma discrepância com relação à teoria para o erro de seguimento do controlador, visto que o polo na origem da planta deveria ter sido capaz de integrar o erro da referência. Essa característica pode ser advinda ou da presença de uma perturbação na entrada do sistema, ou a características não modeladas. No caso do modelo aproximado pela matriz Jacobiana, as não linearidades presentes no modelo proposto não são consideradas pelo sinal de controle, de forma que elas também são integradas pela planta. Assim, sem a integração explícita do sinal de erro, o controlador não é capaz de atingir referências com erro nulo, sempre existirá uma parcela de erro constante, que é mais evidente para o caso da altitude. Para a aplicação, este erro é pequeno o suficiente para não invalidar seu uso, visto que as altitudes desejadas são da ordem de dezenas de metros, enquanto o erro de seguimento é da ordem de centímetros.

O ensaio também revelou um comportamento importante do sistema: o efeito da saturação negativa para a propulsão nos instantes 6 s e 9 s, que se manifesta como um atraso na resposta do controlador. Isso ocorre porque, embora o modelo linear exija um valor negativo de propulsão para alcançar a dinâmica desejada, na simulação o sistema é limitado a um valor mínimo de propulsão igual a zero.

Nos demais casos, onde não foram observados erros de seguimento ou atrasos devido à saturação do sinal, o tempo de acomodação foi de $8,435 \times 10^{-1} \pm 2,436 \times 10^{-2}$ s. Essa discrepância com relação ao tempo de assentamento esperado de 500 milissegundos será analisada posteriormente, comparando-a ao outro método.

6.3.2 Linearização por Realimentação de Estados

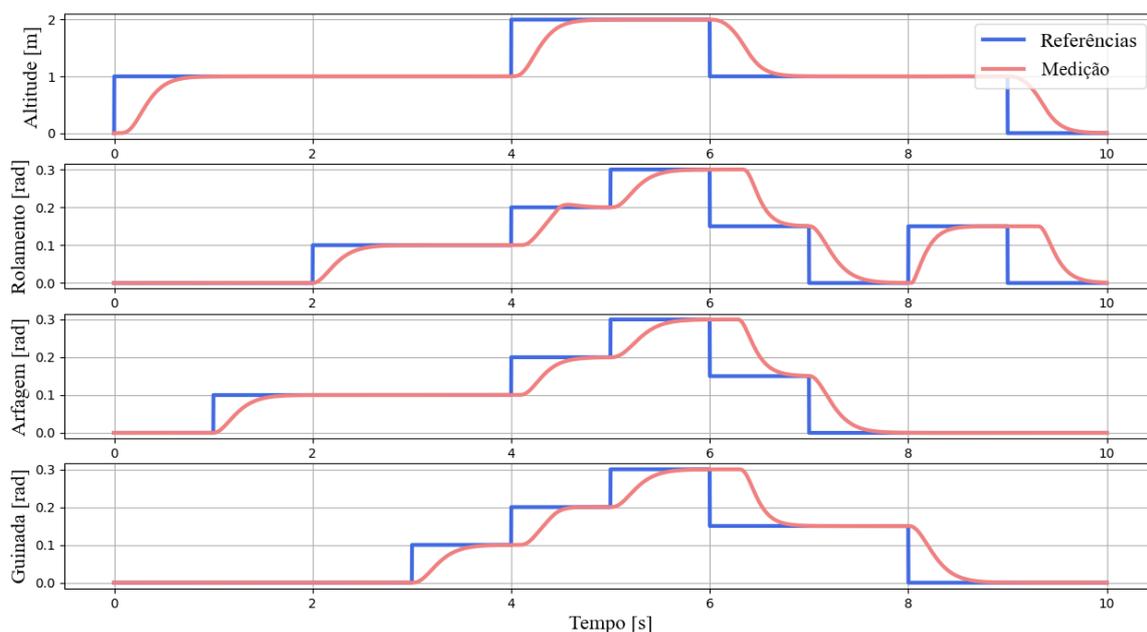
De forma análoga ao controlador PD, inicialmente definem-se os parâmetros dos controladores lineares empregados pelo método para um tempo de acomodação de 500 milissegundos. Os resultados obtidos são apresentados na Tabela 13 e os ensaios ilustrados na Figura 20.

Este novo ensaio, apesar de ainda apresentar a característica de saturação do sinal

Tabela 13: Parâmetros para o controlador linear em cascata da linearização por realimentação de estados.

	Posição		Velocidade
	K_p	K_d	K_p
Altitude	$8,000 \text{ s}^{-1}$	1,000	$8,000 \text{ s}^{-1}$
Rolamento	$8,000 \text{ s}^{-1}$	1,000	$8,000 \text{ s}^{-1}$
Arfagem	$8,000 \text{ s}^{-1}$	1,000	$8,000 \text{ s}^{-1}$
Guinada	$8,000 \text{ s}^{-1}$	1,000	$8,000 \text{ s}^{-1}$

Figura 20: Ensaio de controle para controlador de linearização por realimentação de estados.



para uma propulsão negativa, demonstra a eliminação do desvio do erro de seguimento. Esse resultado é atribuído à linearização por realimentação de estados, que, ao cancelar as não linearidades da planta, permite que o integrador atue conforme o modelo teórico.

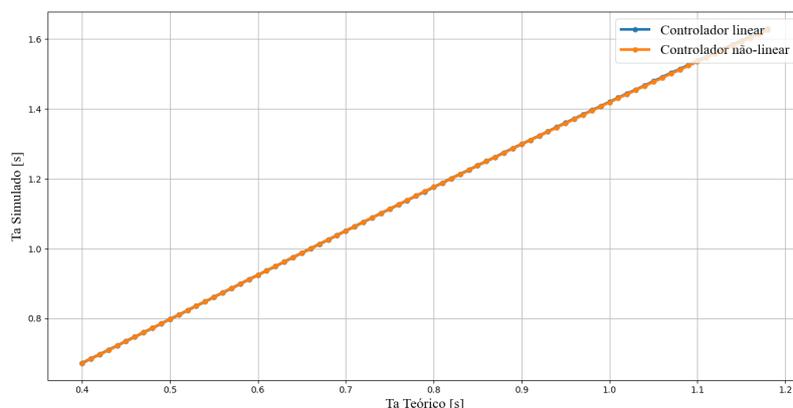
Excluindo os ensaios que apresentaram saturação do sinal, o tempo de acomodação obtido com a linearização por realimentação de estados foi de $8,289 \times 10^{-1} \pm 1,591 \times 10^{-2}$ s. Essa discrepância com relação ao tempo de assentamento esperado de 500 milissegundos será analisada posteriormente, comparando-a ao outro método.

6.3.3 Comparação dos resultados dos controladores

Em ambos os casos, os controladores apresentaram tempos de acomodação semelhantes, no entanto com discrepância em relação aos valores teóricos. A fim de obter uma relação entre o valor esperado e o obtido, e assim ter uma forma de determinar os ganhos necessários do controlador, foi realizada uma bateria de testes para ambos os métodos variando o tempo de assentamento teórico para os controladores e mensurando o simulado. Os resultados destes testes são apresentados na Figura 21.

Observa-se que a relação destes dois termos possui a forma de uma curva linear que não passa pela origem. O valor dessa relação é obtido a partir de uma regressão linear

Figura 21: Relação do tempo de acomodação teórico com o tempo de acomodação simulado.



e possui a forma $T_{as}(T_{at}) = 1,234 \cdot T_{at} + 0,184$ [s], em que T_{as} representa o tempo de acomodação simulado e T_{at} o tempo de acomodação teórico. Essa característica não é estudada em completude no âmbito trabalho, porém identifica-se como hipótese de causa as operações de conversão do sinal após o cálculo do sinal controle, tendo em vista que essas operações podem afetar o tempo de assentamento e ocorrem em ambos os métodos. Para aplicação final, este tempo de acomodação indevido não é um problema, visto que o tempo de assentamento obtido ainda se enquadra na imposição de reação a um obstáculo. Porém, quando projetado, deve-se atentar ao fato de que o tempo de acomodação informado ao controlador é inferior ao aparente.

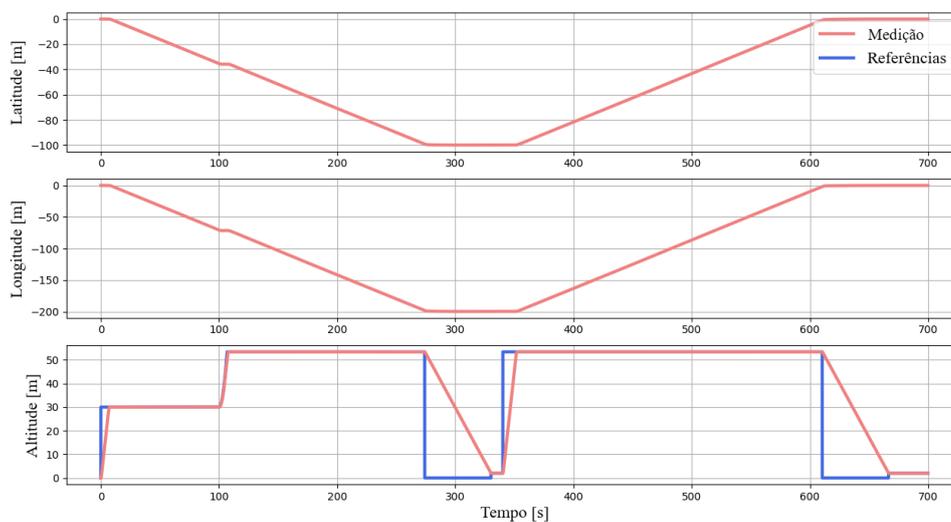
Ainda, é importante destacar que o controlador PD apresentou erro de seguimento, o que pode ser considerado uma desvantagem em relação ao controlador por linearização por realimentação. Assim, determina-se que, para a simulação em ambiente controlado, o método de linearização por realimentação de estados é a melhor opção no caso da simulação com o modelo esperado.

6.4 AVALIAÇÃO DO ALGORITMO

Para analisar o algoritmo, será considerado um ensaio em que o quadricóptero sai da posição $X = 0$ m, $Y = 0$ m e $Z = 0$ m, e deve entregar a encomenda na posição $X = -100$ m, $Y = -200$ m, com um arrasto equivalente (multiplicação de $0,5C_d\rho A_d$) de 1,2 kg/m. Os ajustes utilizados para o ensaio consideram uma altitude de cruzeiro de 30 m, uma arfagem de avanço de 5 graus, uma tolerância de distância para entrega de 2 m, uma tolerância de distância pelo sensor inferior de 2 m e uma tolerância de distância pelo sensor frontal de 3 m. A aplicação do algoritmo é ilustrada na Figura 22. Além disso, restringe-se a velocidade de ascendência e descendência para que o controlador não sature, permitindo um máximo de 5 m/s em ascendência e 1 m/s em descendência.

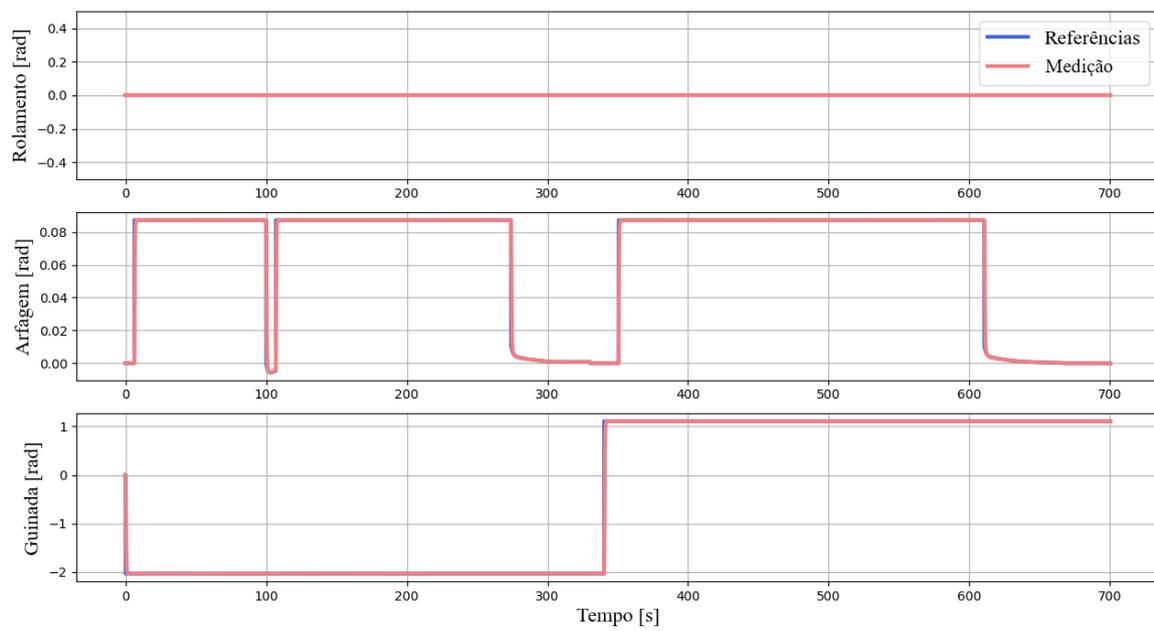
O ensaio mostra a sequência de estados descrita anteriormente. Primeiramente o quadricóptero decola e alcança a altitude de 30 m, efetuando uma guinada de -2,035 rad em direção ao destino. A partir de 6 s, o quadricóptero transiciona para o estado de transporte, mantendo uma arfagem de 5 graus. No instante 100, o quadricóptero detecta um obstáculo e realiza o algoritmo de prevenção de colisões, modificando a altitude de cruzeiro para 53,4 m ao fim do processo. Em seguida, ele continua a tarefa de transporte, que se estende até o instante 274 s, quando o veículo está a uma distância adequada para iniciar a entrega.

Figura 22: Posição do quadricóptero para o ensaio do algoritmo de voo.



Mantendo a posição horizontal (X e Y), o veículo inicia a descida para a entrega, concluindo o pouso no instante 330 s, quando o sensor inferior detecta o solo. Após aguardar 10 segundos para a estabilização do sistema, o *drone* decola novamente, retornando à altitude de cruzeiro. O *drone* então repete o processo em sentido inverso, retornando à base nas coordenadas $X = 0$ m e $Y = 0$ m, onde entra em estado inoperante em 665 s. Na Figura 23, também são apresentados os resultados da orientação do quadricóptero. Conclui-se que o algoritmo foi bem sucedido na missão, com um erro de entrega de 14,43 cm e de retorno à base de 3,50 cm, os quais estão dentro da tolerância de distância determinada de 2 m. Além disso, foi obtido um erro máximo de seguimento do trajeto de 1,2 mm, com quase nenhuma correção de rolamento, visto que o ambiente simulado não possui forças que causem o desvio da rota.

Figura 23: Orientação do quadricóptero para o ensaio do algoritmo de voo.



7 CONCLUSÃO

No âmbito do corrente trabalho foram realizados a modelagem, o projeto de algoritmo de controle e o projeto de algoritmo de voo para um quadricóptero, com foco na aplicação em entregas autônomas. A modelagem matemática do sistema foi realizada por meio de métodos analíticos. A partir do modelo matemático obtido foram projetados controladores os quais foram ajustados utilizando o método de alocação de polos. Os controladores implementados foram o controlador PD, considerando uma simplificação linear do modelo, e o controlador por linearização por realimentação de estados. Em ambos os métodos, foram utilizados controladores PD em cascata para controlar as variáveis odométricas, que, subseqüentemente, controlam as variáveis de velocidade. Após a definição dos controladores, foi elaborada uma sequência de referências para o veículo aéreo, visando a aplicação em entregas autônomas. O modelo de simulação foi implementada em código na linguagem C++.

Os resultados obtidos mostram a eficácia do modelo e dos controladores propostos. A comparação com o modelo de referência revelou que o modelo proposto se diferenciou com uma margem de erro de apenas $1,114 \times 10^{-2} \% \pm 4,554 \times 10^{-1} \%$. Os ensaios com os controladores evidenciaram um melhor desempenho da estratégia de controle por realimentação de estados em comparação ao controlador PD baseado na linearização do modelo, uma vez que não apresentou erro de seguimento. Observou-se, porém, que para ambos os controladores os ganhos do controlador PD não condizem com o tempo de acomodação observado durante a simulação, possuindo um desvio conforme $T_{as}(T_{at}) = 1,234T_{at} + 0,184$ [s], em que T_{as} representa o tempo de acomodação da resposta da simulação e T_{at} o tempo de acomodação teórico. Esse desvio está possivelmente relacionado à cadeia de conversão do sinal, entretanto esta é uma hipótese que deve ser avaliada em trabalhos futuros. Por fim, os testes do algoritmo de voo obtiveram uma resposta considerada adequada, com um erro de posicionamento para a entrega de 14,43 cm e um erro máximo de seguimento do trajeto de 1,2 mm.

É importante enfatizar que os resultados foram obtidos a partir de simulação. Para uma aplicação real, torna-se crucial considerar a robustez do sistema, visto que os métodos implementados não consideram explicitamente incertezas paramétricas, dinâmicas não modeladas ou perturbações externas. A incorporação de técnicas adicionais, como o uso de controladores adaptativos, controle robusto ou técnicas de rejeição de perturbações, seria necessária para garantir a robustez em aplicações práticas. Uma extensão natural deste trabalho seria a implementação prática do sistema e a avaliação de seu desempenho em cenários reais, considerando a presença de perturbações não consideradas nas simulações. Outra possibilidade consiste no aprofundamento do estudo do tempo de assentamento observado na simulação.

REFERÊNCIAS

- A. S. BAZANELLA, J. M. G. *Sistemas de Controle Principios e Métodos de Projeto*. [S.l.]: UFRGS, 2005.
- ALLGOWER, F. IEEE CSS Video Clip Contest 2014 [Technical Activities]. *IEEE Control Systems Magazine*, v. 34, n. 3, p. 18–18, 2014. DOI: 10.1109/MCS.2014.2308623.
- AMAZON. *Amazon Prime Air prepares for drone deliveries*. Jun. 2022a. Disponível em: <<https://www.aboutamazon.com/news/transportation/amazon-prime-air-prepares-for-drone-deliveries>>. Acesso em: 25 jun. 2023.
- AMAZON. *Amazon Stats: Growth, sales, and more*. Mar. 2022b. Disponível em: <<https://sell.amazon.com/blog/amazon-stats>>. Acesso em: 24 jun. 2023.
- ANG, K.; CHONG, G.; LI, Y. PID Control System Analysis, Design, and Technology. *Control Systems Technology, IEEE Transactions on*, v. 13, p. 559–576, ago. 2005. DOI: 10.1109/TCST.2005.847331.
- BATCHELOR, G. *An Introduction to Fluid Dynamics*. [S.l.]: Cambridge University Press, 1967. (Cambridge Mathematical Library). ISBN 9780521663960. Disponível em: <<https://books.google.com.br/books?id=R1a70ihRvUgC>>.
- BENCHOUCHE, W.; MELLAH, R.; BENNOUNA, M. S. The Impact of the dynamic model in feedback linearization trajectory tracking of a mobile robot. *Periodica polytechnica Electrical engineering and computer science*, v. 65, n. 4, p. 329–343, 2021.
- BONNA, R.; CAMINO, J. Trajectory tracking control of a quadrotor using feedback linearization. In: UNIVERSITY OF CAMPINAS SAO PAULO, BRAZIL, 9. INTERNATIONAL Symposium on Dynamic Problems of Mechanics. [S.l.: s.n.], 2015. v. 1.
- BORGES, F. A. P. et al. A neural network-based inversion method of a feedback linearization controller applied to a hydraulic actuator. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Springer, v. 43, n. 5, p. 248, 2021.
- BRYANT, A. Development of an empirical propeller thrust model using experimental measurements and computational fluid dynamics, 2020.
- CRAIG, J. J. *Introduction to robotics : mechanics and control*. 3rd international ed. [S.l.]: Pearson Higher Education, 2014. ISBN 9781292040042; 1292040041. Disponível em: <libgen.li/file.php?md5=a68ae21d06c80f1ff8ad0251f6d113e0>.
- CROCKER, D. *Dictionary of Aviation*. 2. ed. [S.l.]: AC Black, 2010. ISBN 9780713687347; 0713687347.
- D. K. ARROWSMITH BSC PHD, C. M. P. B. *Dynamical Systems: Differential equations, maps and chaotic behaviour*. [S.l.]: Springer Netherlands, 1992.

- DAVID, C. H. *Quadcopter Dynamic Modeling and Simulation (Quad-Sim) v1.00*. 2014. Disponível em: <<https://github.com/dch33/Quad-Sim>>. Acesso em: 29 jul. 2024.
- FERNANDO, H. C. T. E. et al. Modelling, simulation and implementation of a quadrotor UAV. In: 2013 IEEE 8th International Conference on Industrial and Information Systems. [S.l.: s.n.], 2013. P. 207–212. DOI: 10.1109/ICIInfS.2013.6731982.
- FLOREANO, D.; WOOD, R. J. Science, technology and the future of small autonomous drones. *nature*, Nature Publishing Group UK London, v. 521, n. 7553, p. 460–466, 2015.
- GARRIDO, J. et al. Design of Multivariable PID Control Using Iterative Linear Programming and Decoupling. *Electronics*, MDPI, v. 13, n. 4, p. 698, 2024.
- GATTESCHI, V. et al. New Frontiers of Delivery Services Using Drones: A Prototype System Exploiting a Quadcopter for Autonomous Drug Shipments. In: 2015 IEEE 39th Annual Computer Software and Applications Conference. [S.l.: s.n.], 2015. v. 2, p. 920–927. DOI: 10.1109/COMPSAC.2015.52.
- HASHIM, H.; BROWN, L.; MCISAAC, K. Guaranteed Performance of Nonlinear Attitude Filters on the Special Orthogonal Group SO(3). *IEEE Access*, v. 7, p. 3731–3745, jan. 2019. DOI: 10.1109/ACCESS.2018.2889612.
- HEARD, J. B. A. Stability of Numerical Methods for Ordinary Differential Equations. *Numerical Algorithms vol. 31 iss. 1/4*, v. 31, 1/4 2002. DOI: 10.1023/a:1021108006254.
- HIBBELER, R. *Engineering Mechanics: Dynamics*. [S.l.]: Pearson Education, 2011. ISBN 9780133002386. Disponível em: <<https://books.google.com.br/books?id=mTTrAAAAQBAJ>>.
- IFOOD. *Entrega por drones: saiba tudo sobre esse serviço em ascensão*. Fev. 2022. Disponível em: <<https://www.news.ifood.com.br/entrega-por-drones-saiba-tudo-sobre-esse-servico-em-ascensao/>>. Acesso em: 25 jun. 2023.
- IFOOD. *Linha do tempo: conheça a história do iFood*. Abr. 2023. Disponível em: <<https://www.news.ifood.com.br/linha-do-tempo-historia-ifood/>>. Acesso em: 24 jun. 2023.
- JONES, T. *International commercial drone regulation and drone delivery services*. [S.l.], 2017.
- JR., J. D. A. *Fundamentals of Aerodynamics*. 5. ed. [S.l.]: McGraw-Hill Science/Engineering/Math, 2010. ISBN 0073398101; 9780073398105.
- KAREEM, A. A.; OLEIWI, B. K.; MOHAMED, M. J. Planning the Optimal 3D Quadcopter Trajectory Using a Delivery System-Based Hybrid Algorithm. *International Journal of Intelligent Engineering & Systems*, v. 16, n. 2, 2023.
- KOCH, W. et al. Reinforcement Learning for UAV Attitude Control. *ACM Trans. Cyber-Phys. Syst.*, Association for Computing Machinery, New York, NY, USA, v. 3, n. 2, fev. 2019. ISSN 2378-962X. DOI: 10.1145/3301273. Disponível em: <<https://doi.org/10.1145/3301273>>.
- LEE, D.; JIN KIM, H.; SASTRY, S. Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter. *International Journal of control, Automation and systems*, Springer, v. 7, p. 419–428, 2009.

- LEONELLO, R. S. et al. Controle de trajetória do quadrotor AR. Drone 2.0 via abordagens LQR e H_infinity. In: 1. SIMPÓSIO Brasileiro de Automação Inteligente-SBAI. [S.l.: s.n.], 2021. v. 1.
- LI, J.-J. S. W. *Applied Nonlinear Control*. [S.l.: s.n.], 1991. ISBN 0130408905; 9780130408907.
- MOORE, E. F. et al. Gedanken-experiments on sequential machines. *Automata studies*, Princeton, v. 34, p. 129–153, 1956.
- MUZAR, D.; LANTEIGNE, E. Experimental characterization of brushless dc motors and propellers for flight application, 2016.
- NENGSHENG. Research on attitude controller of quadcopter based on cascade PID control algorithm. In: 2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). [S.l.: s.n.], 2017. P. 1493–1497. DOI: 10.1109/ITNEC.2017.8285044.
- NICKOLAS, P. O. F. *AQC - Autonomous Quadcopter Controller*. 2024. Disponível em: <<https://github.com/n1111ck/AQC>>. Acesso em: 29 jul. 2024.
- NIEMIEC, R.; GANDHI, F. A Comparison Between Quadrotor Flight Configurations, 2016.
- OGATA, K. *Modern Control Engineering (5th Edition)*. 5th. [S.l.]: Prentice Hall, 2009. (Prentice-Hall electrical engineering series. Instrumentation and controls series). ISBN 0136156738; 9780136156734. Disponível em: <libgen.li/file.php?md5=862b80f5e6ba8a03add69e3a94dbdab7>.
- POPOOLA, J. J.; OLADEJO, O. J.; ODEYEMI, C. S. Modelling and Simulation of Armature Controlled Direct Current Motor using MATLAB. *Recuperado de http://www.internationaljournalssrg.org/IJEEE/2015/Volume2-Issue3/IJEEE-V2I3P105.pdf*, 2015.
- SHIN, Y. T.; TEH, Y.-K. Design analysis and considerations of power efficient electronic speed controller for small-scale quadcopter unmanned aerial vehicle. In: 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC). [S.l.: s.n.], 2018. P. 773–776. DOI: 10.1109/CCWC.2018.8301770.
- SOBCZYK, M. R. et al. A continuous version of the LuGre friction model applied to the adaptive control of a pneumatic servo system. *Journal of the Franklin Institute*, Elsevier, v. 353, n. 13, p. 3021–3039, 2016.
- SOBCZYK, M. R. Controle em Cascata e a Estrutura Variável com Adaptação de Parâmetros e Compensação de Atrito de um Servoposicionador Pneumático, p. 65–70, ago. 2009.
- THORNTON, S. T.; MARION, J. B. *Dinâmica Clássica de Partículas e Sistemas*. [S.l.: s.n.], 2012. P. 608. ISBN 9788522109067.
- VOOS, H. Nonlinear control of a quadrotor micro-UAV using feedback-linearization. In: 2009 IEEE International Conference on Mechatronics. [S.l.: s.n.], 2009. P. 1–6. DOI: 10.1109/ICMECH.2009.4957154.
- WANG, P. et al. Dynamics modelling and linear control of quadcopter. In: 2016 International Conference on Advanced Mechatronic Systems (ICAMechS). [S.l.: s.n.], 2016. P. 498–503. DOI: 10.1109/ICAMechS.2016.7813499.
- YANO, K. *The Theory of Lie Derivatives and Its Applications*. [S.l.]: North-Holland Publishing Company, 1957. ISBN 9780720421040.

ZIEGLER, J. G.; NICHOLS, N. B. Optimum settings for automatic controllers. *Transactions of the American society of mechanical engineers*, American Society of Mechanical Engineers, v. 64, n. 8, p. 759–765, 1942.

ZIPLINE. *Zipline: Smart Fullfillment*. Disponível em: <<https://www.flyzipline.com/smart-fulfillment>>. Acesso em: 25 jun. 2023.

Apêndices

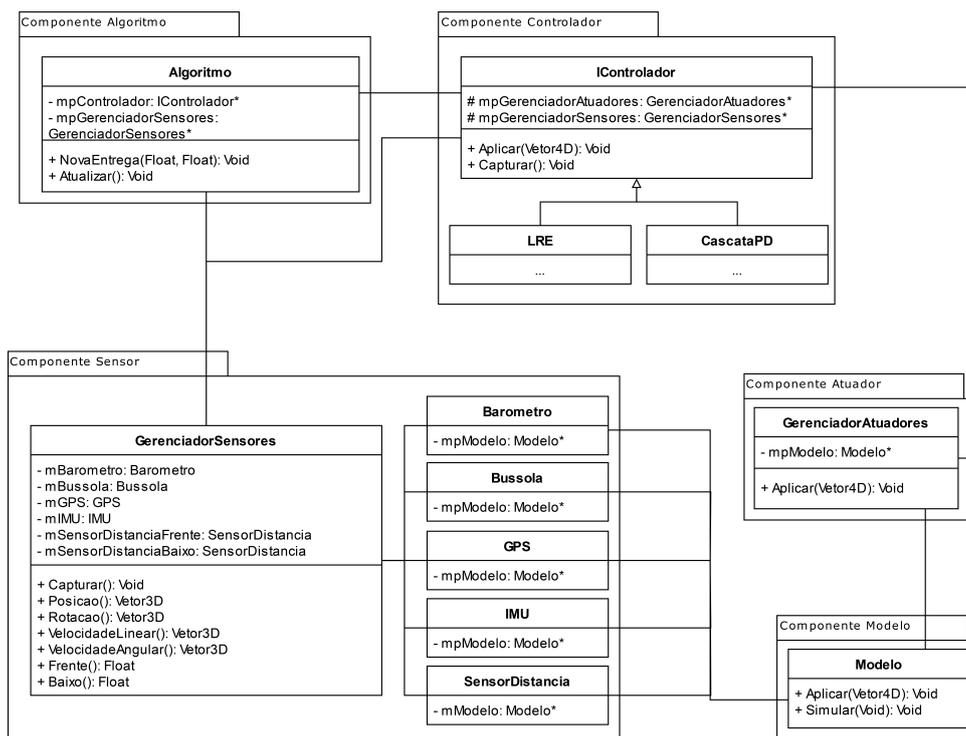
Apêndice A - AMBIENTE DE SIMULAÇÃO

Este capítulo descreve o ambiente de simulação desenvolvido para implementar e validar o sistema de controle. Inicialmente, será apresentado como a modelagem matemática, o controlador projetado e o algoritmo de planejamento de trajetória foram traduzidos em código, detalhando o diagrama de classes, o diagrama de fluxo de dados e o método de integração numérica utilizado. Adicionalmente, será apresentado um ambiente de visualização tridimensional, desenvolvido com o intuito de proporcionar uma representação gráfica intuitiva dos dados gerados pela simulação, facilitando a análise e validação do sistema como um todo.

A.1 DESENVOLVIMENTO

A implementação do sistema utiliza a linguagem C++ com programação orientada a objetos, visando à modularização de cada componente, similar aos diagramas de blocos apresentados em capítulos anteriores. Essa abordagem permite que os componentes de Modelagem, Controle e Algoritmo interajam de maneira adequada e equivalente à aplicação

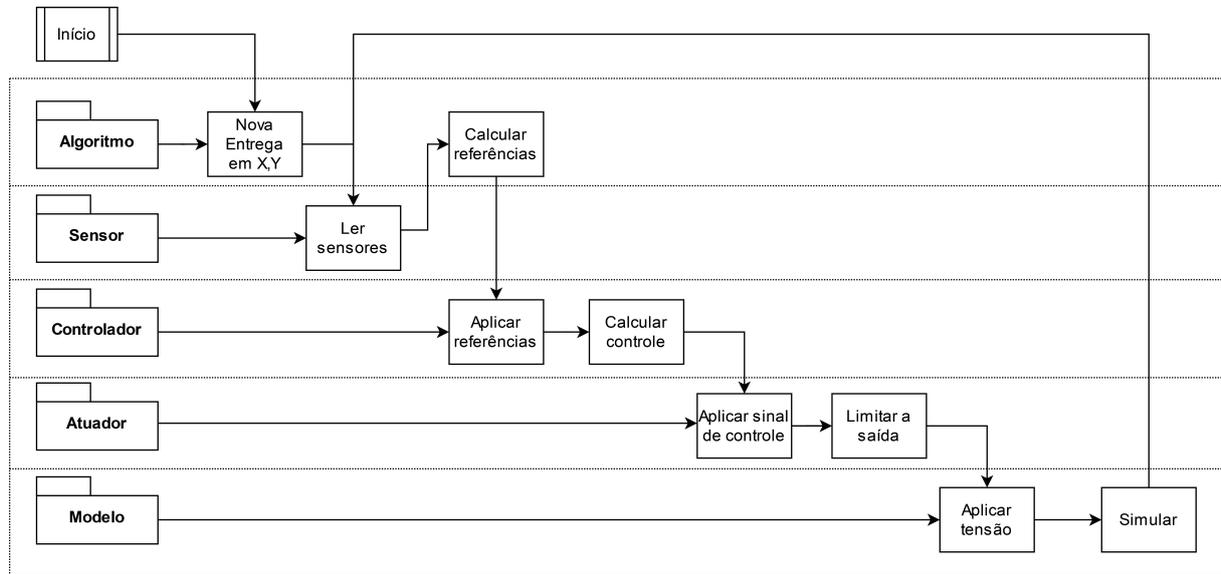
Figura 24: Diagrama UML para a código desenvolvido.



real. Adicionalmente, dois componentes extras, o Atuador e o Sensor, integram o sinal de controle e o sensoriamento dos estados, proporcionando versatilidade para implementação em diferentes plataformas. A interação entre cada um dos componentes e suas respectivas classes é apresentada no diagrama UML (*Unified Modeling Language*) da Figura 24. Denota-se que, neste caso, foram evidenciadas apenas as partes mais importantes ao entendimento do código.

Além da estrutura de cada classe, é fundamental compreender o fluxo de dados, desde o sinal de referência até sua recepção pelo modelo. Esse fluxo pode se originar de três componentes distintos: do Atuador para o Modelo, representando sinais de tensão aplicados aos atuadores em malha aberta; do Controlador para o Modelo, representando sinais de referência a serem controlados, com realimentação proveniente do componente Sensor, em malha fechada; e do Algoritmo para o Modelo, representando a manipulação das referências do Controlador pelo Algoritmo. A Figura 25 ilustra esse comportamento, indicando os métodos e classes correspondentes.

Figura 25: Diagrama de fluxo de dados para a código desenvolvido.



A simulação do modelo emprega o método de integração explícito de Runge-Kutta de quarta ordem (HEARD, 2002). A escolha por este método se baseia em sua boa precisão de integração e na facilidade de implementação em código para sistemas multivariáveis. A Equação 47 demonstra o processo iterativo do algoritmo na simulação a exemplo da rotação angular dos rotores e velocidade linear de altitude, com k 's representando as constantes do método e Δt o passo de integração, e $\dot{\omega}$ e \ddot{z} representando as equações dinâmicas como funções. Assim, o método deve sempre calcular todos os k 's de mesmo índice para só então prosseguir ao próximo k até chegar no quarto índice.

$$\begin{aligned}
 k_{\omega_j 1} &= \Delta t \dot{\omega}(V_j, \omega_j) & k_{z1} &= \Delta t \ddot{z}(\omega_j) \\
 k_{\omega_j 2} &= \Delta t \dot{\omega}(V_j, \omega_j + 0, 5k_{\omega_j 1}) & k_{z2} &= \Delta t \ddot{z}(\omega_j + 0, 5k_{\omega_j 1}) \\
 k_{\omega_j 3} &= \Delta t \dot{\omega}(V_j, \omega_j + 0, 5k_{\omega_j 2}) & k_{z3} &= \Delta t \ddot{z}(\omega_j + 0, 5k_{\omega_j 2}) \\
 k_{\omega_j 4} &= \Delta t \dot{\omega}(V_j, \omega_j + 0, 5k_{\omega_j 3}) & k_{z4} &= \Delta t \ddot{z}(\omega_j + 0, 5k_{\omega_j 3})
 \end{aligned} \tag{47}$$

Ao final do cálculo de todas as constantes, o valor simulado para a variável é encontrado como uma combinação de todas estas, conforme exemplificado na Equação 48.

$$\begin{aligned}\omega_{j1} &= \omega_{j0} + \frac{1}{6}(k_{\omega_{j1}} + 2k_{\omega_{j2}} + 2k_{\omega_{j3}} + k_{\omega_{j4}}) \\ \dot{z}_1 &= \dot{z}_0 + \frac{1}{6}(k_{z1} + 2k_{z2} + 2k_{z3} + k_{z4})\end{aligned}\quad (48)$$

Ainda, considerando que os controladores calculados apresentam características derivativas, torna-se necessário realizar uma aproximação por meio de uma equação de diferenças. Essa abordagem, em conjunto com a frequência de controle, assemelha-se a uma derivação contínua sujeita a um filtro passa-baixas. Sua implementação é descrita pela Equação 49, com u representando a variável de saída, e a variável de entrada, f a frequência de controle e d a frequência do filtro derivativo.

$$\begin{aligned}U(s) &= k_d s \frac{d}{s + d} E(s) \\ U(s) &= k_d s E(s) - d^{-1} s U(s) \\ u(t) &= k_d \dot{e}(t) - d^{-1} \dot{u}(t) \\ u[j] &= k_d f (e[j] - e[j - 1]) - f d^{-1} (u[j] - u[j - 1]) \\ u[j] &= (1 + f d^{-1}) (k_d f (e[j] - e[j - 1]) - u[j - 1] f d^{-1})\end{aligned}\quad (49)$$

O código desenvolvido está disponível no repositório denominado AQC (NICKOLAS, 2024) no Github. A implementação é realizada em formato de uma biblioteca, o que confere versatilidade ao projeto, permitindo sua utilização em diferentes plataformas.

A.2 VISUALIZAÇÃO TRIDIMENSIONAL

Além da implementação da biblioteca *standalone*, utiliza-se também a ferramenta de simulação gráfica Unreal Engine, que permite a integração com o projeto, garantindo a interação do usuário com o quadricóptero e o ambiente virtual através de comandos pelo computador. Na Figura 26, é ilustrado este ambiente de simulação. O projeto também está disponível no repositório AQC.

Figura 26: Visualização gráfica da simulação do quadricóptero no ambiente Unreal Engine.

