



FEB
FACULDADE DE
ENGENHARIA BAURU

FERNANDO MASCAGNA BITTENCOURT LIMA

CONTROLE DE ESTABILIZAÇÃO DE VOO DE VANTS
QUADRIRROTORES: MODELAGEM, CONTROLE,
SIMULAÇÃO E PROTOTIPAGEM

Bauru
2020

FERNANDO MASCAGNA BITTENCOURT LIMA

CONTROLE DE ESTABILIZAÇÃO DE VOO DE VANTS QUADRIRROTORES:
MODELAGEM, CONTROLE, SIMULAÇÃO E PROTOTIPAGEM

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, da Faculdade de Engenharia da UNESP, Universidade Estadual Paulista “Júlio de Mesquita Filho”, Campus de Bauru, como requisito para obtenção do título de Mestre em Engenharia Elétrica.

Área de Concentração: Robótica

Linha de Pesquisa: Mecatrônica

Orientador: Prof. Dr. Átila Madureira Bueno

Coorientador: Prof. Dr. Paulo Sérgio da Silva

Bauru
2020

Lima, Fernando Mascagna Bittencourt.
Controle de estabilização de voo de VANTs
quadrirrotores: modelagem, controle, simulação e
prototipagem / Fernando Mascagna Bittencourt Lima,
2020

198 f. : il.

Orientador: Átila Madureira Bueno
Coorientador: Paulo Sérgio da Silva

Dissertação (Mestrado)-Universidade Estadual
Paulista. Faculdade de Engenharia, Bauru, 2020

1. Quadrirrotor. 2. Modelagem. 3. Controle. 4.
Prototipagem. I. Universidade Estadual Paulista.
Faculdade de Engenharia. II. Controle de estabilização
de voo de VANTs quadrirrotores: modelagem, controle,
simulação e prototipagem.

ATA DA DEFESA PÚBLICA DA DISSERTAÇÃO DE MESTRADO DE FERNANDO MASCAGNA BITTENCOURT LIMA, DISCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA, DA FACULDADE DE ENGENHARIA - CÂMPUS DE BAURU.

Aos 01 dias do mês de junho do ano de 2020, às 09:00 horas, no(a) Via sistemas de videoconferência e outras ferramentas para comunicação a distância, reuniu-se a Comissão Examinadora da Defesa Pública, composta pelos seguintes membros: Prof. Dr. ATILA MADUREIRA BUENO - Orientador(a) do(a) Departamento de Engenharia de Controle e Automação / Instituto de Ciência e Tecnologia - UNESP - Câmpus de Sorocaba, Prof. Dr. JÓ UEYAMA do(a) Instituto de Ciências Matemáticas e de Computação / Universidade de São Paulo / USP, Prof. Dr. DIEGO PAOLO FERRUZZO CORREA do(a) Centro de Engenharia, Modelagem e Ciências Sociais Aplicadas / Universidade Federal do ABC / UFABC, sob a presidência do primeiro, a fim de proceder a arguição pública da DISSERTAÇÃO DE MESTRADO de FERNANDO MASCAGNA BITTENCOURT LIMA, intitulada **CONTROLE DE ESTABILIZAÇÃO DE VOO DE VANTS QUADRIRROTORES: MODELAGEM, CONTROLE, SIMULAÇÃO E PROTOTIPAGEM**. Após a exposição, o discente foi arguido oralmente pelos membros da Comissão Examinadora, tendo recebido o conceito final: Aprovado. Nada mais havendo, foi lavrada a presente ata, que após lida e aprovada, foi assinada pelos membros da Comissão Examinadora.

Prof. Dr. ATILA MADUREIRA BUENO

Prof. Dr. JÓ UEYAMA

Prof. Dr. DIEGO PAOLO FERRUZZO CORREA

A Deus, por ter sido minha fortaleza e ter me guiado para poder chegar até aqui. A meu pai, minha mãe, meus irmãos e amigos que sempre me apoiaram durante esta longa e dura jornada.

Agradecimentos

A Deus, por ter me dado saúde e força para superar todas as dificuldades encontradas ao longo do caminho.

Aos meus pais, Benedito e Janete, e aos meus irmãos, Felipe e Fábio, que sempre me incentivaram, apoiaram e me ajudaram na medida do possível.

Aos meus amigos, que me ajudaram a solucionar problemas encontrados neste projeto.

Aos professores Átila e Paulo Sérgio, que me ensinaram, motivaram e orientaram ao longo deste projeto.

Aos professores que tive, pelos ensinamentos técnicos e de vida.

"Prefiram a minha instrução à prata, e o conhecimento ao ouro puro, pois a sabedoria é mais preciosa do que rubis; nada do que vocês possam desejar compara-se a ela".

(Provérbios 8: 10 - 11)

Resumo

O controle de estabilização do voo de VANTs (veículos aéreos não tripulados) quadrrrotores abrange diversas áreas do conhecimento da engenharia, tais como mecânica, eletrônica e computação. O objetivo deste trabalho é o desenvolvimento de diferentes controladores para tal estabilização. Diferentes controladores (lineares e não lineares) foram desenvolvidos e avaliados a fim de estabelecer um comparativo entre os resultados obtidos para cada um e determinar qual deles é o mais eficiente para diferentes condições de voo. Para tal desenvolvimento, um modelo matemático do VANT em questão foi obtido a fim de possibilitar o projeto dos controladores. Este modelo utilizou os parâmetros de um quadrrrotoz real que também foi desenvolvido neste projeto para validação da eficiência dos controladores desenvolvidos. O protótipo foi projetado de tal forma a possuir uma autonomia de voo suficiente para realização dos testes, resistência mecânica a todos as condições que pode ser exposto durante o voo, tais como vento e vibrações geradas, e ser controlável. Para a validação da eficiência dos controladores desenvolvidos, foram realizadas tanto simulações quanto testes aplicados ao protótipo fixado a uma estrutura giroscópica de testes. Para a realização das simulações, a planta a ser controlada foi o modelo encontrado anteriormente. Os resultados das simulações mostraram que todos os controladores desenvolvidos são capazes de estabilizar o voo do quadrrrotoz. Já os testes aplicados ao protótipo sofreram interferências não modeladas como acoplamento de inércia da estrutura de teste e não linearidades não mapeadas. Isso fez com os resultados encontrados se distanciassem dos resultados obtidos nas simulações.

Palavras-chave: Quadrrrotoz, Modelagem, Controle, Prototipagem.

Abstract

The flight stabilization control of quadrotor UAVs (unmanned aerial vehicles) covers several areas of engineering knowledge, such as mechanics, electronics and computing science. The objective of this work is the development of different controllers for such stabilization. Different controllers (linear and non-linear) were developed and evaluated in order to establish a comparison between the results obtained for each one and to determine which one is the most efficient for different flight conditions. For such development, a mathematical model of the UAV in question was obtained in order to enable the design of the controllers. This model used the parameters of a real quadrirotor that was also developed in this project to validate the efficiency of the developed controllers. The prototype was designed to have sufficient flight autonomy to carry out the tests, mechanical resistance to all conditions that can be exposed during the flight, such as wind and generated vibrations, and be controllable. In order to validate the efficiency of the developed controllers, simulations and tests were applied to the prototype attached to a gyroscopic test structure. To perform the simulations, the plant to be controlled was the model previously found. The results of the simulations showed that all the controllers developed are capable of stabilizing the flight of the quadrotor. The tests applied to the prototype, on the other hand, suffered non-modeled interferences such as inertia coupling of the test structure and non-mapped non-linearities. This caused the results found to differ from the results found in the simulations.

Keywords: Quadrotor, Modeling, Control, Prototyping.

Lista de Figuras

Figura 1.1 – Graus de Liberdade de um VANT.....	1
Figura 1.2 – VANT Multirrotor DJI S1000 em voo.	2
Figura 1.3 – VANT de Asa Fixa XMobots Echar 20D em voo.....	3
Figura 1.4 – VANT Helicóptero Monorrotor em voo.	3
Figura 1.5 – VANT Híbrido Amazon Prime Air em voo.	4
Figura 1.6 – Quadrirrotor em voo.	5
Figura 1.7 – Representação do sentido de rotações e forças de um quadrirrotor com movimentação em cruz.	6
Figura 2.1 – Balões bombardeando Veneza, 1849.	10
Figura 2.2 – Quadrirrotor <i>Gyrosplane No. 1</i> , 1907.....	10
Figura 2.3 – Torpedo aéreo, 1917.....	11
Figura 2.4 – <i>Flying Octopus</i> , 1922.	12
Figura 2.5 – Aeronave F5-L rádio controlada em voo, 1924.	12
Figura 2.6 – Quadrirrotor desenvolvido por Nicolas Florine em voo, 1933.....	13
Figura 2.7 – Aeronave TDR-1 em voo, 1944.	13
Figura 2.8 – Arquitetura de um sistema de controle em cascata de um quadrirrotor.	14
Figura 2.9 – <i>Loop</i> de controle do motor mais interno, <i>loop</i> de controle de atitude intermediário e <i>loop</i> de controle de posição externa.....	15
Figura 2.10 – Sistema simulado para cada controlador aplicado ao ângulo φ extraído do trabalho de Alves (2012).	18
Figura 2.11 – Bancada de testes utilizada no trabalho de Esteves, Moutinho e Azinheira (2015).	18

Figura 2.12 – Resultados em bancada de testes extraídos do trabalho de Esteves, Moutinho e Azinheira (2015).	19
Figura 2.13 – Resultados em voo livre extraídos do trabalho de Esteves, Moutinho e Azinheira (2015).	20
Figura 2.14 – Simulação de um controlador de atitude aplicado a um mini quadricóptero extraído do trabalho de Zhongli <i>et al.</i> (2017).	20
Figura 3.1 – Diagrama de blocos do projeto eletrônico.	23
Figura 3.2 – PocketBeagle Board.	25
Figura 3.3 – IMU Pololu AltIMU-10 v4.	25
Figura 3.4 – Sensor ultrassônico HC-SR04.	26
Figura 3.5 – Regulador de tensão LM2596S.	27
Figura 3.6 – Bateria de LiPo 3S 15C 5200 mAh da TATTU.	28
Figura 3.7 – Motor <i>brushless</i> XXD A2212/13T.	30
Figura 3.8 – Eficiência energética versus corrente aplicada ao motor XXD A2212/13T.	30
Figura 3.9 – ESC HW30A.	31
Figura 3.10 – Esquemático da PCI de Controle.	33
Figura 3.11 – Esquemático da PCI de Potência.	34
Figura 3.12 – Layout da PCI de Controle.	35
Figura 3.13 – PCI de Controle após a corrosão.	36
Figura 3.14 – PCI de Controle sem a SBC.	36
Figura 3.15 – PCI de Controle com a SBC.	37
Figura 3.16 – Layout da PCI de Potência.	38
Figura 3.17 – PCI de Potência após a corrosão.	38
Figura 3.18 – PCI de Potência após a inserção de reforços nas trilhas.	39

Figura 3.19 – PCI de Potência vista de cima.	39
Figura 3.20 – CAD 2D da Estrutura do quadrirrotor.	40
Figura 3.21 – CAD 3D da Estrutura do quadrirrotor.	41
Figura 3.22 – Estrutura do quadrirrotor antes da furação.	41
Figura 3.23 – Pesagem da estrutura do quadrirrotor antes da furação.	42
Figura 3.24 – Conjunto motor XXD A2212/13T + cruzeta de fixação + <i>spinner</i>	43
Figura 3.25 – Pesagem do conjunto motor XXD A2212/13T + cruzeta de fixação + <i>spinner</i> + hélice.	43
Figura 3.26 – CAD 2D da montagem completa do quadrirrotor.	44
Figura 3.27 – CAD 3D da montagem completa do quadrirrotor: vista em perspectiva.	44
Figura 3.28 – CAD 3D da montagem completa do quadrirrotor: vista lateral rotacionada em relação ao eixo z.	45
Figura 3.29 – Quadrirrotor real completamente montado.	45
Figura 3.30 – Diagrama de Classes do software do quadrirrotor.	48
Figura 3.31 – Captura de tela da aplicação do controle remoto.	50
Figura 4.1 – Rotação plana de coordenadas.	54
Figura 4.2 – Ponto em movimento em dois sistemas de referência em movimento. .	57
Figura 4.3 – Estrutura desenvolvida para medição da constante de empuxo.	61
Figura 4.4 – Empuxo gerado pelo conjunto rotor-hélice vs largura de pulso aplicada.	61
Figura 4.5 – Corrente drenada pelo motor vs largura de pulso aplicado.	62
Figura 4.6 – Estrutura desenvolvida para medição da constante de torque.	63
Figura 4.7 – Torque gerado pelo conjunto rotor-hélice vs largura de pulso aplicado.	64
Figura 4.8 – Esquema de variáveis de estado de um quadrirrotor.	65

Figura 5.1 – Controlador PID aplicado a um sistema de segunda ordem.....	70
Figura 5.2 – Controlador PID utilizado para o ajuste dos ganhos.....	72
Figura 5.3 – Ferramenta <i>Signal Constraint</i> aplicada ao ajuste dos ganhos do controlador PID aplicado ao controle de φ	73
Figura 5.4 – Ganhos do controlador PID aplicado a φ vs tensão de entrada.....	74
Figura 5.5 – Controlador PID aplicado ao controle de φ e θ (modelo simplificado)...	75
Figura 5.6 – Interior do bloco controlador PID.	75
Figura 5.7 – Interior do bloco do Decompositor Completo.	76
Figura 5.8 – Ganhos do controlador LQR aplicado a φ vs tensão de entrada.	79
Figura 5.9 – Controlador LQR aplicado ao controle de φ e θ (modelo simplificado).	79
Figura 5.10 – Controlador P aplicado ao controle de ψ' (modelo simplificado).....	81
Figura 5.11 – Controlador <i>Backstepping</i> utilizado no <i>Simulink</i> para ajuste de ganhos.	84
Figura 5.12– Ganhos do controlador <i>Backstepping</i> aplicado a φ vs tensão de entrada.	85
Figura 5.13 - <i>Backstepping</i> aplicado ao controle de φ e θ (modelo simplificado).....	86
Figura 5.14 - Interior do bloco do controlador <i>Backstepping</i>	87
Figura 6.1 – Sistema de controle digital.....	89
Figura 6.2 – CAD 3D da estrutura de teste giroscópica.....	94
Figura 6.3 – CAD 3D da estrutura de teste giroscópica com o quadrirrotor acoplado.	94
Figura 6.4 – Estrutura de teste giroscópica com o quadrirrotor acoplado.....	95
Figura 7.1 – Modelo Não Linear utilizado nas simulações.	97
Figura 7.2 – Bloco gerador de torques.	98
Figura 7.3 – Bloco “Modelo” para simulação não linear.....	99

Figura 7.4 – Bloco “Modelo_Posicao_Inercial” para simulação não linear.	100
Figura 7.5 – Simulação 1: Controladores PID's Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).....	102
Figura 7.6 – Simulação 1: Saídas δ_1 a δ_4 dos Controladores PID's Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).	102
Figura 7.7 – Simulação 2: Controladores PID's Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).....	103
Figura 7.8 – Simulação 2: Saídas δ_1 a δ_4 dos Controladores PID's Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).	103
Figura 7.9 – Diagrama de Blocos utilizados para simulações aplicadas ao modelo não linear.	105
Figura 7.10 – Diagrama de Blocos utilizados para simulação dos controladores PID's aplicados ao modelo não linear.	106
Figura 7.11 – Simulação 1: Controladores Contínuos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	107
Figura 7.12 – Simulação 1: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos PID's aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	107
Figura 7.13 – Simulação 2: Controladores Contínuos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	108
Figura 7.14 – Simulação 2: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos PID's aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	108
Figura 7.15 – Simulação 3: Controladores Contínuos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	109
Figura 7.16 – Simulação 3: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos PID's aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	109

Figura 7.17 – Simulação 4: Controladores Contínuos PIDs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear) (zoom em φ e θ).....	110
Figura 7.18 – Simulação 4: Controladores Contínuos PIDs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	110
Figura 7.19 – Simulação 4: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos PIDs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	111
Figura 7.20 – Planejador de voo acoplado ao sistema.	112
Figura 7.21 – Resultado da simulação de voo utilizando os controladores PIDs e P.	112
Figura 7.22 – Ângulos de Referência e entrada na simulação de voo utilizando os Controladores Contínuos PIDs e P.	113
Figura 7.23 – Ângulos de Referência e entrada na simulação de voo utilizando os Controladores Contínuos PIDs e P (zoom).	113
Figura 7.24 – Simulação 1: Controladores LQRs Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).	115
Figura 7.25 – Simulação 1: Saídas δ_1 a δ_4 dos Controladores LQRs Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).	116
Figura 7.26 – Simulação 2: Controladores LQRs Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).	116
Figura 7.27 – Simulação 2: Saídas δ_1 a δ_4 dos Controladores LQRs Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).	117
Figura 7.28 – Diagrama de Blocos utilizados para simulação dos controladores LQRs aplicados ao modelo não linear.	119
Figura 7.29 – Bloco controlador LQR utilizado para simulação.	119

Figura 7.30 – Simulação 1: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	120
Figura 7.31 – Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos LQRs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	120
Figura 7.32 – Simulação 2: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	121
Figura 7.33 – Simulação 2: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos LQRs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	121
Figura 7.34 – Simulação 3: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	122
Figura 7.35 – Simulação 3: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos LQRs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	122
Figura 7.36 – Simulação 4: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear) (zoom em φ e θ).	123
Figura 7.37 – Simulação 4: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	123
Figura 7.38 – Simulação 4: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos LQRs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	124
Figura 7.39 – Resultado da simulação de voo utilizando os controladores LQRs e P.	124
Figura 7.40 – Ângulos de Referência e entrada na simulação de voo utilizando os controladores LQRs e P.	125
Figura 7.41 – Ângulos de Referência e entrada na simulação de voo utilizando os controladores LQRs e P (zoom).	125

Figura 7.42 – Controlador P Contínuo aplicado ao controle de ψ' (modelo linear simplificado).....	127
Figura 7.43 – Controlador P Contínuo aplicado ao controle de ψ' mais ψ (modelo linear simplificado).	128
Figura 7.44 – Saídas $\delta 1$ a $\delta 4$ do Controlador P Contínuo aplicado ao controle de ψ' (modelo linear simplificado).	128
Figura 7.45 – Simulação 1: Controladores Contínuos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo linear simplificado).....	130
Figura 7.46 – Simulação 1: Saídas $\delta 1$ a $\delta 4$ dos Controladores <i>Backsteppings</i> Contínuos aplicados a φ e θ e P aplicado a ψ' (modelo linear simplificado).....	131
Figura 7.47 – Simulação 2: Controladores Contínuos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo linear simplificado).....	131
Figura 7.48 – Simulação 2: Saídas $\delta 1$ a $\delta 4$ dos Controladores <i>Backsteppings</i> Contínuos aplicados a φ e θ e P aplicado a ψ' (modelo linear simplificado).....	132
Figura 7.49 – Diagrama de Blocos utilizados para simulação dos controladores <i>Backstepping</i> aplicados ao modelo não linear.....	134
Figura 7.50 – Simulação 1: Controladores Contínuos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	135
Figura 7.51 – Simulação 1: Comandos $\delta 1$ a $\delta 4$ gerados pelos Controladores Contínuos <i>Backsteppings</i> aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	135
Figura 7.52 – Simulação 2: Controladores Contínuos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	136

Figura 7.53 – Simulação 2: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos <i>Backsteppings</i> aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	136
Figura 7.54 – Simulação 3: Controladores Contínuos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	137
Figura 7.55 – Simulação 3: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos <i>Backsteppings</i> aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	137
Figura 7.56 – Simulação 4: Controladores Contínuos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear) (<i>zoom</i>).	138
Figura 7.57 – Simulação 4: Controladores Contínuos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).	138
Figura 7.58 – Simulação 4: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos <i>Backsteppings</i> aplicados a φ e θ e P aplicado a ψ' (modelo não linear).	139
Figura 7.59 – Resultado da simulação de voo utilizando os controladores <i>Backsteppings</i> e P.	139
Figura 7.60 – Ângulos de Referência e entrada na simulação de voo utilizando os Controladores Contínuos <i>Backstepping</i> e P.	140
Figura 7.61 – Ângulos de Referência e entrada na simulação de voo utilizando os Controladores Contínuos <i>Backstepping</i> e P (<i>zoom</i>).	140
Figura 7.62 – Controladores Discretos PIDs aplicados ao controle de φ e θ e P aplicado ao controle de ψ'	142
Figura 7.63 – Controladores Discretos PIDs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' mais ψ estimado.	143

Figura 7.64 – Controladores discretos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ'	144
Figura 7.65 – Controladores discretos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' mais ψ estimado.....	145
Figura 7.66 – Controladores discretos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ'	146
Figura 7.67 – Controladores discretos <i>Backsteppings</i> aplicados ao controle de φ e θ e P aplicado ao controle de ψ' mais ψ estimado.	146

Lista de Tabelas

Tabela 3.1 – Especificações da SBC PocketBeagle Board.....	24
Tabela 3.2 – Especificações do Regulador de Tensão LM2596S.....	27
Tabela 3.3 – Especificações do Motor <i>Brushless</i> XXD A2212/13T.....	29
Tabela 3.4 – Especificações do ESC HW30A.....	31
Tabela 3.5 – Comando de movimento (5 bytes).....	50
Tabela 3.6 – Comando de decolagem/pouso (2 bytes).....	51
Tabela 3.7 – Mensagem de telemetria (9 bytes).....	51
Tabela 7.1 – Ganhos dos controladores PIDs contínuos.....	104
Tabela 7.2 – Erros quadráticos médios obtidos pelos controladores PIDs contínuos aplicados a φ e θ (modelo linear simplificado).....	104
Tabela 7.3 – Erros quadráticos médios obtidos pelos controladores PIDs contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear).....	114
Tabela 7.4 – Ganhos dos controladores LQRs contínuos.....	117
Tabela 7.5 – Erros quadráticos médios obtidos pelos controladores LQRs contínuos aplicados a φ e θ (modelo linear simplificado).....	117
Tabela 7.6 – Erros quadráticos médios obtidos pelos controladores LQRs contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear).....	126
Tabela 7.7 – Ganhos do Controlador P Contínuo.....	129
Tabela 7.8 – Erro quadrático médio obtido pelo controlador P contínuo aplicado a ψ' (modelo linear simplificado).....	129
Tabela 7.9 – Ganhos dos controladores <i>Backstepping</i> contínuos.....	132
Tabela 7.10 – Erros quadráticos médios obtidos pelos controladores contínuos <i>Backsteppings</i> aplicados a φ e θ e P aplicado a ψ' (modelo linear simplificado).....	133

Tabela 7.11 – Erros quadráticos médios obtidos pelos controladores <i>Backsteppings</i> contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear).....	141
Tabela 7.12 – Erros quadráticos médios para os controladores discretos PIDs aplicados a φ e θ e P aplicado a ψ'	143
Tabela 7.13 – Erros quadráticos médios para os controladores discretos LQRs aplicados a φ e θ e P aplicado a ψ'	145
Tabela 7.14 – Erros quadráticos médios para os controladores discretos <i>Backsteppings</i> aplicados a φ e θ e P aplicado a ψ'	147
Tabela 8.1 – Médias dos erros quadráticos médios obtidos pelos controladores contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo linear).....	149
Tabela 8.2 – Médias dos erros quadráticos médios obtidos pelos controladores contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear) nas simulações 1 a 4.....	151
Tabela 8.3 – Médias dos erros quadráticos médios obtidos pelos controladores discretos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear) nas simulações 1 a 4.....	151

Lista de Abreviaturas e Siglas

2D	Duas Dimensões
3D	Três Dimensões
BEC	<i>Battery Eliminator Circuit</i>
CAD	<i>Computer-Aided Design</i>
DIP	<i>Dual In-line Package</i>
Eagle	Software de automação de projetos eletrônicos da empresa Autodesk
ESC	<i>Electronic Speed Controller</i>
GPS	<i>Global Positioning System</i>
I2C	<i>Inter-Integrated Circuit</i>
IMU	<i>Inertial Measurement Unit</i>
LiPo	Lítio Polímero
LDU	<i>Lower–Upper Decomposition</i>
LQR	<i>Linear Quadratic Regulator</i>
MATLAB	Software de computação numérica da empresa MathWorks
MOSFET	<i>Metal Oxide Semiconductor Field Effect Transistor</i>
MTP	<i>Message Type</i>
P	Proporcional
PCB	<i>Printed Circuit Board</i>
PCI	Placa de Circuito Impresso
PD	Proporcional Derivativo
PI	Proporcional Integral
PID	Proporcional Integral Derivativo
PWM	<i>Pulse Width Modulation</i>

SBC	<i>Single Board Computer</i>
SO	Sistema Operacional
UAV	<i>Unmanned Aerial Vehicle</i>
VANT	Veículo Aéreo Não Tripulado

Lista de Símbolos

b_1	ganho 1 do controlador <i>Backstepping</i>
b_2	ganho 2 do controlador <i>Backstepping</i>
b_3	ganho 3 do controlador <i>Backstepping</i>
b_4	ganho 4 do controlador <i>Backstepping</i>
F^i	sistema de coordenadas inercial
F^v	sistema de coordenadas do veículo
F^b	sistema de coordenadas do corpo
F_i	força de empuxo gerada pelo conjunto rotor-hélice i
$\hat{i}^i$	versor definido em F^i
$\hat{i}^v$	versor definido em F^v
$\hat{i}^b$	versor definido em F^b
$\hat{j}^i$	versor definido em F^i
$\hat{j}^v$	versor definido em F^v
$\hat{j}^b$	versor definido em F^b
$\hat{k}^i$	versor definido em F^i
$\hat{k}^v$	versor definido em F^v
$\hat{k}^b$	versor definido em F^b
K_1	ganho 1 do controlador LQR
K_2	ganho 2 do controlador LQR
K_d	ganho derivativo
K_p	ganho proporcional
K_i	ganho integral
K_F	fator de empuxo

K_{F_0}	constante de empuxo
k_τ	fator de torque
K_{τ_0}	constante de torque
l	comprimento do braço do quadrrrotor
m	massa do quadrrrotor
p	taxa de rolagem medida ao longo de $\hat{i}^b$ em relação a F^i
p_e	posição inercial do quadrrrotor ao longo de $\hat{j}^i$ em F^i
p_n	posição inercial do quadrrrotor ao longo de $\hat{i}^i$ em F^i
p_u	altitude do quadrrrotor medida ao longo de $\hat{k}^i$ em F^i
q	taxa de arfagem medida ao longo de $\hat{j}^b$ em relação a F^i
r	taxa de guinada medida ao longo de $\hat{k}^b$ em relação a F^i
t	tempo
u	velocidade do quadrrrotor medida ao longo de $\hat{i}^b$ em F^b
U	tensão da bateria
v	velocidade do quadrrrotor medida ao longo de $\hat{j}^b$ em F^b
w	velocidade do quadrrrotor medida ao longo de $\hat{k}^b$ em F^b
x	eixo de coordenada paralelo ao rotor 2
y	eixo de coordenada paralelo ao rotor 1
z	eixo de coordenada perpendicular aos eixos x e y
θ	ângulo de arfagem
τ_i	torque gerado pelo conjunto rotor-hélice i
τ_θ	torque no eixo de arfagem
τ_φ	torque no eixo de rolagem
τ_ψ	torque no eixo de guinada

φ ângulo de rolagem

ψ ângulo de guinada

Sumário

1	INTRODUÇÃO.....	1
1.1	Motivação e Objetivos	6
1.2	Organização do Trabalho.....	7
2	REVISÃO BIBLIOGRÁFICA.....	9
2.1	Histórico	9
2.2	Técnicas de Controle	14
2.3	Trabalhos Recentes	16
3	PROJETO DO QUADRIRROTOR.....	23
3.1	Projeto Eletrônico.....	23
3.1.1	SBC	24
3.1.2	IMU	25
3.1.3	Sensor ultrassônico	26
3.1.4	Regulador de tensão.....	26
3.1.5	Bateria	27
3.1.6	Motores	28
3.1.7	ESC	31
3.1.8	Circuito Elétrico	32
3.1.8.1	Circuito de controle	32
3.1.8.2	Circuito de potência	33
3.1.9	PCIs.....	34
3.1.9.1	PCI de Controle.....	35
3.1.9.2	PCI de Potência	37
3.2	Projeto Mecânico	40
3.3	Projeto Computacional.....	46
3.3.1	Software do quadrirrotor.....	46
3.3.2	Software do controle remoto	49
4	MODELAGEM DINÂMICA DO QUADRIRROTOR PROJETADO	53
4.1	Cinemática	53
4.1.1	Sistemas de Coordenadas.....	53
4.1.2	Relação entre os sistemas de coordenadas por meio de ângulos de Euler.....	54

4.1.2.1	Relação entre velocidades angulares definidas em dois referenciais expressos em termos dos ângulos de Euler	55
4.1.3	Relação entre as derivadas de um vetor em dois sistemas de referência	56
4.1.4	Relação entre as velocidades e acelerações em dois sistemas de referência	57
4.2	Dinâmica.....	58
4.2.1	Movimento de Translação.....	58
4.2.2	Movimento de Rotação.....	58
4.3	Forças e Momentos no Quadrrrotor	59
4.3.1	Relação entre força gerada e comando aplicado ao motor	60
4.3.2	Relação entre torque gerado e comando aplicado ao motor	63
4.3.3	Modelagem em função das forças atuantes	64
4.4	Modelo Simplificado.....	66
4.5	Modelo Linearizado	66
5	CONTROLE CONTÍNUO	69
5.1	PID.....	69
5.2	LQR.....	76
5.3	Controle de ψ	80
5.4	<i>Backstepping</i>.....	82
6	CONTROLE DISCRETO	89
6.1	PID.....	90
6.2	LQR.....	91
6.3	Controle de ψ	91
6.4	<i>Backstepping</i>.....	92
6.5	Estrutura de Testes Giroscópica.....	93
7	RESULTADOS	97
7.1	Controladores Contínuos.....	97
7.1.1	PID.....	101
7.1.1.1	Modelo Linear Simplificado	101
7.1.1.2	Modelo Não Linear	105
7.1.2	LQR	115
7.1.2.1	Modelo Linear Simplificado	115
7.1.2.2	Modelo Não Linear	118
7.1.3	Controle de ψ'	126
7.1.4	<i>Backstepping</i>	129

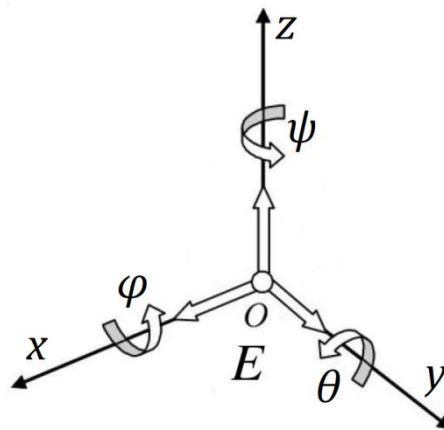
7.1.4.1	Modelo Linear Simplificado	130
7.1.4.2	Modelo Não Linear.....	133
7.2	Controladores Discretos	141
7.2.1	PID.....	142
7.2.2	LQR	144
7.2.3	<i>Backstepping</i>	145
8	CONCLUSÕES	149
	REFERÊNCIAS.....	155
	Apêndice A – Propriedades de massa extraídas do CAD 3D.....	159

1 Introdução

Um VANT (Veículo Aéreo Não Tripulado), do inglês UAV (*Unmanned Aerial Vehicle*), é todo e qualquer tipo de aeronave que pode ser controlada e que não necessita de um piloto embarcado para executar tal ação, podendo ser autônoma ou pilotada remotamente. O estudo das tecnologias relacionadas à VANTs tem sido fortemente impulsionado nos últimos anos devido a forte demanda no mercado, com inúmeras aplicações tais como militares, médicas, agrícolas, monitoramentos, mapeamentos, entregas, filmagens, entre outras.

Segundo Santana e Borges (2009), um VANT possui 6 graus de liberdade: translações nos eixos x , y e z e rotações nos ângulos de rolagem (*roll*) φ , arfagem (*pitch*) θ e guinada (*yaw*) ψ . Tais graus de liberdade podem ser vistos na Figura 1.1. Desta forma, o controle de um VANT é realizado controlando-se estes seis graus de liberdade de forma independente ou não, dependendo de sua arquitetura, através do nível de excitação sobre cada um de seus atuadores.

Figura 1.1 – Graus de Liberdade de um VANT.



Fonte: Adaptado de Santana e Borges (2009).

Um VANT pode possuir diversos tipos de sensores como GPS, magnetômetro, giroscópio, acelerômetro, barômetro, sensor ultrassônico, higrômetro, termômetro, entre outros, que alimentam a malha de controle com informações como posicionamento, direção de deslocamento, ângulos e eixos de rotação, altitude, umidade do ar e temperatura, a fim de determinar como os atuadores devem ser excitados para que se obtenha o comportamento desejado.

Segundo Chapman (2016), a arquitetura de VANT mais popular atualmente é a Multirrotor. Porém, existem outros tipos de arquiteturas tais como a de Asa Fixa, a Helicóptero Monorotor e a de Asa Fixa Híbrida. Cada qual possui suas vantagens e desvantagens em relação às outras. As definições do autor para cada uma destas arquiteturas são apresentadas nos próximos parágrafos.

A arquitetura Multirrotor apresenta como principais vantagens um custo menor e um controle de posicionamento mais preciso em relação às demais arquiteturas, o que a torna eficiente para captura de imagens aéreas. Em contrapartida, apresenta como principais desvantagens o elevado consumo de energia, que a torna ineficiente para percursos de longa distância, e velocidades de deslocamento mais baixas em relação à de Asa Fixa. Um ponto a se notar é que, quanto maior for a carga a ser transportada por este tipo de VANT, maior será o consumo de energia, o que faz com que sua autonomia de voo seja diminuída. Outro ponto é que, quanto maior o número de rotores, maior será o consumo de energia do VANT. A Figura 1.2 mostra um exemplo de VANT com arquitetura Multirrotor em voo.

Figura 1.2 – VANT Multirrotor DJI S1000 em voo.



Fonte: Página mydronelab¹.

Já a arquitetura de Asa Fixa, por possuir asas, necessita do motor mais para se locomover do que para se manter em voo, o que faz com que sua eficiência energética seja muito superior em relação às demais. Apresenta como desvantagens o fato de não ser capaz de se manter estática em um ponto fixo no espaço durante o voo e nem conseguir se transladar isoladamente em todos os eixos de coordenada. A Figura 1.3 mostra um exemplo de VANT com arquitetura Asa Fixa em voo.

¹ Disponível em: <<https://www.mydronelab.com/reviews/dji-s1000.html>> Acesso em: 10 out. 2019.

Figura 1.3 – VANT de Asa Fixa XMobots Echar 20D em voo.



Fonte: Página do fabricante².

Os Helicópteros Monorrotores, embora menos populares entre os VANTs, apresentam como vantagem maior eficiência energética em relação aos multirrotores e a possibilidade de um uso mais facilitado para motores a combustão interna, o que pode aumentar consideravelmente sua autonomia de voo. Porém, em contrapartida, não são tão estáveis quanto os Multirrotores, além de possuírem um custo mais elevado em relação aos mesmos. A Figura 1.4 mostra um exemplo de VANT com arquitetura Helicóptero Monorrotor em voo.

Figura 1.4 – VANT Helicóptero Monorrotor em voo.



Fonte: Chapman (2016).

² Disponível em: <<https://xmobots.com.br/echar-20d/>> Acesso em: 10 out. 2019.

Por fim, um VANT Híbrido, que possui a arquitetura mais recente, apresenta algumas características das arquiteturas anteriores, podendo tanto se locomover como um VANT de Asa Fixa quanto como um Multirrotor, e decolar e pousar verticalmente. Apresenta como desvantagem uma menor autonomia e velocidade em relação à Asa Fixa e menos estabilidade em relação às demais arquiteturas aqui apresentadas. A Figura 1.5 mostra um exemplo de VANT com arquitetura Híbrida em voo.

Figura 1.5 – VANT Híbrido Amazon Prime Air em voo.



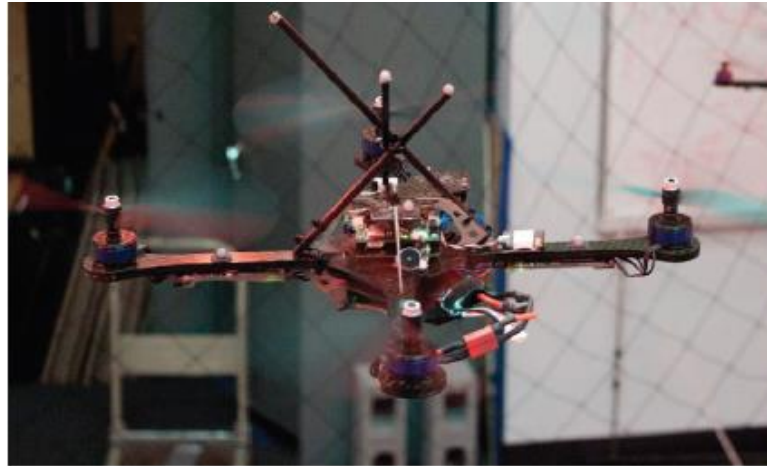
Fonte: Página da Amazon³.

Segundo Mahony, Kumar e Corke (2012), a robótica aérea é uma área que se encontra em um rápido e constante crescimento, assim como a popularidade dos quadricópteros que tem se tornado a arquitetura de VANTs padrão para pesquisa em todo o mundo. Segundo o autor, esta arquitetura suporta voos em locais fechados e ao ar livre e permite a utilização em experimentos para mapeamentos de baixo custo, navegações e estratégias de controle para movimentação tridimensional (3D) de robôs. Ainda segundo o autor, esta mobilidade 3D traz novas possibilidades em relação aos robôs terrestres que se movimentavam através de rodas (maior alvo de estudos da década anterior) como o deslocamento vertical, acrobacias, equilíbrio e giros. Além disso, mais motores podem ser acoplados ao robô transformando-o em um multirrotor pra um determinado uso como um aumento de carga de transporte por exemplo.

³ Disponível em: <<https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011>> Acesso em: 10 out. 2019.

O termo quadricóptero também pode ser empregado, porém o termo quadrirrotor é mais comumente utilizado. Este tipo de arquitetura se consiste basicamente do uso de quatro motores individuais fixos a uma estrutura rígida como pode ser visto na Figura 1.6.

Figura 1.6 – Quadrirrotor em voo.

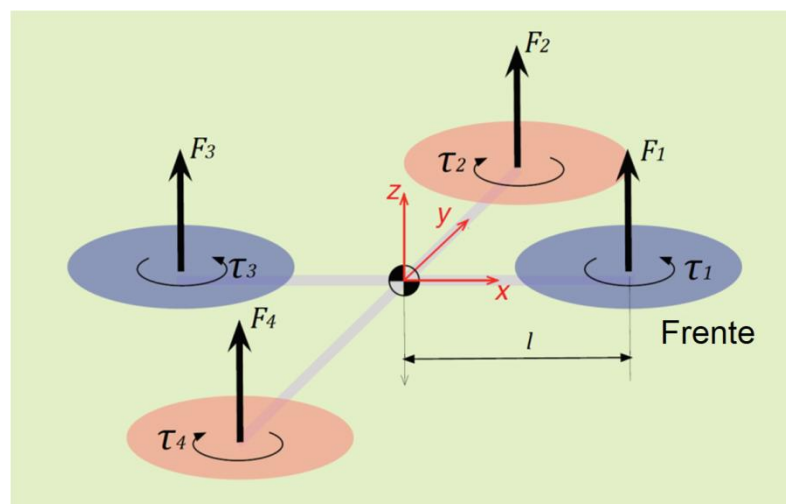


Fonte: Mahony, Kumar e Corke (2012).

O controle de um quadrirrotor, segundo Mahony, Kumar e Corke (2012), é realizado através do controle diferencial do empuxo gerado por seus rotores. Conforme pode ser visto na Figura 1.7, que representa as forças atuantes em um quadrirrotor com movimentação em cruz, o rotor i rotaciona no sentido horário (em relação ao eixo z) enquanto o rotor $i + 1$ rotaciona no sentido anti-horário para permitir o controle e estabilidade da estrutura. Caso contrário, não seria possível anular os torques gerados pelos conjuntos hélice rotor, o que faria com que o controle da estrutura se tornasse inviável.

Para a obtenção de uma rotação no eixo de guinada, basta alterar a velocidade de um par de rotores que rotacionam para o mesmo sentido. Para a obtenção de uma rotação no eixo de arfagem ou de rolagem, basta elevar a velocidade de um rotor e reduzir a de seu par conforme o ângulo e direção desejados, o que faz com uma translação no eixo x ou y ocorra em simultâneo, respectivamente. Já para a obtenção um deslocamento no eixo z , é necessário que os rotores trabalhem em conjunto, variando suas velocidades de rotação de acordo com a translação desejada. Uma combinação simultânea de rotações e translações em todos os eixos também é possível dependendo da necessidade. Ou seja, em um

Figura 1.7 – Representação do sentido de rotações e forças de um quadrirrotor com movimentação em cruz.



Fonte: Adaptado de Mahony, Kumar e Corke (2012).

quadrirrotor existem apenas quatro atuadores (rotores) que são responsáveis pelos deslocamentos em seus seis graus de liberdade. Segundo Santana e Borges (2009), a dinâmica de um quadrirrotor é marcada pela subatuação (seis graus de liberdade e apenas quatro atuadores), pelo grande acoplamento entre os modos e por não linearidades desconhecidas.

1.1 Motivação e Objetivos

Alguns estudos e trabalhos, como os de Beard (2008) e Zu, Zhang e Shan (2017), dedicam-se à determinação de modelos matemáticos abrangentes e úteis aos problemas de simulação e controle de quadrirrotores. Outros, como o de Alves (2012) e Khatoon, Gupta e Das (2014), dedicam-se ao estudo de técnicas para o controle dos mesmos, através do uso de técnicas de controle lineares e não lineares.

Este trabalho se dedica ao desenvolvimento completo de um quadrirrotor, envolvendo tanto o desenvolvimento de um protótipo, quanto a modelagem matemática dinâmica do mesmo, o projeto dos controladores de estabilização de voo e o desenvolvimento do software embarcado. Este trabalho também tem como objetivo ser *open source*, o que permite sua replicação, e a um baixo custo, visto que os materiais utilizados foram selecionados baseando-se em seus custos de mercado e nos requisitos mínimos impostos durante a fase de desenvolvimento. O

quadrrorotor desenvolvido trata-se de um quadrrorotor em formato de cruz com dois pares de rotores que rotacionam em sentidos opostos a fim de eliminar os torques gerados.

Diferentes controladores para a estabilização de voo do quadrrorotor foram desenvolvidos, dois lineares e um não linear, a fim de estabelecer um comparativo entre os resultados obtidos e determinar qual deles é o mais eficiente para cada situação. Os controladores lineares desenvolvidos fizeram uso das técnicas conhecidas como PID (Proporcional Integral Derivativo) (Ogata, 2010) e LQR (*Linear Quadratic Regulator* ou regulador linear quadrático) (Åström e Murray, 2009), enquanto o controlador não linear desenvolvido fez uso da técnica conhecida como *Backstepping* (Zhou e Wen, 2008).

1.2 Organização do Trabalho

Para o desenvolvimento dos controladores desenvolvidos neste trabalho, um modelo matemático do VANT projetado foi obtido a fim de possibilitar o projeto dos controladores. Este modelo utiliza os parâmetros de um protótipo de quadrrorotor real que também foi desenvolvido neste projeto para validação da eficiência dos controladores desenvolvidos. Para isto, foram realizadas tanto simulações quanto testes reais aplicados ao protótipo. Durante as simulações, o sistema controlado foi o modelo encontrado anteriormente. Levando em consideração que o modelo apresenta algumas simplificações, a eficiência dos controladores somente foi verificada após a execução de simulações de voo com a utilização do protótipo fixo a uma estrutura giroscópica também desenvolvida neste projeto especificamente para este fim.

A estrutura deste trabalho encontra-se dividida em oito seções. A primeira seção é uma introdução e apresenta alguns conceitos sobre VANTs e quadrrorotores e a forma na qual este trabalho está dividido. A segunda seção apresenta uma revisão bibliográfica expondo um breve histórico sobre os primeiros VANTs, alguns conceitos sobre técnicas de controle que podem ser aplicadas ao controle de quadrrorotores e alguns trabalhos recentes na área. A terceira seção apresenta o projeto do protótipo desenvolvido, do qual foram extraídos os parâmetros utilizados na modelagem. A quarta seção apresenta a modelagem matemática dinâmica do

sistema, de onde será extraída a planta do sistema a ser controlada. A quinta seção apresenta os projetos dos controladores contínuos desenvolvidos para estabilização de voo da aeronave. A sexta seção apresenta o projeto dos controladores discretos aplicados ao modelo real e a estrutura de testes giroscópica desenvolvida para realização dos testes de eficiência destes controladores aplicados ao modelo real. A sétima seção apresenta os resultados obtidos nas simulações e nos testes realizados no protótipo. Por fim, a oitava seção apresenta as discussões sobre os resultados encontrados, as conclusões deste trabalho e os trabalhos futuros que podem ser realizados para dar sequência a este. As referências bibliográficas e os apêndices se encontram logo abaixo da seção 8.

2 Revisão Bibliográfica

Esta seção apresenta um breve histórico referente ao surgimento e evolução de VANTs e quadrirrotores, conceitos sobre algumas técnicas de controle que podem ser empregadas para o controle de estabilização de voo de quadrirrotores e alguns trabalhos recentes realizados na área. Estes tópicos são abordados nas subseções seguintes.

2.1 Histórico

Segundo Cyprian (2018), o avanço conceitual na construção de VANTs ocorreu no século XIX. Segundo o autor, em 1849 australianos utilizaram balões aerostáticos, feitos de papel e tecido, equipados com uma bomba que continha um mecanismo de fusível com um temporizador, especialmente preparado para uma missão de ataque aéreo à cidade de Veneza. O guiamento destes balões foi realizado única e exclusivamente pela ação do vento. Segundo o autor, este é o primeiro VANT de combate da história. A Figura 2.1 mostra este tipo de VANT sendo utilizado no ataque.

Já o primeiro quadrirrotor da história, segundo Leishman (2000), foi construído em 1907, pelos irmãos franceses Louis e Jacques Breguet, juntamente com o professor Charles Richet. Esta aeronave foi denominada *Gyrosplane No. 1* e pode ser vista na Figura 2.2. Tratava-se de uma aeronave para dois tripulantes e, segundo o autor, foram registrados voos à baixa altitude com duração de poucos segundos com a mesma. Segundo Lima (2015), este quadrirrotor possuía uma massa de cerca de 580 kg, incluindo os dois tripulantes, e contribuiu para com a teoria dos pares de rotores que rotacionam em sentidos opostos, a fim de anular os torques gerados.

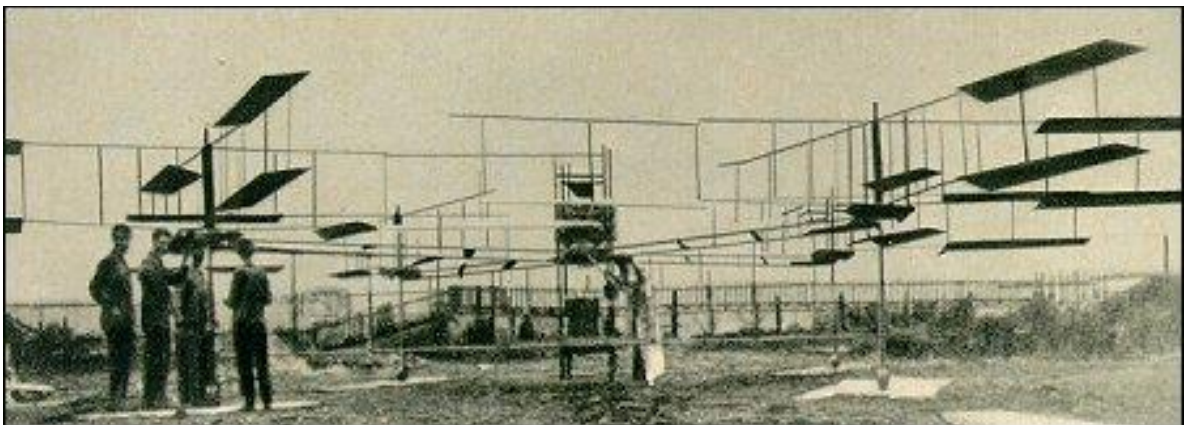
Segundo Keane e Carr (2013), os desenvolvimentos atuais em veículos aéreos não tripulados remontam ao início da Primeira Guerra Mundial. Os esforços durante o período entre guerras, a Segunda Guerra Mundial e o pós Guerra acabaram por levar ao desenvolvimento de mísseis de cruzeiro e à família atual de VANTs. Segundo os autores, em 1917 as forças armadas americanas desenvolveram um torpedo aéreo guiado e o mesmo pode ser visto na Figura 2.3.

Figura 2.1 – Balões bombardeando Veneza, 1849.



Fonte: Página sciencephoto⁴.

Figura 2.2 – Quadrirrotor *Gyrosplane* No. 1, 1907.

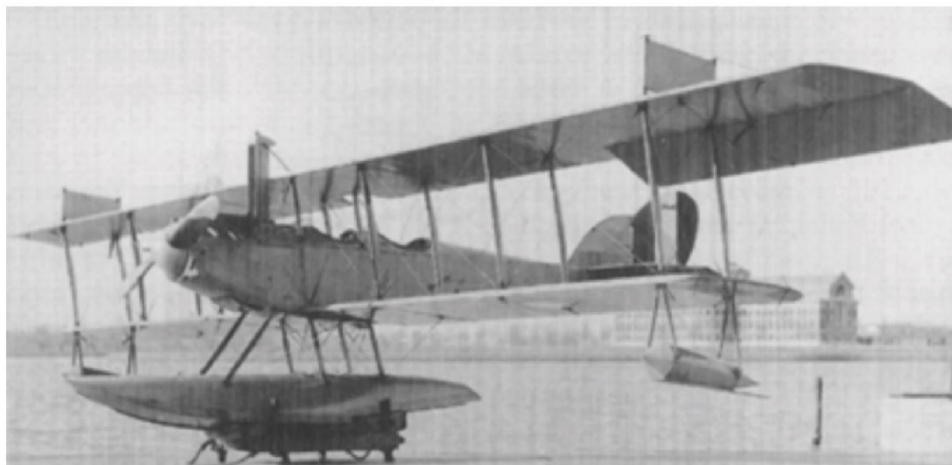


Fonte: Página aviastar⁵.

⁴ Disponível em: < <https://www.sciencephoto.com/media/642145/view/balloons-bombing-venice-1849> > Acesso em: 20 mar. 2020.

⁵ Disponível em: < http://www.aviastar.org/helicopters_eng/breguet_gyro.php > Acesso em: 20 mar. 2020.

Figura 2.3 – Torpedo aéreo, 1917.



Fonte: Keane e Carr (2013).

Um torpedo era acoplado a uma aeronave não tripulada que seguia uma rota pré-determinada através do uso de um giroscópio, um sistema pneumático, um sistema elétrico e um barômetro. Um sistema de controle mecânico foi desenvolvido para esta aeronave para que a mesma pudesse estimar sua distância percorrida.

Em 1922, segundo Leishman (2000), a *US Army Air Corps* desenvolveu um quadrirrotor com hélices de seis pás para cada rotor. Este quadrirrotor era conhecido como *Flying Octopus* e se consistia de uma estrutura em cruz formada por treliças e vigas, na qual um rotor era localizado em cada extremidade. Ele era capaz de realizar voos curtos a baixas altitudes e velocidades. Segundo Lima (2015), ele possuía uma massa de cerca de 1700 kg. Este quadrirrotor pode ser na Figura 2.4.

Já em 1924, segundo Keane e Carr (2013), pela primeira vez na história, uma aeronave rádio controlada foi pilotada remotamente em todas as fases do voo: decolagem, manobra e pouso. Tratava-se de uma aeronave F5-L, utilizada em experimentos de torpedos aéreos e montada com um sistema de rádio. Tal aeronave pode ser vista na Figura 2.5.

Em 1929, segundo Leishman (2000), o engenheiro russo Nicolas Florine construiu um dos primeiros multirrotores de sucesso. Os rotores rotacionavam na mesma direção, porém foram inclinados em direções opostas para cancelar a reação dos torques. Em 1933, Nicolas Florine desenvolveu um segundo projeto que realizou um voo de mais de 9 minutos a uma altitude de cerca de 5 metros. Segundo o autor, esse feito excedeu o registro de duração de voos da época. Este quadrirrotor pode ser visto na Figura 2.6.

Figura 2.4 – *Flying Octopus*, 1922.



Fonte: *Quadcopter Dynamics*⁶.

Figura 2.5 – Aeronave F5-L rádio controlada em voo, 1924.



Fonte: Keane e Carr (2013).

Em 1943, segundo Keane e Carr (2013), foi desenvolvido, em parceria com a Marinha americana, um VANT que era capaz de transmitir imagens em tempo real utilizando o sistema de televisão da época, o TDR-1. Segundo o autor, desta forma, o piloto era capaz de pilotar a aeronave à distância mesmo sem contato visual com a mesma. Esta aeronave foi utilizada na Segunda Guerra Mundial em 1944 e pode ser vista na Figura 2.7.

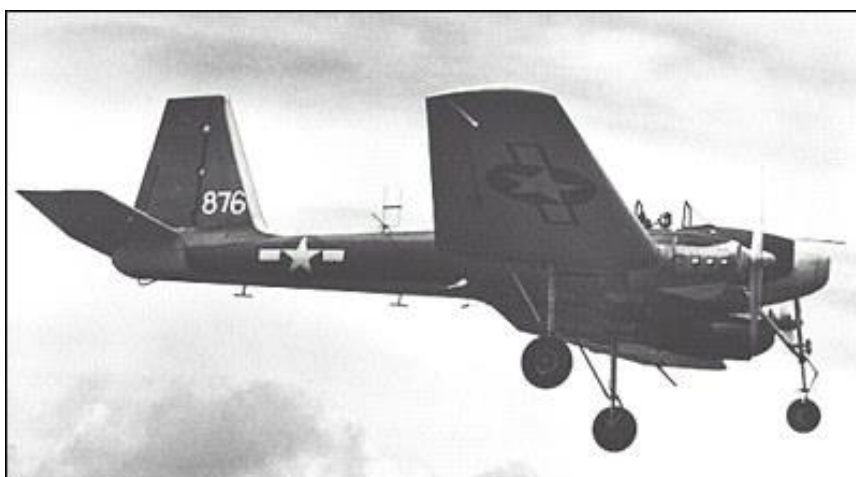
⁶ Disponível em: < <https://catsr.vse.gmu.edu/SYST460/QuadcopterDynamics.pdf> > Acesso em: 20 mar. 2020.

Figura 2.6 – Quadrirrotor desenvolvido por Nicolas Florine em voo, 1933.



Fonte: Leishman (2000).

Figura 2.7 – Aeronave TDR-1 em voo, 1944.



Fonte: Página avistar⁷.

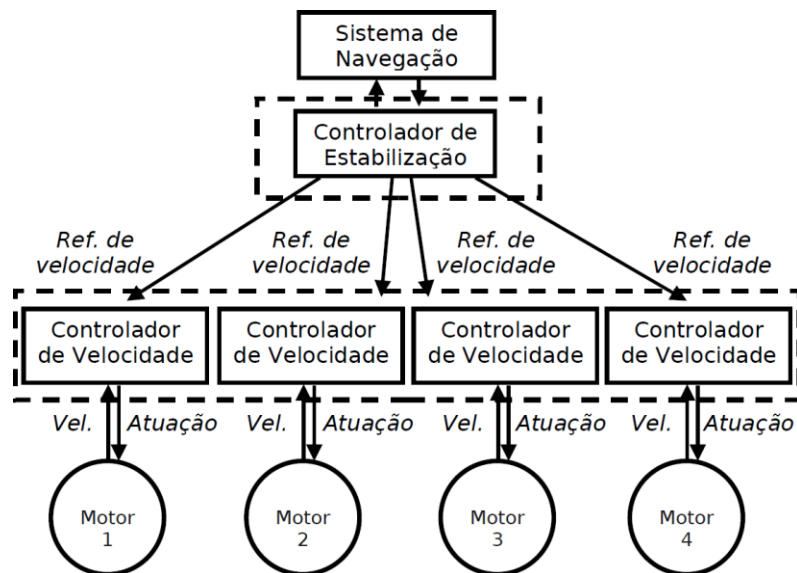
Segundo Lima (2015), os projetos de quadrirrotores acabaram sendo esquecidos por várias décadas devido à dificuldade de estabilização de voo, bem como os elevados custos para a época. Ainda segundo o autor, apenas após a década de 80, com os avanços tecnológicos em atuadores e sensores de baixa escala, também chamados de sistemas microeletrônicos, no armazenamento de energia e no processamento de informações, os quadrirrotores ressurgiram para serem utilizados com veículos aéreos não tripulados. Ainda segundo o autor, desde então empresas e universidades vêm pesquisando o tema de VANTs quadrirrotores.

⁷ Disponível em: < http://www.avistar.org/air/usa/interstate_tdr-1.php > Acesso em: 20 mar. 2020.

2.2 Técnicas de Controle

Segundo Santana e Borges (2009), uma possível arquitetura para o controle de estabilização de um quadricóptero é o modelo em cascata representado na Figura 2.8. Nela existe uma malha externa de controle para estabilização que transmite referências de velocidade às malhas internas de controle de rotações dos conjuntos hélice-rotor, o que faz com que o sistema de estabilização se torne menos sensível à dinâmica do sistema de propulsão. Segundo os autores, este modelo é válido desde que a malha interna convirja mais rapidamente que a externa.

Figura 2.8 – Arquitetura de um sistema de controle em cascata de um quadricóptero.

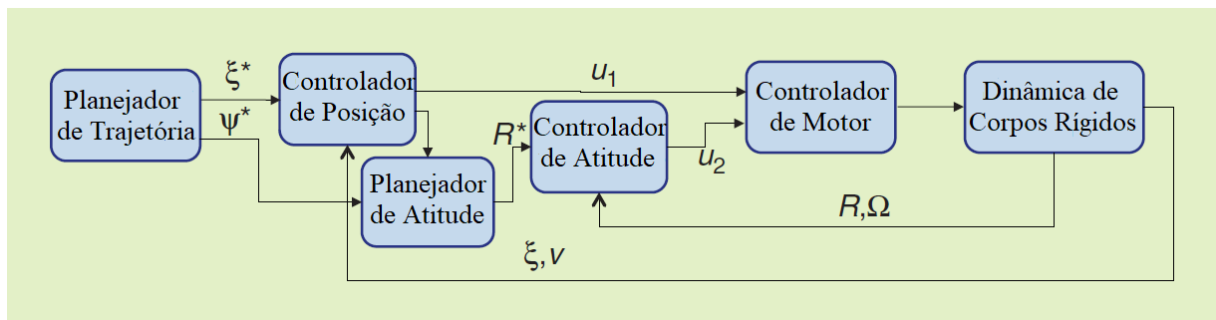


Fonte: Santana e Borges (2009).

Uma abordagem de controle hierárquico também é comum para quadricópteros segundo Mahony, Kumar e Corke (2012). No nível mais baixo, o de maior largura de banda, está o controle da velocidade de rotação do rotor. No próximo nível está o controle da atitude do veículo, e no nível superior está o controle de posição ao longo de uma trajetória. Esses níveis formam *loops* de realimentação aninhados, como pode ser visto na Figura 2.9.

Segundo Gao, Du e Ji (2017), o sistema de um quadricóptero é não linear, possui um forte acoplamento e é muito suscetível a interferências, o que torna o controle dessa estrutura bastante complexo. Atualmente existem vários métodos de controle de quadricópteros na literatura, como os baseados em controle PID, modos

Figura 2.9 – *Loop* de controle do motor mais interno, *loop* de controle de atitude intermediário e *loop* de controle de posição externa.



Fonte: Adaptado de Mahony, Kumar e Corke (2012).

deslizantes, *Backstepping*, adaptativos, redes neurais, H-Infinity, entre outros. Cada qual possui suas vantagens e desvantagens em relação aos outros. Portanto, a escolha do método de controle deve ser efetuada com cautela, a fim de se obter um resultado satisfatório perante a aplicação desejada. Uma breve discussão sobre cada uma destas técnicas, segundo as conclusões destes autores, é apresentada nos próximos parágrafos.

Um controle baseado em algoritmos PID é um tipo de controle que não necessita de um modelo matemático extremamente preciso. Isso faz com que algumas aproximações do modelo real sejam aceitáveis. Este tipo de tecnologia é relativamente madura, simples para utilizar e implementar. Por outro lado, é difícil se obter um resultado realmente satisfatório para sistemas que apresentam não linearidades e forte acoplamento como é o caso de quadricópteros. Outra desvantagem é que a escolha dos parâmetros do controlador depende de experiências prévias e é difícil se obter um resultado realmente satisfatório através destas escolhas.

Um controle baseado em modos deslizantes apresenta grande robustez, porém, devido à necessidade de manter algum algoritmo de lógica de chaveamento, pode levar ao fenômeno de vibração da estrutura, o que normalmente não é desejado, visto que este fenômeno pode levar à ruptura da estrutura e perturbações nos sensores que alimentam a malha de estabilização, ou seja, o erro se torna recursivo.

O controlador *Backstepping* necessita manter um modelo matemático mais preciso. Em contrapartida, pode ser uma boa escolha quando se trabalha com sistemas não lineares, como é o caso de quadrrrotores.

Um controlador que faz uso de redes neurais ganha robustez através do treinamento dos valores de velocidade e potência, porém exige um tempo razoável de treinamento para se atingir um controle satisfatório. Ou seja, embora possa trazer resultados muito bons, estes resultados podem demorar mais tempo do que o permitido durante a fase de desenvolvimento do projeto.

Um controlador baseado na técnica H-Infinity apresenta boa robustez, é adequado para a estabilização do sistema e traz confiabilidade. Possui, como principal vantagem a não necessidade um modelo preciso. Porém, necessita algumas diretivas de entrada que a princípio podem ser desconhecidas.

Segundo Khatoon, Gupta e Das (2014), o controlador LQR, aparenta ser um bom controlador devido ao seu excelente desempenho e robustez na planta. No trabalho desenvolvido por estes autores, "*PID & LQR Control for a Quadrotor: Modeling and Simulation*", foi concluído que o sistema do quadrrrotoz pode ser estabilizado utilizando tanto o controlador PID quanto o LQR e que o controlador PID ofereceu maior estabilidade, trazendo os polos de circuito fechado para o lado mais negativo do plano s em comparação com o controlador LQR. Os autores observaram também que os controladores LQR são robustos e produzem um erro de estado estacionário muito baixo com maior velocidade de resposta. Por outro lado, o controlador PID fornece resposta atrasada, o que a torna uma má escolha quando o sistema precisa de atualização rápida de parâmetros e não tem acesso direto a todos os estados do sistema.

Na subseção seguinte, são apresentados alguns trabalhos recentes que fazem uso das técnicas de controle abordadas nesta subseção.

2.3 Trabalhos Recentes

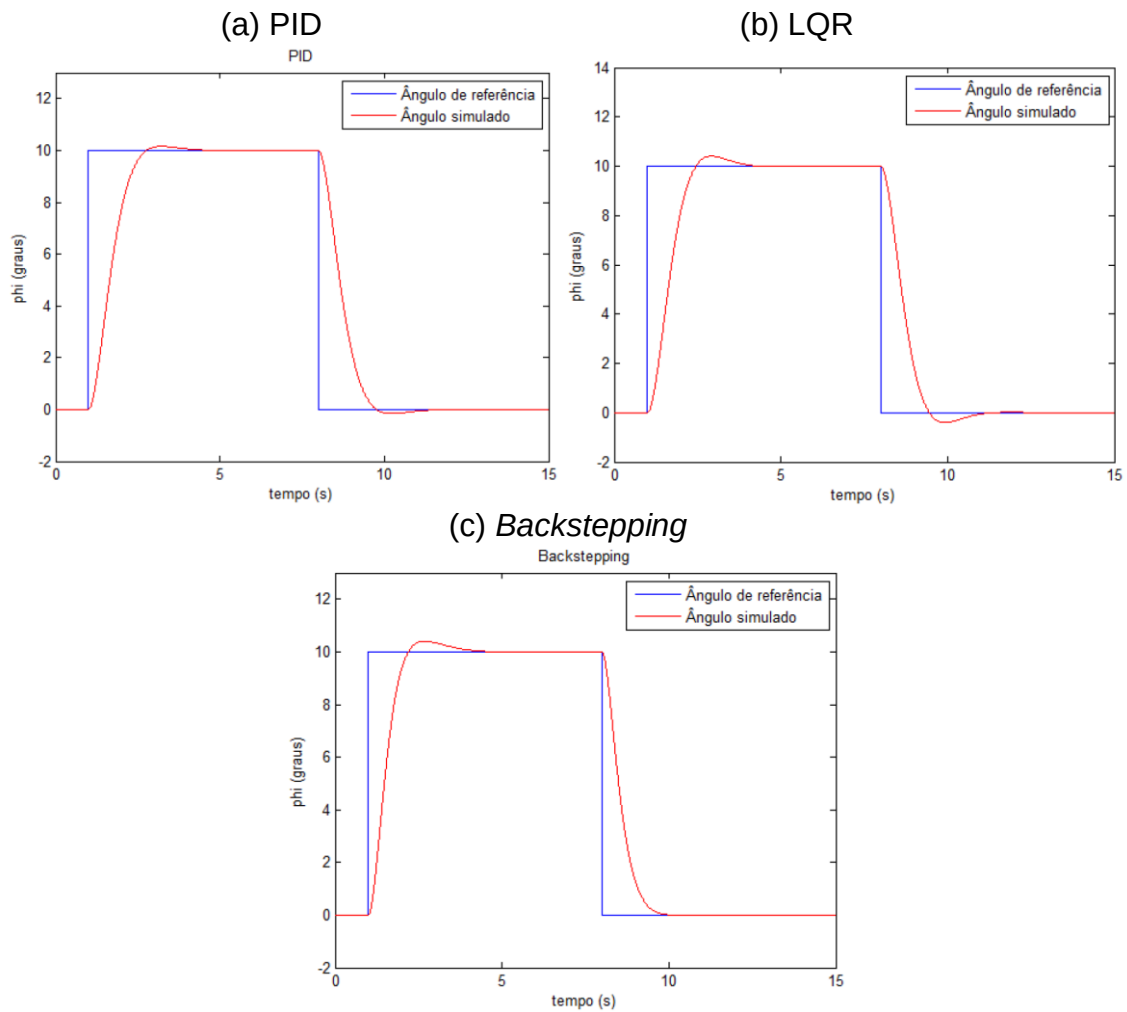
O trabalho desenvolvido por Alves (2012), "Estudo e Aplicação de Técnicas de Controle Embarcadas para Estabilização de Voo de Quadricopteros", aborda, assim como este trabalho, o desempenho de diferentes técnicas aplicadas ao controle de estabilização de voo de quadrrrotores. A autora concluiu que a técnica

de controle não linear utilizada apresentou um melhor desempenho, no que diz respeito à robustez, se comparado às técnicas lineares utilizadas e que, apesar de um desempenho pior, as técnicas de controle lineares são muito utilizadas no controle de estabilidade de quadricópteros devido à simplicidade para implementação em software. Contudo, a técnica de controle não linear mostrou-se mais robusta por levar em consideração, além do ângulo de interesse, os outros ângulos de rotação referentes à estabilidade da aeronave. Os resultados obtidos neste trabalho podem ser vistos nas Figuras 2.10a, 2.10b e 2.10c. Uma mesma entrada foi aplicada em cada controlador para que fosse possível estabelecer um comparativo entre o desempenho de cada um. A entrada do sistema, que corresponde ao valor de referência do ângulo φ , foi dada por um degrau de amplitude de 10° durante 6s que se iniciou no instante $t = 1s$. O tempo total de simulação foi de $t = 15s$ e a resposta do sistema foi o ângulo medido na saída.

Através da análise das Figuras 2.10a, 2.10b e 2.10c, é possível notar que o controlador *Backstepping* é o que atinge a referência no menor tempo, seguido pelo LQR e por último pelo PID. Em contrapartida, o PID é o que apresenta o menor sobressinal, seguido pelo *Backstepping* e pelo LQR. Por fim, em termos de tempo de acomodação, o controlador que apresenta o menor tempo é o LQR, seguido pelo PID e pelo *Backstepping*.

Assim como no projeto desenvolvido neste trabalho, o projeto desenvolvido no trabalho de Esteves, Moutinho e Azinheira (2015), “*Stabilization and Altitude Control of an Indoor Low-Cost Quadrotor: Design and Experimental Results*”, se baseia no desenvolvimento de quadricópteros de baixo custo. Neste trabalho, foi utilizado um simulador com parâmetros identificados experimentalmente a fim de permitir testar e ajustar os estimadores de Kalman e os controladores de atitude e altitude utilizando a técnica LQR. Os controladores foram implementados em software e embarcados no protótipo desenvolvido. Para a validação do controlador de atitude, foi utilizada a bancada de testes mostrada na Figura 2.11. Os resultados obtidos para os ângulos de rolagem, arfagem e guinada nesta bancada podem ser vistos nas Figuras 2.12a, 2.12b e 2.12c. Nesta Figura, é possível notar que a resposta possui um comportamento oscilatório e um desvio de aproximadamente 5° para os ângulos de rolagem e arfagem. Segundo os autores, este erro pode ser atribuído à tensão do cabo presente na estrutura de testes. Já o desvio no eixo de guinada de cerca de 23° , pode ser devido ao atrito presente entre a estrutura e o quadricóptero.

Figura 2.10 – Sistema simulado para cada controlador aplicado ao ângulo φ extraído do trabalho de Alves (2012).



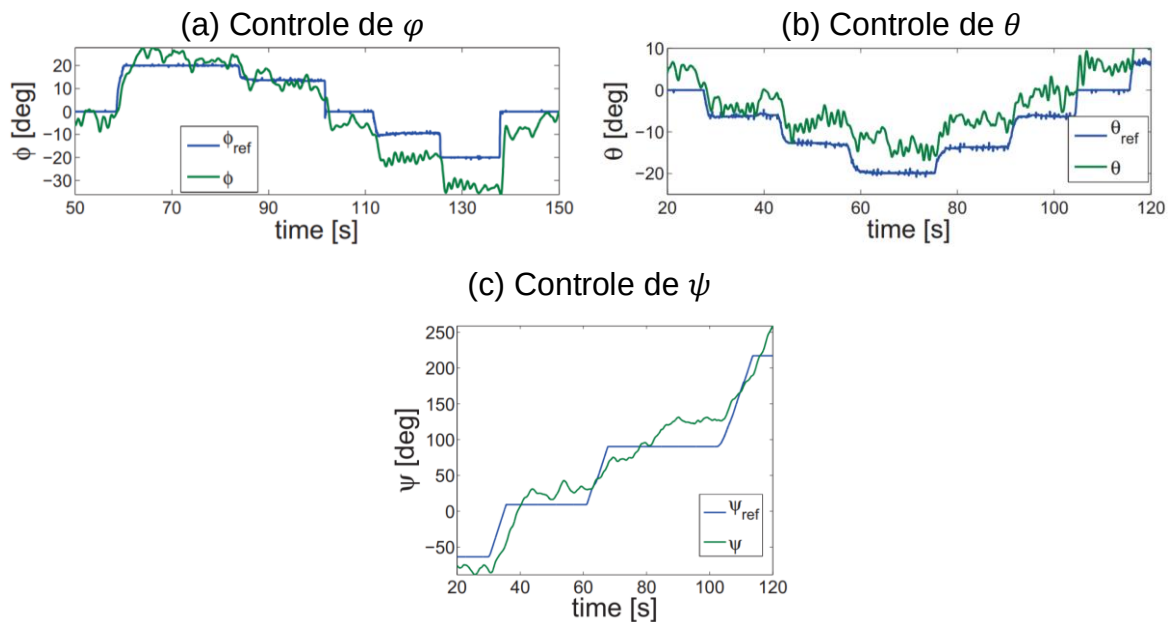
Fonte: Alves (2012).

Figura 2.11 – Bancada de testes utilizada no trabalho de Esteves, Moutinho e Azinheira (2015).



Fonte: Esteves, Moutinho e Azinheira (2015).

Figura 2.12 – Resultados em bancada de testes extraídos do trabalho de Esteves, Moutinho e Azinheira (2015).

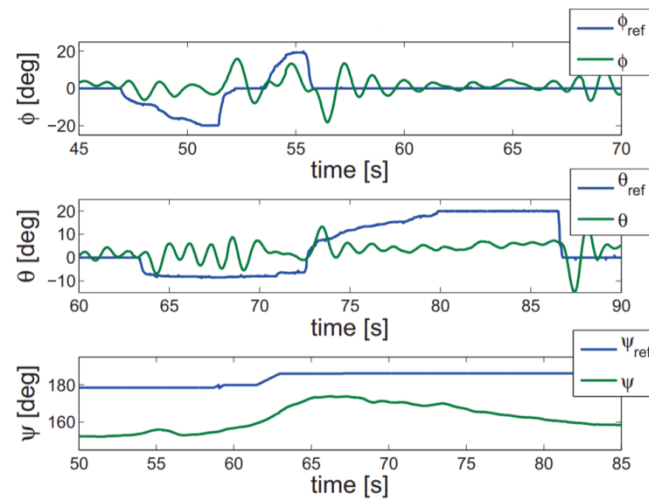


Fonte: Esteves, Moutinho e Azinheira (2015).

Os resultados obtidos em voo livre podem ser vistos na Figura 2.13. Com estes resultados, os autores concluíram que para os ângulos ϕ e θ , o sistema reage às referências fornecidas. Porém, imediatamente retornam a ângulos próximos de zero. Este fenômeno foi creditado pelos autores aos efeitos aerodinâmicos não modelados, causados pelo movimento de translação do quadrotor. Já para o eixo de guinada, o quadrotor se manteve mais próximo à referência se comparado ao teste realizado em bancada.

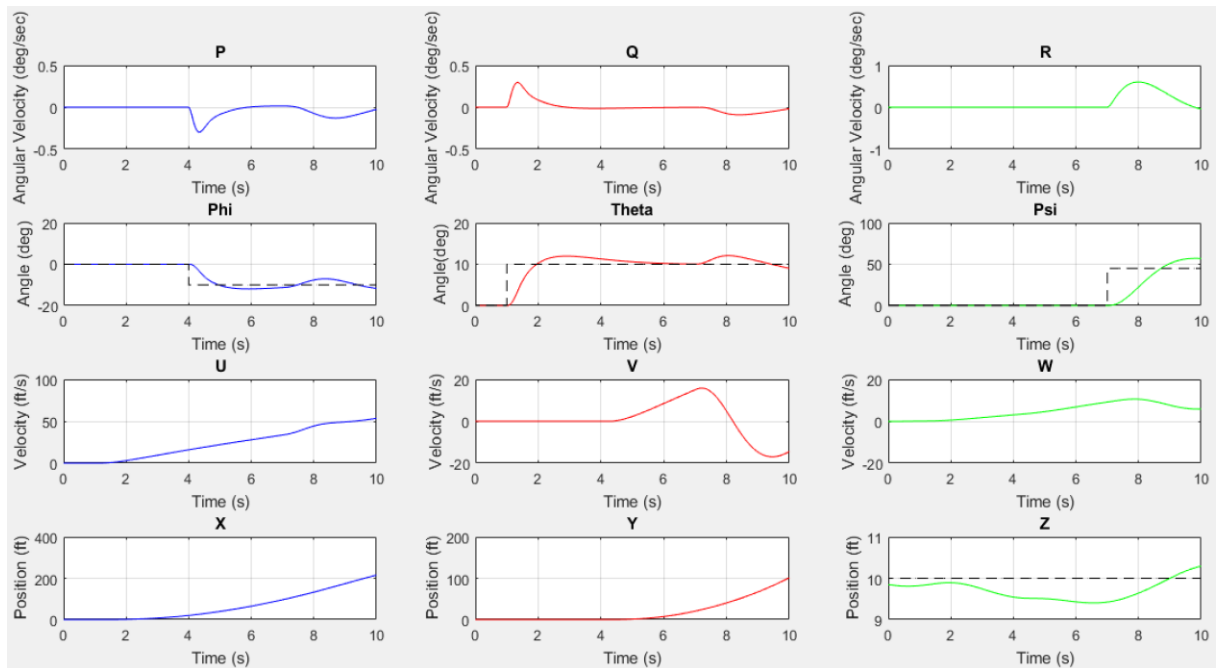
O trabalho de Zhongli *et al.* (2017), "*Design of Flight Attitude Control Model for the Miniquadrotor*", que aborda o desenvolvimento de um modelo e controlador para mini quadricópteros, aborda controladores com topologia PID. Entretanto, neste trabalho os controladores PI e PD foram utilizados separadamente pelo *loop* externo e interno na malha de controle, onde o *loop* externo é responsável pelo controle de posição horizontal e o *loop* interno pela velocidade angular. A Figura 2.14 mostra os resultados obtidos neste trabalho. As entradas utilizadas se tratam de degraus aplicados em um determinado ângulo de rotação. Para os ângulos ϕ e θ , foram aplicados degraus de amplitude de 10° enquanto que para o ângulo ψ foi aplicado um degrau de amplitude de 50° . Estes ângulos foram aplicados em momentos distintos durante a simulação.

Figura 2.13 – Resultados em voo livre extraídos do trabalho de Esteves, Moutinho e Azinheira (2015).



Fonte: Esteves, Moutinho e Azinheira (2015).

Figura 2.14 – Simulação de um controlador de atitude aplicado a um mini quadrrrotor extraído do trabalho de Zhongli *et al.* (2017).



Fonte: Zhongli *et al.* (2017).

Através dos resultados apresentados na Figura 2.14, é possível notar que os ângulos controlados, após atingirem o nível de referência, não se estabilizaram por completo, apresentando uma pequena oscilação em torno do nível de referência. Segundo os autores, isso deve ao fato de o *loop* do controlador de posição não ter sido incluído na simulação.

Assim como abordado na seção 1, uma das desvantagens da arquitetura Multirrotores é a autonomia reduzida se comparada às aeronaves de Asa Fixa. A fim de eliminar esta desvantagem, alguns estudos vêm sendo realizados, como é o caso do trabalho desenvolvido por Qian e Liu (2017), denominado “*Dynamics and control of a quadrotor with a cable suspended payload*”. Para elevar a autonomia de voo do quadrirrotores desenvolvido, o mesmo se mantém conectado a um cabo que transmite energia para aeronave. Isso faz com que pousos e decolagens para carregamento de baterias sejam desnecessários.

Porém, a presença do cabo altera a dinâmica de voo do quadrirrotores, visto que a massa do cabo gera uma carga extra que faz com que haja um desvio do centro de massa do quadrirrotores, denominada carga suspensa descentralizada. Segundo os autores, a principal dificuldade no controle deste tipo de quadrirrotores se consiste no fato de que o movimento da carga útil descentralizada se une ao movimento rotacional do quadrotor, o que dificulta a modelagem e o controle de derivação.

Para a modelagem matemática, os autores utilizaram o método descrito por Kane e Levinson (1985). Já para o controle da estrutura, foi utilizado uma decomposição LDU, *Lower–Upper Decomposition* ou Decomposição Inferior-Superior, decompondo o sistema em dois sistemas: um sistema rotacional e outro contendo a translação do quadrirrotores e a carga móvel do cabo.

Uma das formas mais usuais de se obter os dados para realimentação das malhas dos controladores de atitude é através do uso de GPS e unidades inerciais. Porém, o trabalho desenvolvido por Cabecinhas et al. (2019), “*Integrated Visual Servoing Solution to Quadrotor Stabilization and Attitude Estimation Using a Pan and Tilt Camera*”, utiliza câmeras para realimentar sua malha de controle. A solução compreende uma lei de controle para a posição e atitude do quadrotor, uma lei de controle para a plataforma de movimentação da câmera e um estimador para a atitude do veículo, cuja interconexão estabiliza o quadrotor pairando acima de determinados pontos de referência no solo. Segundo os autores, um dos principais obstáculos em se transpor esta solução de um ambiente de laboratório para ambiente externo é a necessidade do conhecimento dos locais de referência para a identificação visual, que nem sempre são disponíveis.

Na seção seguinte, é apresentado o projeto do quadrirrotores desenvolvido neste trabalho.

3 Projeto do Quadrirrotor

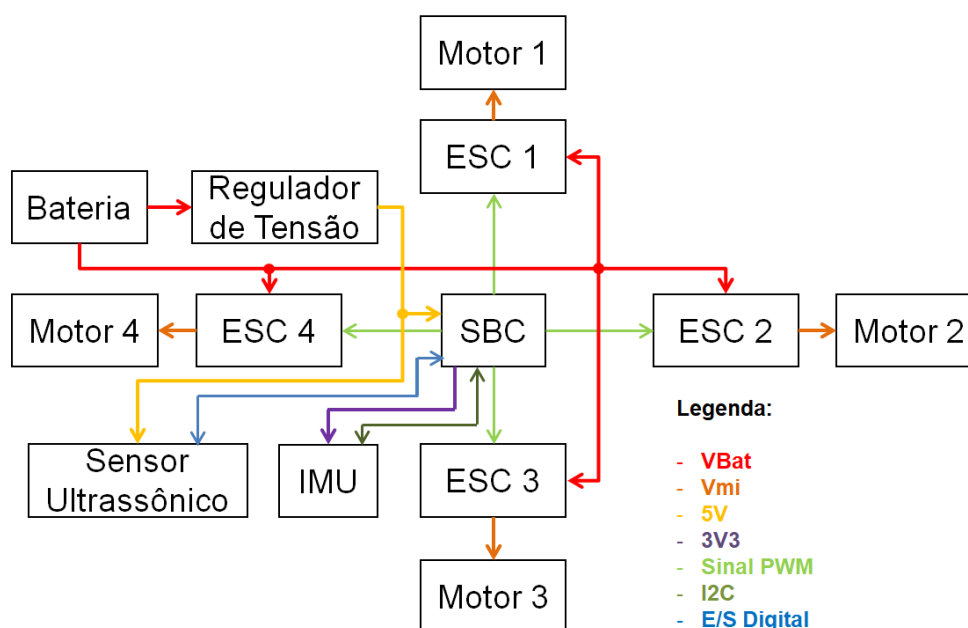
O desenvolvimento de um protótipo de quadrirrotor abrange diversas áreas do conhecimento da engenharia, tais como eletrônica, mecânica e computação. O protótipo desenvolvido neste trabalho foi projetado de tal forma a possuir uma autonomia de voo suficiente para realização dos testes e resistência mecânica a todas as condições que pode ser exposto durante o voo, tais como vento e vibrações geradas.

As subseções seguintes apresentam o projeto detalhado desenvolvido para cada uma destas áreas de conhecimento.

3.1 Projeto Eletrônico

A seleção dos componentes eletrônicos utilizados se deu a partir de uma análise dos requisitos mínimos necessários e levando em consideração seus custos de mercado. O diagrama de blocos que representa a conexão entre os componentes eletrônicos utilizados pode ser visto na Figura 3.1. As descrições destes componentes e o projeto eletrônico completo, englobando o circuito elétrico e as placas de circuito impresso, se encontram nas subseções seguintes.

Figura 3.1 – Diagrama de blocos do projeto eletrônico.



Fonte: Elaboração do próprio autor.

3.1.1 SBC

A SBC, *Single Board Computer* ou computador de placa única, utilizada para este projeto foi a PocketBeagle Board⁸. A Tabela 3.1 apresenta suas principais características relevantes para a este projeto. O fato desta SBC possuir quatro canais de saída PWM (*Pulse Width Modulation* ou modulação por largura de pulso) por hardware foi essencial para sua escolha neste projeto, visto que os sinais de controle enviados para os ESCs (*Electronic Speed Controller* ou controlador eletrônico de velocidade detalhados na subseção 3.1.7), que controlam individualmente a velocidade de rotação de cada motor, necessitam deste tipo de modulação. Suas dimensões e massa reduzida, se comparadas as do quadrirrotor como um todo, também foram essenciais para sua escolha. A Figura 3.2 apresenta a SBC utilizada.

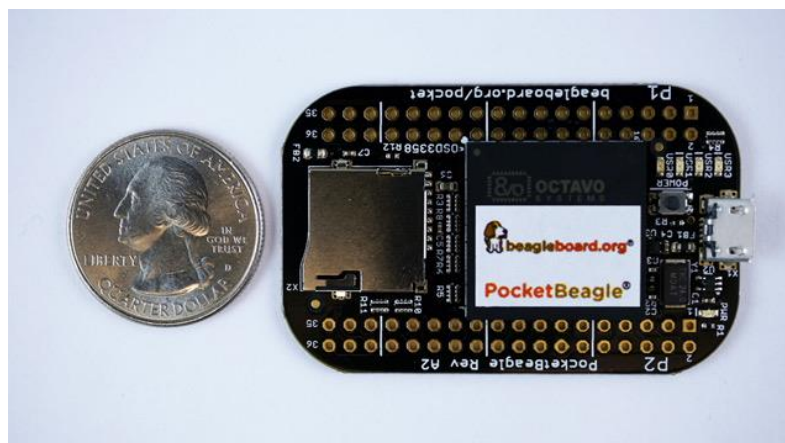
Tabela 3.1 – Especificações da SBC PocketBeagle Board.

Processador	Texas Instruments 1GHz Sitara™ AM3358 ARM® Cortex®-A8 com acelerador de ponto flutuante NEON
Dimensões	35 x 56 x 5 mm
Memória SDRAM	512MB DDR3 800MHz RAM
SD/MMC	Cartão de microSD para dados e boot de sistema
Entradas Analógicas	8 com resolução de 10 bits
Entradas/Saídas Digitais	44
I2C	2
PWM	4
Alimentação	5V
Massa	10g

Fonte: Página do fabricante⁸.

⁸ Disponível em: <<http://beagleboard.org/pocket>> Acesso em: 20 out. 2019.

Figura 3.2 – PocketBeagle Board.

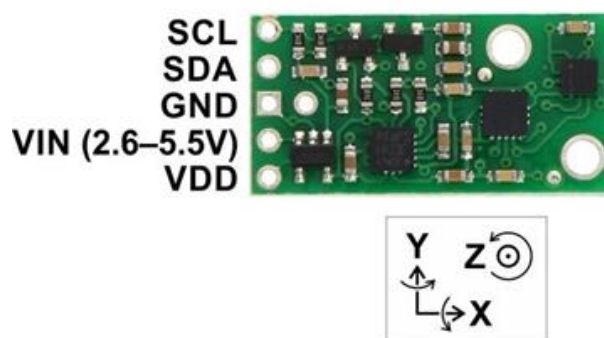


Fonte: Página do fabricante⁸.

3.1.2 IMU

Outro componente essencial para o funcionamento do controle do quadricóptero é a IMU, *Inertial Measurement Unit* ou unidade de medida inercial. Esta é a unidade que realiza o sensoriamento e realimenta a malha de controle para que esta possa atuar da forma desejada. A IMU utilizada neste projeto foi a Pololu AltIMU-10 v4⁹. Esta unidade possui sensores giroscópicos, acelerômetros e magnetômetro nos três eixos, além de um altímetro (barômetro), totalizando assim a medida de 10 variáveis. Sua interface de comunicação é a I2C e sua tensão de alimentação é de 3,3 V. A IMU utilizada pode ser vista na Figura 3.3.

Figura 3.3 – IMU Pololu AltIMU-10 v4.



Fonte: Página do fabricante⁹.

⁹ Disponível em: <<https://www.pololu.com/product/2470>> Acesso em: 20 out. 2019.

3.1.3 Sensor ultrassônico

O sensor ultrassônico utilizado neste projeto tem como finalidade realizar a medição da altitude do quadricóptero durante a decolagem e pouso, já que nestes dois cenários, a precisão necessária é maior do que a que pode ser fornecida pelo altímetro presente na IMU utilizada, apresentada na subseção anterior. Neste tipo de sensor, uma onda ultrassônica é emitida e um receptor fica aguardando a reflexão desta onda. A partir da medição do tempo entre a emissão e recepção, e sabendo de antemão a velocidade de propagação da onda no ar, é possível realizar um cálculo que diz qual é a distância entre o sensor e objeto que refletiu a onda.

Desta forma, para que este sensor possa ser utilizado com a finalidade de determinação da altitude do quadricóptero, ele deve ficar apontado para baixo em direção ao solo.

O sensor utilizado neste projeto é o HC-SR04¹⁰ e pode ser visto na Figura 3.4. Seu alcance mínimo é de 2,0 cm e máximo de 4,0 m. Sua precisão é de 3,0 mm.

Figura 3.4 – Sensor ultrassônico HC-SR04.



Fonte: Página do vendedor¹⁰.

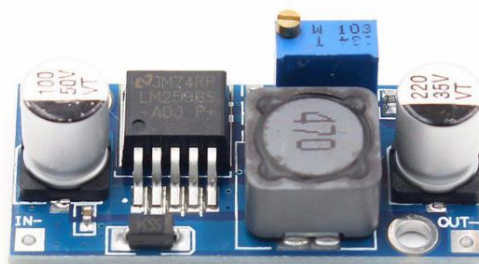
3.1.4 Regulador de tensão

O regulador de tensão utilizado neste projeto é um conversor DC-DC chaveado abaixador de tensão do tipo *buck* do modelo LM2596S¹¹ e pode ser visto na Figura 3.5.

¹⁰ Disponível em: < <https://www.filipeflop.com/produto/sensor-de-distancia-ultrassonico-hc-sr04/>> Acesso em: 20 out 2019.

¹¹ Disponível em: <<https://www.tindie.com/products/blinkingsilver/lm2596-lm2596s-dc-dc-step-down-module/>> Acesso em: 20 out 2019.

Figura 3.5 – Regulador de tensão LM2596S.



Fonte: Página do vendedor¹¹.

A tensão de saída pode ser ajustada através de seu *trimpot*. Suas características mais relevantes para o projeto estão descritas na Tabela 3.2. Como a SBC utilizada deve ser alimentada com uma tensão de 5V, a saída do regulador utilizado foi ajustada para esta tensão.

Tabela 3.2 – Especificações do Regulador de Tensão LM2596S.

Tensão de entrada	3 – 40 V
Tensão de saída	1.3 – 35 V
Corrente máxima de saída	3 A
Eficiência máxima	92 %
Frequência de chaveamento	65 kHz

Fonte: Página do vendedor¹¹.

3.1.5 Bateria

A bateria utilizada neste projeto é uma bateria do tipo LiPo (Lítio Polímero). Uma imagem da bateria utilizada pode ser vista na Figura 3.6.

A bateria de LiPo é uma bateria feita com um polímero de lítio. É uma bateria relativamente leve e de baixa manutenção, uma vantagem que a maioria das baterias químicas não possui. Nenhum ciclisto programado é necessário para prolongar a vida desta bateria e a mesma não possui nenhum tipo de memória. Além disso, a auto descarga é inferior à metade em relação à bateria de níquel-cádmio, o que a torna adequada para aplicações que exigem elevados níveis de energia e corrente.

Figura 3.6 – Bateria de LiPo 3S 15C 5200 mAh da TATTU.



Fonte: Página do vendedor¹².

Cada célula da bateria possui uma tensão nominal de 3,7 V, sendo que sua tensão de carga máxima atinge 4,2 V. A máxima corrente de descarga da bateria é proporcional à carga da mesma multiplicada por uma constante de descarga específica de cada bateria. O mesmo vale para a máxima corrente de carga, porém utilizando uma constante de carga.

A bateria utilizada possui três células (12,6 V em carga máxima), capacidade de carga de 5200 mAh e constante de descarga de 15C. Logo, a corrente máxima de saída constante equivale a 78A.

Sua massa é de 358 g. Na subseção 3.2 será visto que esta massa é bastante significativa ao compará-la com a massa total do quadricóptero.

3.1.6 Motores

Os motores utilizados para gerarem a propulsão para o quadricóptero são do tipo *brushless* e, como o próprio nome indica, não possuem escovas.

Segundo a empresa Renesas¹³, nos motores com escovas, as escovas fornecem corrente através do comutador nas bobinas do rotor. O mesmo não acontece num motor sem escovas porque as bobinas não estão localizadas no rotor. Em vez disso, o rotor é um ímã permanente. As bobinas não giram, mas são fixas no lugar do estator. Como as bobinas não se movem, não há necessidade de escovas nem de um comutador.

¹² Disponível em: <<https://www.fabmodelismo.com.br/produto/pecas-e-acessorios/baterias-em-geral/8277-bateria-lipo-5200-3s-15c-11-1v-tattu-gens-ace-drone-aero>> Acesso em: 20 out 2019.

¹³ Disponível em: <<https://www.renesas.com/br/en/support/technical-resources/engineer-school/brushless-dc-motor-01-overview.html>> Acesso em: 20 out 2019.

Uma grande vantagem deste tipo de motor é a sua eficiência, pois esses motores podem controlar continuamente a força rotacional (torque) máxima. Os motores com escovas, em contraste, atingem o torque máximo em apenas alguns pontos da rotação. Para um motor com escovas fornecer o mesmo torque que um modelo sem escovas, ele precisaria usar ímãs maiores. É por isso que mesmo pequenos motores *brushless* podem fornecer uma potência maior do que motores com escovas maiores.

Ainda segundo a Renesas¹³, outra grande vantagem é a controlabilidade. Os motores *brushless* podem ser controlados, usando mecanismos de realimentação, para fornecerem de forma precisa tanto o torque quanto a velocidade de rotação desejados. O controle de precisão, por sua vez, reduz o consumo de energia e a geração de calor e, nos casos em que os motores são alimentados por bateria, prolonga a vida útil da mesma.

Motores *brushless* também oferecem alta durabilidade e baixa geração de ruído elétrico devido à ausência de escovas. Em motores com escovas, as escovas e o comutador desgastam-se como resultado do contato móvel contínuo e também produzem faíscas onde o contato é feito. O ruído elétrico, em particular, é o resultado das fortes faíscas que tendem a ocorrer nas áreas onde as escovas passam sobre as aberturas no comutador. Segundo a Renesas¹³ é por isso que os motores *brushless* são frequentemente considerados preferíveis em aplicações onde é importante evitar o ruído elétrico.

Os motores utilizados neste projeto são do modelo XXD A2212/13T¹⁴. Este modelo de motor pode ser visto na Figura 3.7 e suas principais características relevantes para este projeto podem ser vistas na Tabela 3.3. A curva da eficiência energética versus corrente aplicada pode ser vista na Figura 3.8. Nesta curva, é possível notar que a eficiência máxima do motor se encontra entre 4 e 6 A.

Tabela 3.3 – Especificações do Motor *Brushless* XXD A2212/13T.

Número de células (bateria)	2-3 para LiPo / 6-10 para NiCd ou NiMH
kV (Rotações por Volt aplicado)	1000 RPM
Eficiência máxima	80 %

¹⁴ Disponível em: <https://www.rhydolabz.com/documents/26/BLDC_A2212_13T.pdf> Acesso em: 20 out 2019.

Corrente na eficiência máxima	4-10 A (>75%)
Corrente sem carga	0.5 A a 10V
Resistência	0.090 Ω
Corrente máxima	13 A durante 60 segundos
Potência máxima	150 W
Massa	52,7 g

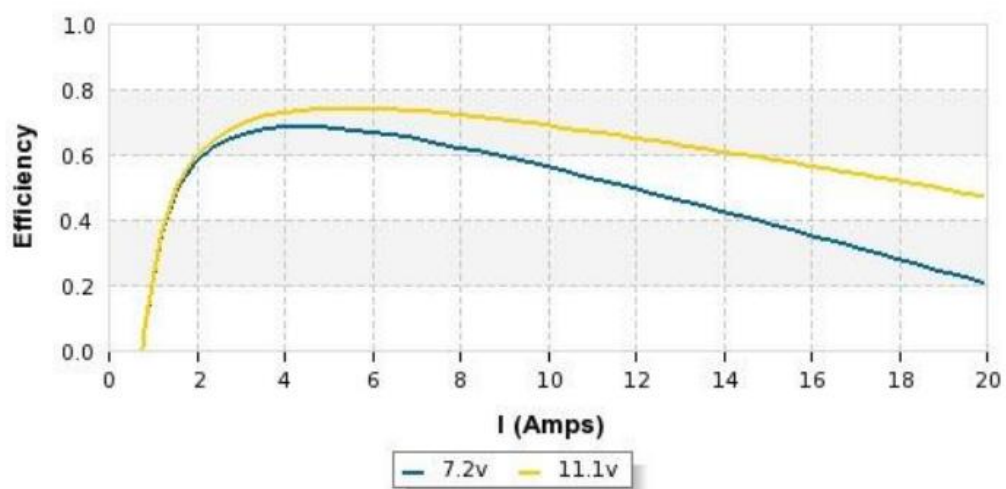
Fonte: Página do fabricante¹⁴.

Figura 3.7 – Motor *brushless* XXD A2212/13T.



Fonte: Página do fabricante¹⁴.

Figura 3.8 – Eficiência energética versus corrente aplicada ao motor XXD A2212/13T.



Fonte: Página do fabricante¹⁴.

3.1.7 ESC

O ESC, *Electronic Speed Controller* ou controlador eletrônico de velocidade, é o componente utilizado para realizar o controle da velocidade de rotação e torque dos motores. Ele recebe como entrada uma alimentação de corrente contínua e um sinal de controle do tipo PWM. Sua saída é trifásica e as defasagens entre as fases e sua frequência são ajustas automaticamente de modo a produzir a velocidade de rotação na saída conforme o sinal de controle aplicado.

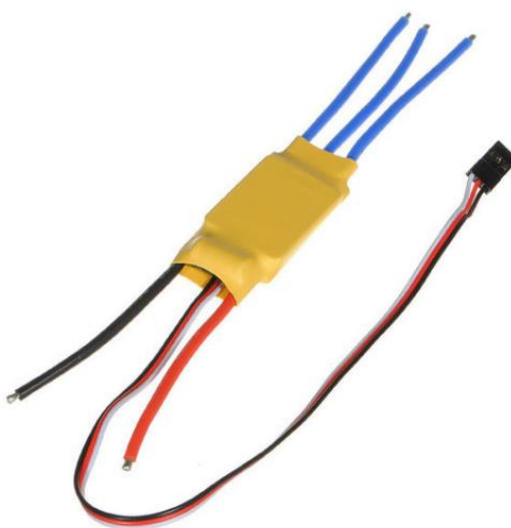
Os ESCs utilizados neste projeto são do modelo HW30A¹⁵. Suas principais características são apresentadas na Tabela 3.4. Uma imagem do ESC utilizado pode ser vista na Figura 3.9.

Tabela 3.4 – Especificações do ESC HW30A.

Corrente máxima de saída	30 A contínuo / 40 A de pico até 10 segundos
Tensão de entrada	5,6 – 16,8 V
Massa	25 g
BEC Linear	2 A / 5 V

Fonte: Página do vendedor¹⁵.

Figura 3.9 – ESC HW30A.



Fonte: Página do vendedor¹⁵.

¹⁵ Disponível em: <<https://www.filipeflop.com/produto/esc-30a-brushless-com-bec-interno-2a5v-aeromodelos/#tab-accessories>> Acesso em: 20 out 2019.

Este ESC possui um BEC, *Battery Eliminator Circuit* ou circuito eliminador de bateria, que se trata de um circuito eletrônico destinado a fornecer potência elétrica a outro circuito sem a necessidade de uma bateria. Porém, como o BEC existente neste ESC é linear, a sua eficiência é baixa e o calor gerado é alto se comparados a um regulador chaveado com a mesma potência de saída. Portanto, os BECs presente nestes ESCs não foram utilizados para alimentar a SBC, e sim o regulador chaveado descrito na subseção 3.1.4.

3.1.8 Circuito Elétrico

Para se realizar a conexão entre todos os componentes de forma organizada e enxuta, foi necessário o desenvolvimento de duas PCBs (*Printed Circuit Boards*) ou PCIs (Placas de Circuito Impresso). Para isso, foi utilizado o Software Eagle v. 9.2.2. Esta ferramenta foi utilizada por possuir um tipo de licença gratuita voltada ao uso de estudantes.

O circuito elétrico do quadrirrotor foi dividido em duas partes: controle e potência. As subseções seguintes detalham cada uma delas.

3.1.8.1 Circuito de controle

O circuito de controle é o cérebro do quadrirrotor. É ele quem recebe todos os sinais dos sensores e comandos de movimentação e envia os comandos de atuação aos motores para que tanto a estabilização quanto a movimentação desejada sejam realizadas.

Além dos componentes descritos nas subseções de 3.1.1 a 3.1.7, alguns outros componentes foram utilizados, conforme descrito nos próximos parágrafos.

Como o nível de saída da SBC utilizada é de 3,3V e o sinal de controle de entrada no ESC deve possuir amplitude de 5V, foi necessária a inserção de um circuito *level shifter*, ou deslocador de nível, que é composto pelo uso de dois resistores e um transistor MOSFET canal N. No total, foram inseridos quatro circuitos do tipo *level shifter*, já que cada ESC necessita de um.

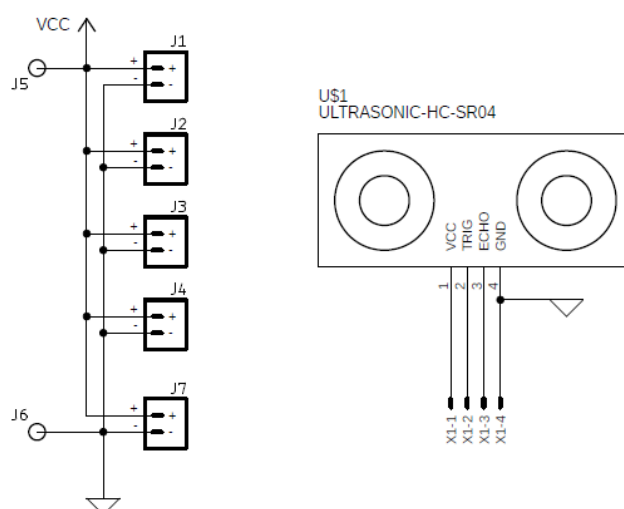
Também foi inserido um DIP *switch* (interruptor eletrônico) de seis vias a fim de ser possível realizar a seleção do tipo de controlador a ser utilizado durante o voo. A saída de cada via foi ligada a uma entrada digital da SBC, que faz a leitura

drenada pelos motores será equivalente a 52A. Para facilitar tanto o layout quanto a disposição dos componentes pelo quadrirrotor, todos componentes que exigem correntes altas estão conectados a este circuito. Na subseção 3.1.9.2 será mostrado como foi solucionado a questão das correntes elevadas que circulam pela PCI.

Como será visto na subseção 3.2.1, a PCI deste circuito de potência é posicionada na parte inferior do quadrirrotor. Portanto, o sensor ultrassônico também foi inserido nesta PCI somente pelo motivo da localização e não por necessitar de uma corrente elevada para seu funcionamento.

A Figura 3.11 mostra o circuito, que nada mais é do que um conjunto de conectores. A bateria é conectada aos conectores J5 e J6, os conectores J1, J2, J3 e J4 são conectadas aos ESCs e o conector J7 à placa de controle. Os motores são então conectados nas saídas dos ESCs e os sinais de controle dos ESCs são conectados na placa de controle apresentada na subseção anterior.

Figura 3.11 – Esquemático da PCI de Potência.



Fonte: Elaboração do próprio autor.

3.1.9 PCIs

Conforme descrito na seção anterior, foram confeccionadas duas PCIs para este projeto. Para ambas, foi utilizado o software Eagle para o desenvolvimento do *layout* a partir do esquemático. Foram utilizadas PCIs de fibra de vidro devido às suas melhores características de leveza e resistência mecânica se comparadas às de fenolite.

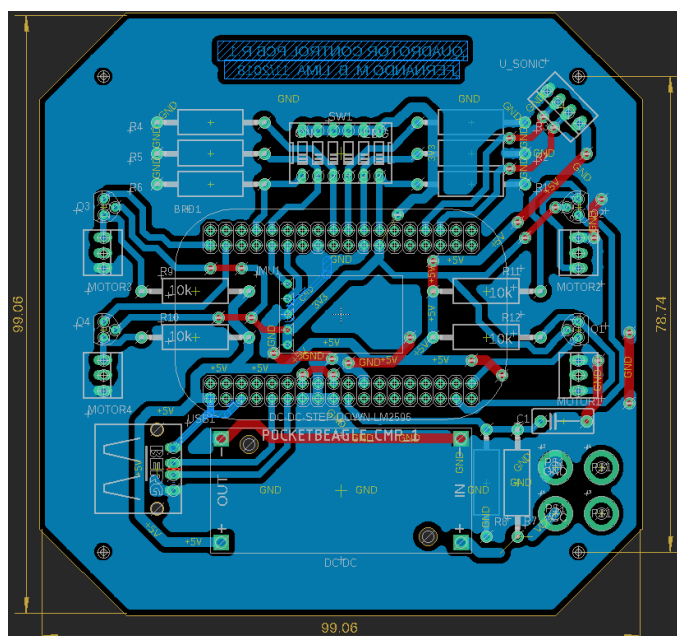
O processo de confecção utilizado foi a transferência térmica aplicada a uma placa de fibra de vidro cobreada. Após a transferência térmica, a placa passou pelo processo de corrosão química em percloroeto de ferro. Ao entrar em contato com o percloroeto de ferro, o cobre se dissolve. Já a região de cobre que está coberta pela tinta transferida não se dissolve. Após este processo, a tinta foi removida, as furações necessárias foram realizadas e os componentes foram montados e soldados.

A fim de melhorar a aparência das PCIs desenvolvidas, ambas foram pintadas de preto e envernizadas antes da montagem dos componentes. Tanto os layouts quanto as PCIs desenvolvidas estão descritas nas subseções 3.1.9.1 e 3.1.9.2.

3.1.9.1 PCI de Controle

O layout da PCI de controle pode ser visto na Figura 3.12. Já a PCI real após a transferência térmica e corrosão pode ser vista na Figura 3.13.

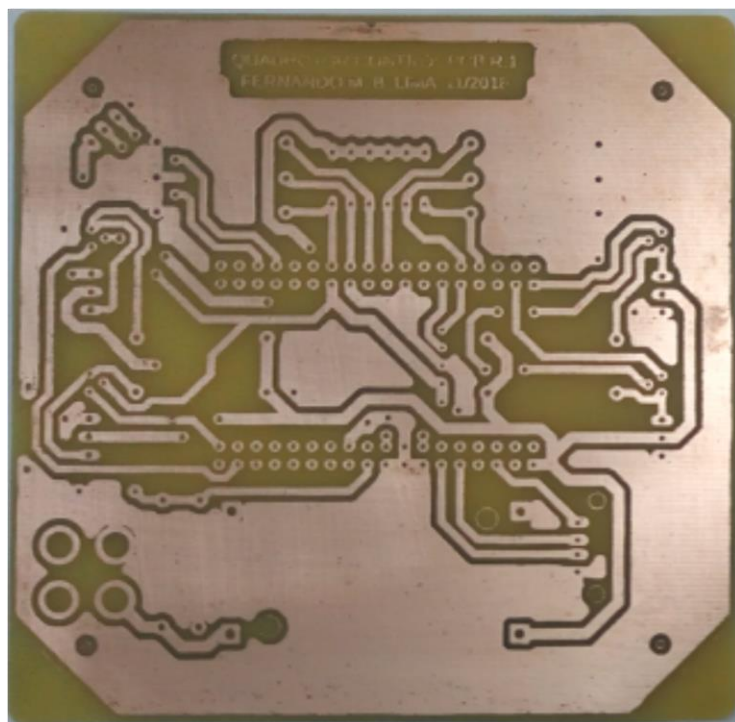
Figura 3.12 – Layout da PCI de Controle.



Fonte: Elaboração do próprio autor.

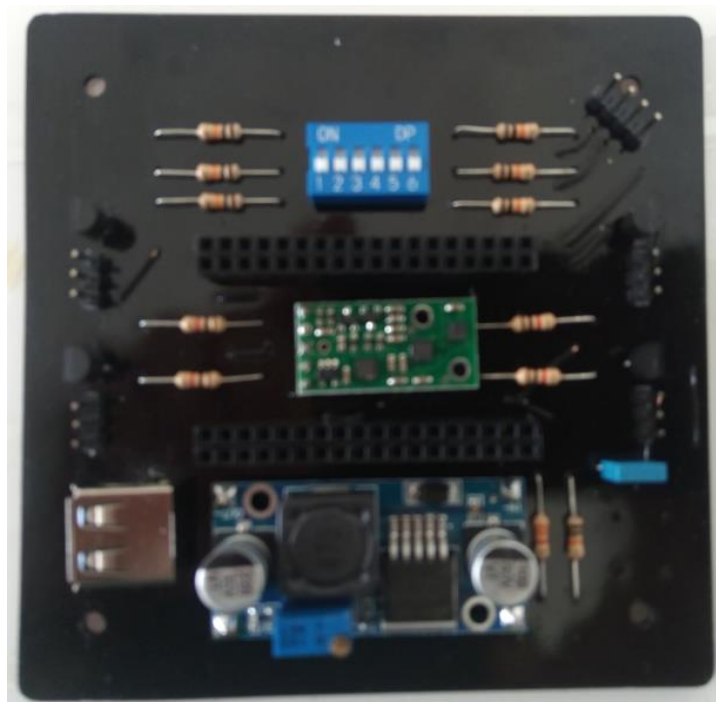
A PCI finalizada já pintada, furada e com todos os componentes montados, exceto a SBC, pode ser vista na Figura 3.14. Já na Figura 3.15 é mostrada a PCI completa inclusive com a SBC montada.

Figura 3.13 – PCI de Controle após a corrosão.



Fonte: Elaboração do próprio autor.

Figura 3.14 – PCI de Controle sem a SBC.



Fonte: Elaboração do próprio autor.

Figura 3.15 – PCI de Controle com a SBC.



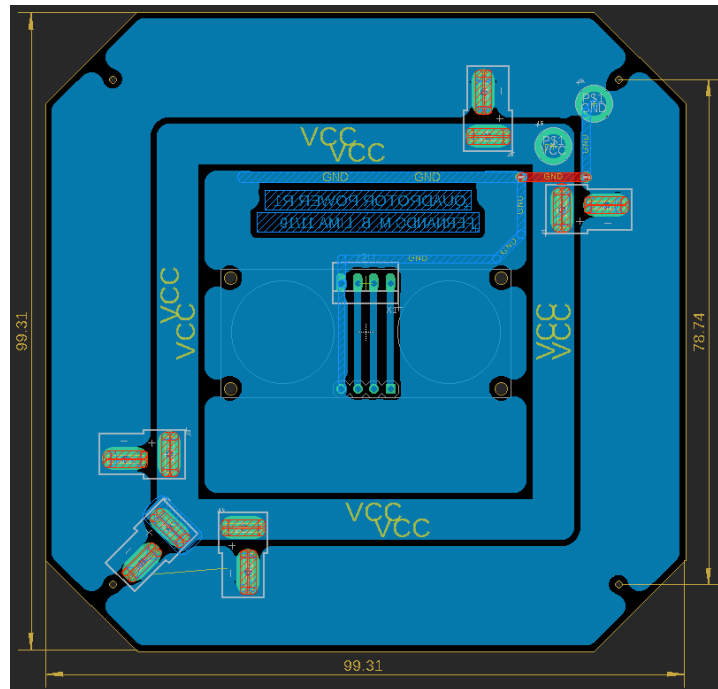
Fonte: Elaboração do próprio autor.

3.1.9.2 PCI de Potência

O layout da PCI de potência pode ser visto na Figura 3.16. Já a PCI real após a transferência térmica e corrosão pode ser vista na Figura 3.17. A PCI real apresenta um recorte diferenciado. Estas abas laterais tem como finalidade a proteção da bateria e será mais detalhada na seção 3.2.1.

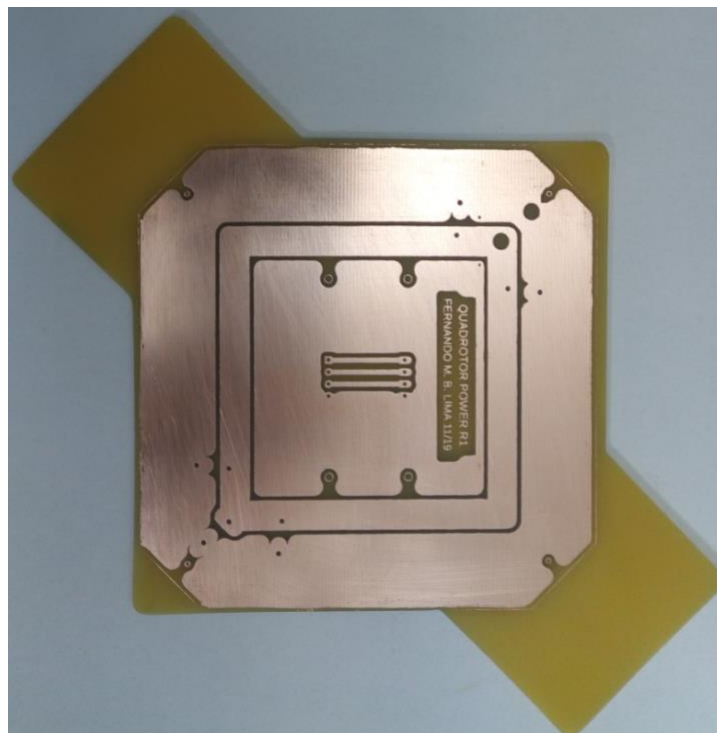
Porém, conforme mencionado na seção 3.1.8.2, a corrente que circula por esta PCI pode ser elevada se comparada às correntes máximas suportadas em placas de circuito impresso convencionais. Logo, foi necessária a inserção de reforços nas trilhas de corrente para que as mesmas fossem capazes de suportar a corrente elevada que pode passar sobre elas. Tais reforços se consistem da adição de cobre e estanho sobre a PCI, de forma a aumentar a área da seção de condução. A PCI com os reforços pode ser vista na Figura 3.18. Já a face superior com os componentes montados pode ser vista na Figura 3.19.

Figura 3.16 – Layout da PCI de Potência.



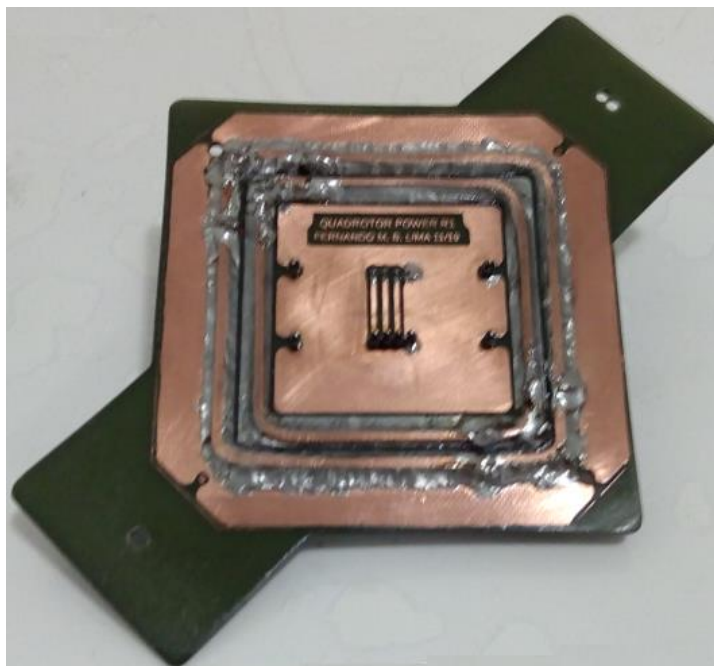
Fonte: Elaboração do próprio autor.

Figura 3.17 – PCI de Potência após a corrosão.



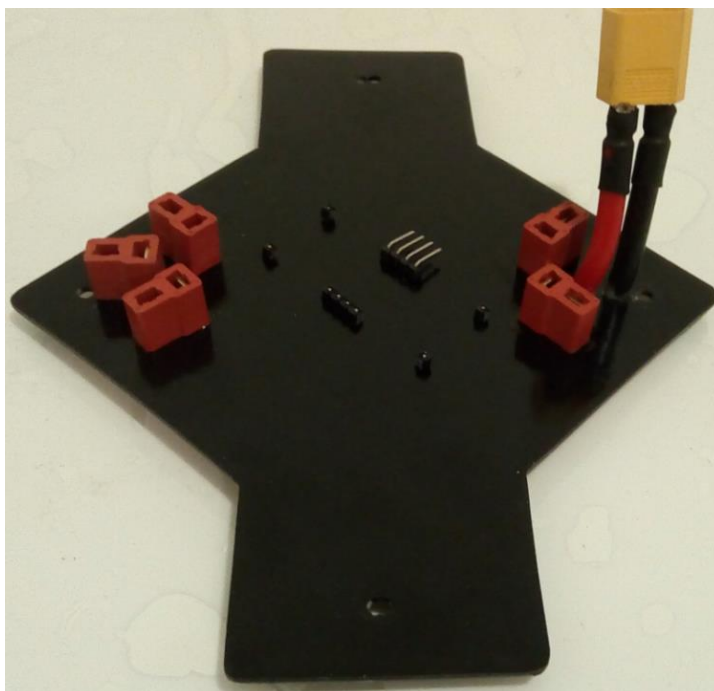
Fonte: Elaboração do próprio autor.

Figura 3.18 – PCI de Potência após a inserção de reforços nas trilhas.



Fonte: Elaboração do próprio autor.

Figura 3.19 – PCI de Potência vista de cima.



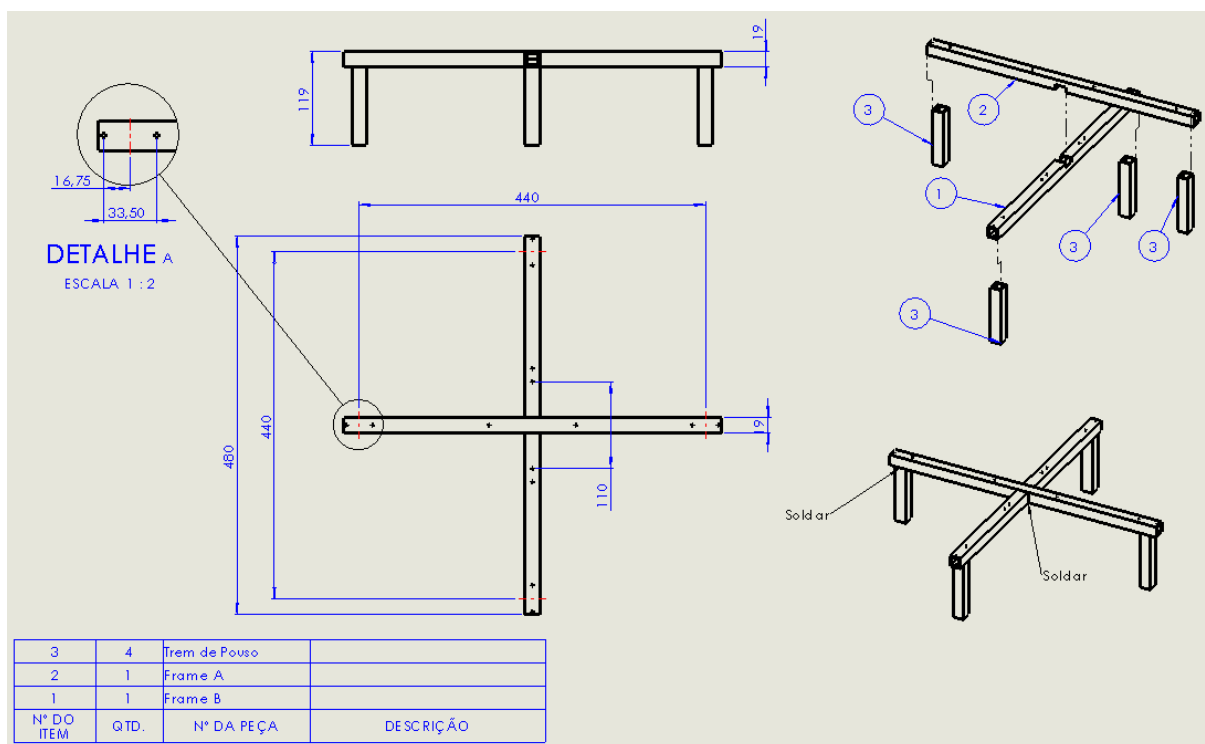
Fonte: Elaboração do próprio autor.

3.2 Projeto Mecânico

Para o desenvolvimento do projeto mecânico, foi utilizado um software tanto para design do CAD (*Computer-Aided Design* ou desenho assistido por computador) 2D quanto do 3D. Todas as peças foram projetadas separadamente e depois acopladas utilizando a funcionalidade de montagem desta ferramenta. Desta forma, em todas as etapas deste projeto mecânico serão apresentados o CAD 2D contendo as cotas (em mm), o CAD 3D e o modelo real.

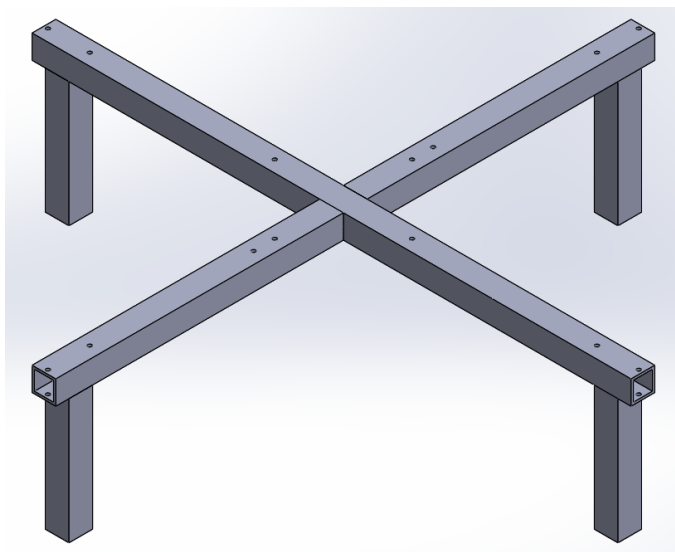
A estrutura do quadricóptero se consiste basicamente de uma cruz formada por dois tubos de perfil quadrado de alumínio e quatro pés que estão situados nas extremidades dos tubos e servem como trem de pouso. Esta estrutura foi desenvolvida em tubos de alumínio na liga 6063 devido às suas características, como baixa densidade ($2,69 \text{ g/cm}^3$) e dureza elevada (60 a 62 HB). Todos os componentes da estrutura foram soldados, a fim de manter a rigidez da estrutura a maior possível. A Figura 3.20 mostra o CAD 2D da estrutura, a 3.21 o seu CAD 3D e a 3.22 a estrutura real antes da realização das furações.

Figura 3.20 – CAD 2D da Estrutura do quadricóptero.



Fonte: Elaboração do próprio autor.

Figura 3.21 – CAD 3D da Estrutura do quadrirrotor.



Fonte: Elaboração do próprio autor.

Figura 3.22 – Estrutura do quadrirrotor antes da furação.

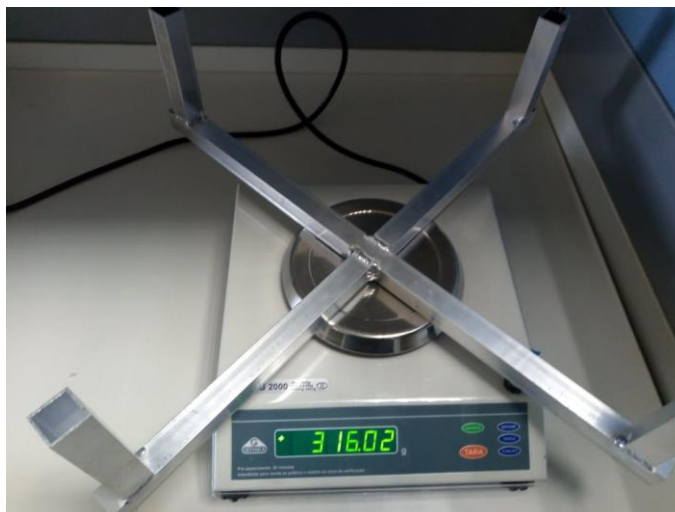


Fonte: Elaboração do próprio autor.

Conforme pode ser visto na Figura 3.20, a largura do perfil de alumínio utilizado é de 19 mm e a espessura da parede do mesmo é de 1,2 mm. O comprimento de cada barra da cruz é de 480 mm e a distância entre o centro de cada rotor após a fixação é de 440 mm. Logo, o valor que será inserido na modelagem da seção 4, como a distância entre dois rotores na mesma barra, é de 440 mm. Já os pés, que servirão como trem de pouso, possuem 100 mm de comprimento cada. A Figura 3.23 mostra a pesagem da estrutura indicando que sua

massa total é de 316,02 g, onde 23,58 g equivalem à massa de cada pé e 221,69 g equivalem à massa da cruz. Estes valores serão utilizados na modelagem matemática dinâmica na seção 4.

Figura 3.23 – Pesagem da estrutura do quadrirrotor antes da furação.



Fonte: Elaboração do próprio autor.

A fixação da bateria na estrutura se deu por meio do uso de abraçadeiras auto travantes de nylon e um calço de borracha entre ela e a estrutura. Já as fixações das PCIs, apresentadas nas seções 3.1.9.1 e 3.1.9.2, foram efetuadas por meio do uso de parafusos, barras roscadas, porcas e arruelas. Todas as porcas utilizadas são do tipo auto travantes e arruelas de pressão também foram utilizadas. Estes tipos de porcas e arruelas foram utilizados, pois a estrutura estará sujeita a vibração e as porcas não devem se desprender de forma indesejada durante os voos.

A fixação dos motores se deu por meio do uso de parafusos, porcas e de cruzetas específicas para a fixação do tipo de motor utilizado. Já a fixação das hélices aos motores se deu por meio do uso de *spinners*. A Figura 3.24 mostra tanto a cruzeta quanto o *spinner* utilizados para a fixação.

As hélices utilizadas para gerarem a propulsão juntamente com os motores são hélices fabricadas em plástico ABS e são do tipo 1045, ou seja, possuem 10 polegadas de comprimento e passo de 45°. A Figura 3.25 mostra a pesagem do conjunto formado pelo motor, cruzeta de fixação, *spinner* e hélice.

Figura 3.24 – Conjunto motor XXD A2212/13T + cruzeta de fixação + *spinner*.



Fonte: Página do vendedor¹⁶.

Figura 3.25 – Pesagem do conjunto motor XXD A2212/13T + cruzeta de fixação + *spinner* + hélice.



Fonte: Elaboração do próprio autor.

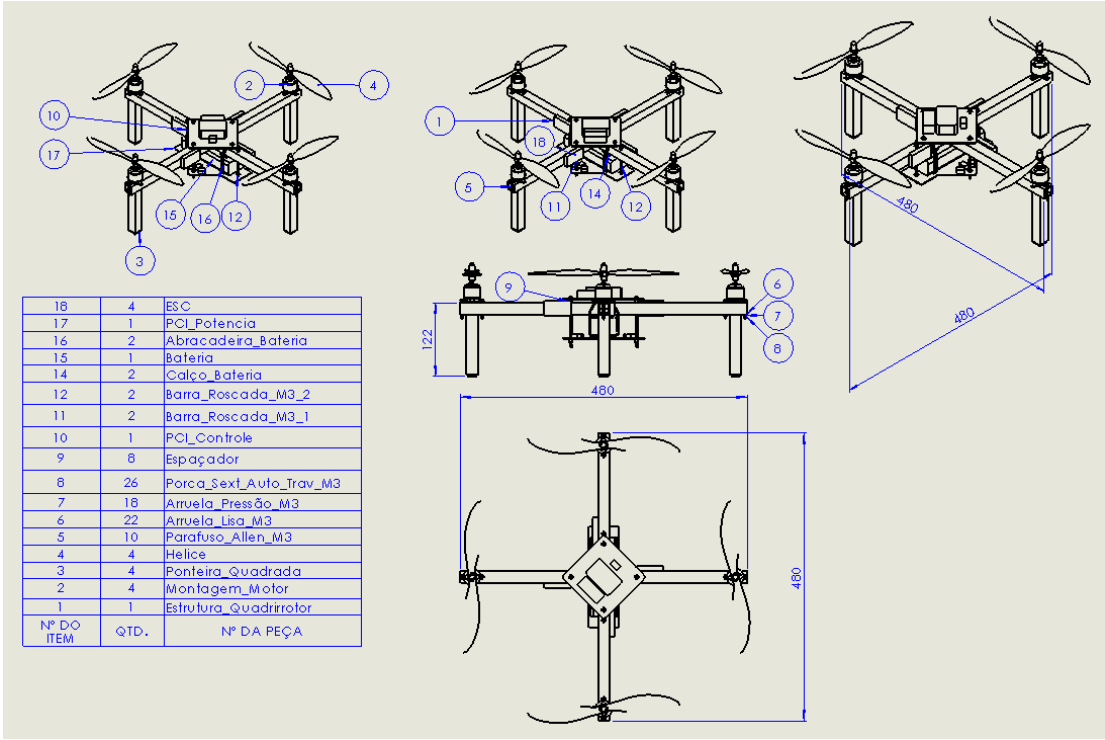
Por fim, a fixação dos ESCs e dos cabos à estrutura também se deram por meio do uso de abraçadeiras auto travantes de nylon.

Como a bateria é uma parte sensível do quadricóptero e pode entrar em combustão caso sofra alguma colisão, ela deve ficar protegida contra qualquer tipo de choque. Este foi o motivo de a PCI apresentada na subseção 3.1.9.2 possuir abas laterais. Ela ficará fixada abaixo da bateria, protegendo-a contra possíveis choques.

¹⁶ Disponível em: <https://produto.mercadolivre.com.br/MLB-1103525590-motor-brusheless-a2212-13t-1000kv-com-spinner-novo-_JM?quantity=1> Acesso em: 20 out 2019.

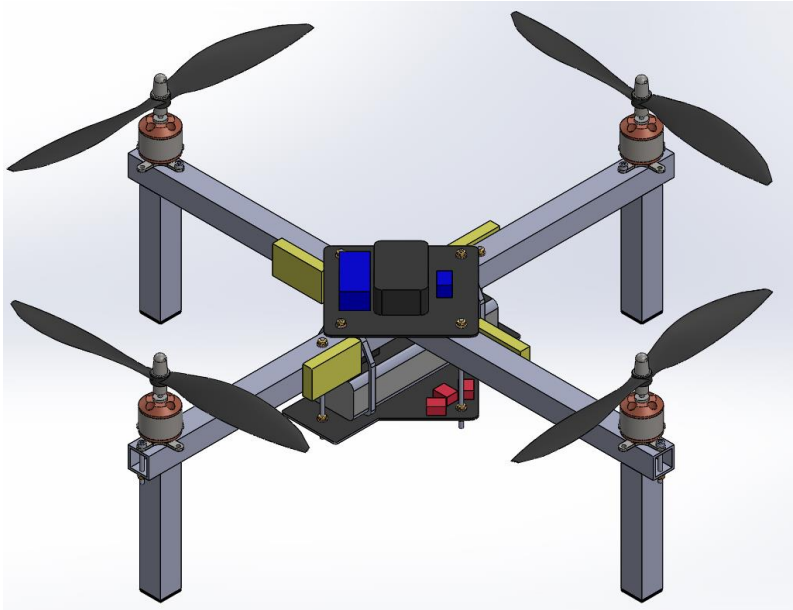
A Figura 3.26 mostra o CAD 2D da montagem completa do quadrirrotor com todos os seus componentes mecânicos e eletrônicos, as Figuras 3.27 e 3.28 o CAD 3D em diferentes visões e a Figura 3.29 o quadrotor real completamente montado.

Figura 3.26 – CAD 2D da montagem completa do quadrirrotor.



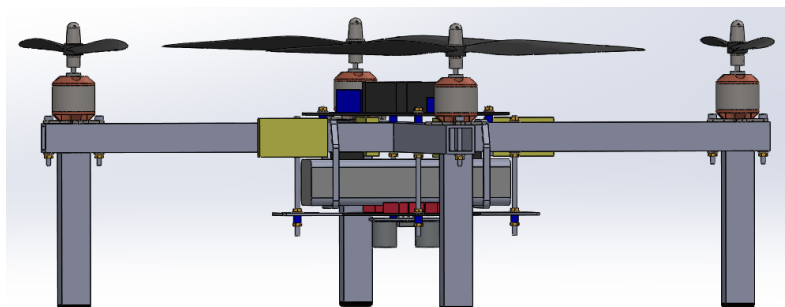
Fonte: Elaboração do próprio autor.

Figura 3.27 – CAD 3D da montagem completa do quadrirrotor: vista em perspectiva.



Fonte: Elaboração do próprio autor.

Figura 3.28 – CAD 3D da montagem completa do quadrirrotor: vista lateral rotacionada em relação ao eixo z.



Fonte: Elaboração do próprio autor.

Figura 3.29 – Quadrirrotor real completamente montado.



Fonte: Elaboração do próprio autor.

A massa total da estrutura toda montada e pronta para o voo medida foi de 1298,00 gramas. Se comparada com os modelos comerciais com a mesma envergadura, como é caso do DJI Phantom 4 Advanced que possui 1380 gramas, esta massa está coerente.

3.3 Projeto Computacional

O projeto computacional do quadrrrotor envolve tanto o hardware descrito na subseção 3.1 quanto os softwares desenvolvidos. O projeto dos softwares desenvolvidos pode ser dividido em duas partes: software do quadrrrotor e software do controle remoto. As subseções seguintes descrevem cada uma destas partes.

3.3.1 Software do quadrrrotor

O software do quadrrrotor é executado na SBC descrita na subseção 3.1.1. Este software é composto por um SO (Sistema Operacional) e um software desenvolvido exclusivamente para este projeto. Como uma das premissas deste projeto é que ele seja *open source*, um SO *open source* foi utilizado. O SO escolhido para este projeto foi o Linux Debian que já possui uma versão customizada especificamente para a SBC utilizada e pode ser encontrada na página do fabricante da SBC¹⁷.

O SO é responsável pelo gerenciamento dos recursos de hardware da SBC. Entre eles a memória RAM e externa, *dongle* WiFi para estabelecer a comunicação entre a SBC e controle remoto, periféricos de I2C, entradas e saídas digitais, saídas PWM, entre outros. Ele também é o responsável por alocar e gerenciar recursos para todas as aplicações que são executadas no processador.

O software desenvolvido é responsável por realizar o controle do quadrrrotor. Para executar tal ação, ele realiza leituras e conversões dos dados contidos da IMU e calcula como os atuadores devem ser excitados para que os ângulos de rotação desejados sejam alcançados. Para realizar a seleção de qual tipo de controlador deve ser empregado, ele realiza uma leitura na chave seletora (DIP *switch*) presente na PCI de controle apresentada na subseção 3.1.9.1.

Para a implementação dos controladores em software, os controladores que serão apresentados na seção 6, que já estão discretizados e expressos em termos de equação de diferenças para possibilitarem a implementação em um computador digital, foram convertidos em código.

A linguagem de programação utilizada para a implementação do software foi a linguagem C++. Ela foi escolhida por ser livre, compilada e orientada a objetos.

¹⁷Disponível em: <<http://beagleboard.org/latest-images>> Acesso em: 20 out 2019.

Já o compilador utilizado foi o g++ para processadores com arquitetura arm e unidades de processamentos de pontos flutuantes, o “g++-arm-linux-gnueabi”. Este compilador também foi escolhido por ser *open source*.

Uma linguagem orientada a objetos foi escolhida a fim de facilitar e agilizar o desenvolvimento. Conceitos como herança e polimorfismo foram utilizados neste projeto. O diagrama de classes do software do quadricóptero desenvolvido pode ser visto na Figura 3.30. Nele é possível ver tais conceitos sendo empregados.

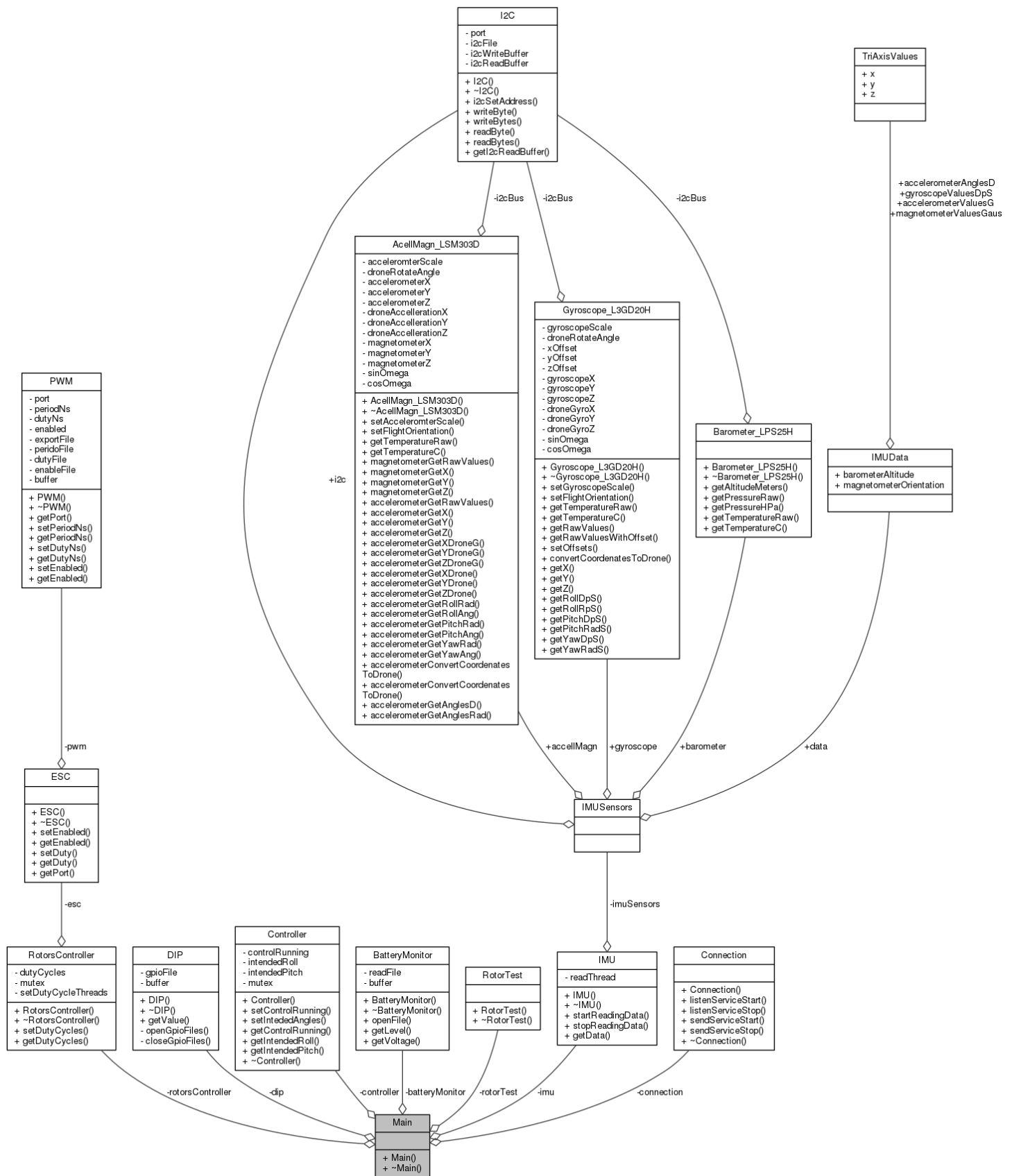
Como pode ser visto na Figura 3.30, a estrutura do software é composta por várias classes e objetos. A Main é a responsável por inicializar todo o processo e realizar a alocação e execução de todos os objetos e *threads* presentes no sistema.

A classe RotorController é responsável por realizar o controle de velocidade de forma independente em cada rotor. Para isso, ela instancia um objeto da classe ESC que realizam o controle dos rotores através do envio de comandos para os ESCs que por sua vez instancia um objeto da classe PWM que é responsável por manipular, junto com o SO, os parâmetros da saída PWM, tais como frequência, largura de pulso e habilitação. Quatro objetos da classe RotorController são instanciados pela Main, um para cada rotor.

A classe DIP é responsável por realizar a leitura da chave seletora (DIP *switch*) para definir qual tipo de controlador será implementado pela classe Controller. A classe Controller é uma classe abstrata que pode ser implementada por algumas das classes que ela herda, como, PIDCross, LQRCross ou BacksteppingCross. Assim que ela é instanciada com algumas das classes que ela herda, ela é responsável por realizar o cálculo dos comandos de saídas que serão enviados para o motor através da comunicação com a classe ESC através da Main. Cada uma destas classes herdadas implementa em software o respectivo controlador discreto apresentado na seção 6. Ela utiliza como entradas os dados das classes IMU, do BatteryMonitor e Connection (comandos de ângulos desejados).

A classe BatteryMonitor é responsável pela leitura do nível de tensão da bateria a partir da leitura de um pino analógico da SBC, que é a saída de um divisor resistivo presente na PCI de controle. A partir do nível de tensão de bateria, é possível estimar quanto de carga a bateria possui e se é seguro continuar em voo. No caso da bateria utilizada, um nível de tensão que foi considerado seguro foi de 10,6 V, que corresponde a aproximadamente 3,53 volts por célula. Este dado de nível de tensão na bateria também é utilizado na classe Controller.

Figura 3.30 – Diagrama de Classes do software do quadricóptero.



Fonte: Elaboração do próprio autor.

A classe IMU é responsável por realizar a configuração dos sensores e leituras de seus dados. Como a IMU possui três sensores, existe uma classe por sensor. A classe AcellMagn_LSM303D é responsável por comunicar com o sensor LSM303D que possui um acelerômetro nos três eixos e um magnetômetro nos três eixos. A classe Gyroscope_L3GD20H é responsável por comunicar com o sensor L3GD20H que possui um sensor giroscópico nos 3 eixos. Por fim, a classe Barometer_LPS25H é responsável pela comunicação com o sensor barométrico LPS25H. Todos estes sensores se comunicam com a SBC através de barramento I2C e por este motivo a classe I2C foi implementada.

Por fim a classe Connection é responsável por estabelecer a comunicação via WiFi com o controle remoto para possibilitar a recepção de comandos como ângulo de arfagem, rolagem e guinada e elevação ou diminuição da altitude. Através desta comunicação o software do quadrirrotor envia alguns dados de telemetria como ângulos de rolagem e arfagem, orientação (pontos cardeais), nível de bateria do quadrirrotor, altitude e tipo de controlador implementado.

3.3.2 Software do controle remoto

Como mencionado na subseção 3.1.8.1, o controle remoto é constituído por um smartphone. Um aplicativo foi desenvolvido especificamente para este propósito e é responsável por manter o link de comunicação WiFi com o quadrirrotor, enviar comandos de movimentação para o quadrirrotor e monitorar os dados que periodicamente são enviados pelo quadrirrotor.

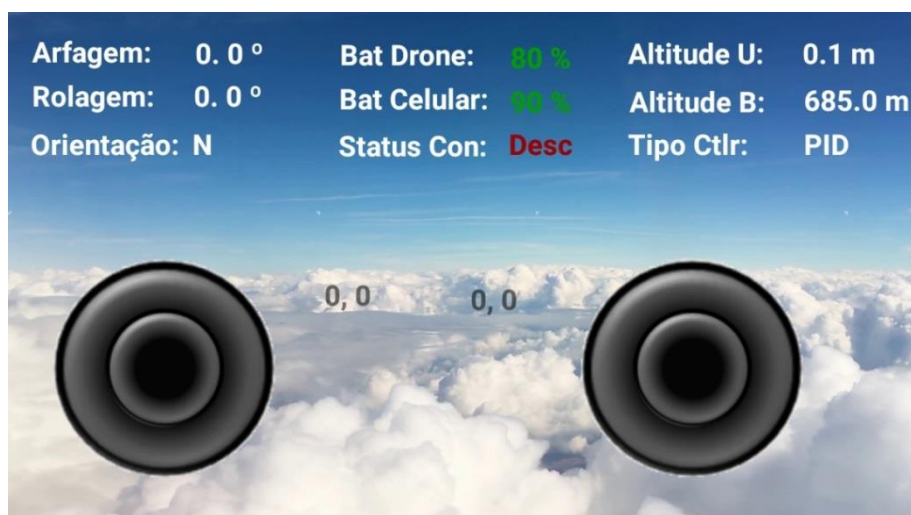
O sistema do smartphone utilizado é composto por um sistema operacional Android e um aplicativo desenvolvido em Java. Desta forma, a aplicação pode ser executada em qualquer smartphone que possua este SO com uma versão superior a mínima utilizada no desenvolvimento. A versão utilizada foi a 6.0.

O smartphone executa também a função de *access point* (ponto de acesso), ou seja, ele provê uma rede WiFi no qual o *dongle* WiFi da PCI de controle se conecta. O número máximo de conexões foi limitado para que não haja o risco de rede ficar sobrecarregada visto que a principal função desta rede é manter a conexão apenas entre o smartphone e o quadrirrotor.

Um captura de tela da aplicação em execução pode ser vista na Figura 3.31. Nela, é possível ver todos os dados de telemetria, os comandos enviados para o

quadrrrotor e os joysticks virtuais criados para facilitar o manuseio por parte do usuário.

Figura 3.31 – Captura de tela da aplicação do controle remoto.



Fonte: Elaboração do próprio autor.

A fim de reduzir a latência de comunicação, um protocolo binário enxuto foi criado para estabelecer a troca de mensagens entre o controle remoto e o quadrrrotor.

As mensagens que podem ser enviadas do controle remoto para o quadrrrotor podem ser vistas na Tabela 3.5 e 3.6, onde MTP é o tipo de mensagem utilizado no *parser* (analisador) da mensagem. Já a mensagem que pode ser enviada do quadrrrotor para o controle remoto pode ser vista na Tabela 3.7. Abaixo de cada uma das tabelas há um descritivo de o que significa cada um dos campos da mensagem.

Tabela 3.5 – Comando de movimento (5 bytes).

MTP = 0x00	Rolagem	Arfagem	Subida/Descida	Guinada
1 byte	1 byte	1 byte	1 byte	1 byte

Os valores dos campos de Rolagem e Arfagem podem variar entre -100 e 100, onde este número representa a porcentagem em relação ao máximo ângulo permitido que se deseja alcançar. O campo de Subida/Descida pode variar entre -1 e 1, onde -1 representa descida, 0 altitude estável e 1 subida. Por fim, o campo de

Guinada também pode variar entre -100 e 100 onde números negativos representam movimento no sentido anti-horário em relação ao eixo z na porcentagem em relação à taxa máxima permitida, 0 a estabilização em relação a esta rotação e números positivos movimentos no sentido horário na porcentagem da taxa máxima permitida.

Tabela 3.6 – Comando de decolagem/pouso (2 bytes).

MTP = 0x01	Comando
1 byte	1 byte

O campo Comando pode variar entre -1 e 1, onde -1 representa que os motores devem ser desligados, 0 representa um rotação constante nos motores mas ainda no solo e 1 representa decolagem.

Tabela 3.7 – Mensagem de telemetria (9 bytes).

MTP=0x03	Bateria	Rolagem	Arfagem	Guinada	Altitude (US)	Altitude (Bar)	Potência
1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	2 byte	1 byte

Por fim, na mensagem de telemetria, o campo Bateria representa o nível atual de bateria do quadricóptero (0 a 100%), os campos de Rolagem e Arfagem representam a porcentagem do valor máximo que os dois podem assumir (-100% a 100%), o campo de Guinada representa o ângulo de guinada no sentido horário em relação ao norte com uma precisão de 2° (valores entre 0 e 180), a Altitude(US) representa a altitude medida pelo sensor ultrassônico, a Altitude(Bar) a altitude medida pelo sensor barométrico e a Potência representa a porcentagem da soma de potência aplicada a todos os motores.

Na próxima seção, é abordada a modelagem dinâmica do quadricóptero, que é utilizada no projeto dos controladores, que por sua vez são utilizados nos controladores implementados em software apresentados nesta seção.

4 Modelagem Dinâmica do Quadrrrotor Projetado

As subseções seguintes descrevem os passos seguidos para obtenção de um modelo matemático que representa a cinemática e dinâmica do quadrrrotor desenvolvido a fim de que seja possível, a partir do modelo encontrado, projetar os controladores.

Para tal modelagem, foram assumidas que a Terra é plana e estacionária. Logo, suas acelerações de translação e rotação serão consideradas nulas. Esta suposição é válida para quadrrrotores visto que suas autonomias de voo são pequenas (percorrem curtas distâncias) e suas velocidades de deslocamento são muito inferiores às velocidades de rotação e translação da Terra.

Também foi assumido que o quadrrrotor não possui superfícies aerodinâmicas de elevação e, por consequência, as forças e momentos aerodinâmicos foram consideradas desprezíveis.

Alguns dos conceitos e fundamentos sobre a modelagem da dinâmica de quadrrrotores aqui expostos foram extraídos do trabalho de Beard (2008), Stevens e Lewis (2003) e Alves (2012).

4.1 Cinemática

A cinemática é estudo do movimento de um objeto sem levar em consideração os agentes que causam este movimento. As subseções seguintes tratam da cinemática de um quadrrrotor.

4.1.1 Sistemas de Coordenadas

O primeiro passo tomado na modelagem matemática foi a definição dos sistemas de coordenadas. Três referenciais foram adotados: o referencial inercial, o referencial do veículo e o referencial fixo ao corpo.

O sistema de coordenadas inercial, F^i , é um sistema de coordenadas fixo à Terra, com origem definida na posição inicial da aeronave: ponto de decolagem. O vetor unitário $\hat{i}^i$ aponta para o leste, $\hat{j}^i$ aponta para norte e $\hat{k}^i$ aponta para cima (sentido contrário ao centro da Terra).

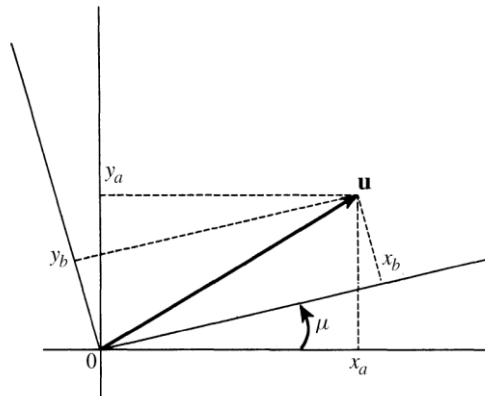
O sistema de coordenadas do veículo, F^v , possui sua origem no centro de massa do quadrirrotor. Este sistema está alinhado com F^i , de forma que $\hat{i}^v$ aponta para o rotor 1, $\hat{j}^v$ apontando para o rotor 2 e $\hat{k}^v$ apontando para o “teto” do quadrirrotor. Este sistema pode ser visto na Figura 1.7.

Por fim, o sistema de coordenadas fixo ao corpo, F^b , também possui sua origem no centro de massa do quadrirrotor. A orientação de F^b pode ser obtida a partir do F^v por meio de três rotações consecutivas, denominadas ângulos de Euler, segundo a sequência ZYX, indicadas, respectivamente, por ψ , θ e φ .

4.1.2 Relação entre os sistemas de coordenadas por meio de ângulos de Euler

Como mencionado anteriormente, a orientação do F^b pode ser obtida a partir do F^v por meio de três rotações consecutivas. Estas rotações são realizadas em três planos coordenados sucessivos e, por isso, também são denominadas rotações planas. A Figura 4.1 mostra um exemplo de uma rotação plana. Neste exemplo, o sistema de coordenada F^b está rotacionado em torno de $\hat{k}^a$ por um ângulo μ .

Figura 4.1 – Rotação plana de coordenadas.



Fonte: Stevens e Lewis (2003).

Ao aplicar as três rotações consecutivas na ordem ZYX, a transformação entre o sistema de coordenadas do veículo e o fixo ao corpo é dada por:

$$R_v^b = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

$$R_v^b = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\varphi\sin\theta\cos\psi - \cos\varphi\sin\psi & \sin\varphi\sin\theta\sin\psi + \cos\varphi\cos\psi & \sin\varphi\cos\theta \\ \cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi & \cos\varphi\sin\theta\sin\psi - \sin\varphi\cos\psi & \cos\varphi\cos\theta \end{bmatrix} \quad (4.2)$$

Para uma dada orientação, os ângulos de Euler não são únicos. A ambiguidade na escolha dos ângulos pode ser eliminada restringindo-se as rotações aos seguintes intervalos:

$$-\pi \leq \varphi < \pi \text{ ou } 0 \leq \varphi < 2\pi$$

$$-\frac{\pi}{2} \leq \theta < \frac{\pi}{2}$$

$$-\pi \leq \psi < \pi \text{ ou } 0 \leq \psi < 2\pi$$

Para o projeto dos controladores do quadrrorotor, serão assumidos que $-\frac{\pi}{2} \leq \varphi < \frac{\pi}{2}$, $-\frac{\pi}{2} \leq \theta < \frac{\pi}{2}$ e $-\pi \leq \psi < \pi$.

Visto que o sistema de coordenadas F^v está alinhado com F^i , sendo apenas sua origem translada em relação à outra, pode-se afirmar que as matrizes de rotação que os relacionam ao sistema de coordenadas F^b são iguais. Portanto:

$$R_i^b = R_v^b \quad (4.3)$$

Definindo (p_e, p_n, p_u) como a posição inercial do quadrrorotor ao longo $(\hat{i}^i, \hat{j}^i, \hat{k}^i)$ respectivamente em F^i e (u, v, w) como velocidade do quadrrorotor medida ao longo de $(\hat{i}^b, \hat{j}^b, \hat{k}^b)$ respectivamente em F^b e fazendo uso da Equação (4.3), é possível chegar à seguinte relação:

$$\frac{d}{dt_i} \begin{bmatrix} p_e \\ p_n \\ p_u \end{bmatrix} = R_b^i \begin{bmatrix} u \\ v \\ w \end{bmatrix} = (R_i^b)^T \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (4.4)$$

$$\begin{bmatrix} \dot{p}_e \\ \dot{p}_n \\ \dot{p}_u \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & \sin\varphi\sin\theta\cos\psi - \cos\varphi\sin\psi & \cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi \\ \cos\theta\sin\psi & \sin\varphi\sin\theta\sin\psi + \cos\varphi\cos\psi & \cos\varphi\sin\theta\sin\psi - \sin\varphi\cos\psi \\ -\sin\theta & \sin\varphi\cos\theta & \cos\varphi\cos\theta \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (4.5)$$

4.1.2.1 Relação entre velocidades angulares definidas em dois referenciais expressos em termos dos ângulos de Euler

Para definir a relação entre as velocidades angulares definidas em dois referenciais, dois referenciais intermediários serão utilizados, F^α e F^β , onde F^α é definido através de uma rotação de um ângulo φ em relação à F^i , F^β é definido através de uma rotação de um ângulo θ em relação à F^α e F^b é definido através de uma rotação de um ângulo ψ em relação à F^β .

Através do uso da propriedade aditiva da velocidade angular, é possível realizar o seguinte desenvolvimento:

$$\omega_{b/i} = \omega_{b/\beta} + \omega_{\beta/\alpha} + \omega_{\alpha/i} \quad (4.6)$$

$$\omega_{b/i} = R_{\beta}^b(\dot{\phi}) \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_{\beta}^b(\dot{\phi}) R_{\alpha}^{\beta}(\dot{\phi}) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_{\beta}^b(\dot{\phi}) R_{\alpha}^{\beta}(\dot{\phi}) R_i^{\alpha}(\dot{\psi}) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad (4.7)$$

$$\omega_{b/i} = \dot{\phi} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \dot{\theta} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + \dot{\psi} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (4.8)$$

$$\omega_{b/i} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\varphi & \sin\varphi\cos\theta \\ 0 & -\sin\varphi & \cos\varphi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (4.9)$$

Realizando a inversão da matriz, a relação pode ser reescrita como:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\varphi\tan\theta & \cos\varphi\tan\theta \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi\sec\theta & \cos\varphi\sec\theta \end{bmatrix} \omega_{b/i} \quad (4.10)$$

Esta inversão aplicada em (4.10) é válida desde que $\theta \neq \frac{\pi}{2} + k\pi$. Desta forma, levando-se em consideração às restrições impostas para eliminar as ambiguidades, será assumido que $-\frac{\pi}{2} < \varphi < \frac{\pi}{2}$, $-\frac{\pi}{2} < \theta < \frac{\pi}{2}$ e $-\pi \leq \psi < \pi$.

Definindo (p, q, r) como as taxas de rolagens ao longo de F^b em relação à F^i , na sequência XYZ, a Equação (4.10) pode ser reescrita como:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\varphi\tan\theta & \cos\varphi\tan\theta \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi\sec\theta & \cos\varphi\sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4.11)$$

4.1.3 Relação entre as derivadas de um vetor em dois sistemas de referência

A equação de Coriolis, Stevens e Lewis (2003), relaciona as derivadas de um vetor $\mathbf{v}$ medidas em dois sistemas distintos. No caso dos referenciais definidos para descrever a cinemática do quadrirrotor, a Equação pode ser definida por:

$$\frac{d}{dt_i} \mathbf{v}_i = \frac{d}{dt_b} \mathbf{v}_b + \omega_{b/i} \times \mathbf{v}_i \quad (4.12)$$

onde $\omega_{b/i}$ representa a velocidade angular vetorial instantânea de F^b em relação a F^i .

Uma propriedade interessante sobre este vetor de velocidade angular é que

$$\frac{d}{dt_i} \omega_{i/b} = \frac{d}{dt_b} \omega_{b/i}, \text{ ou seja, sua derivada é igual em ambos os sistemas.}$$

4.1.4 Relação entre as velocidades e acelerações em dois sistemas de referência

Na Figura 4.2 é mostrado um ponto p no espaço se movendo em relação aos referenciais F^i e F^b onde se localizam os pontos fixos O e Q respectivamente. A relação entre a posição dos vetores mostrados na Figura 4.2 pode ser dada por:

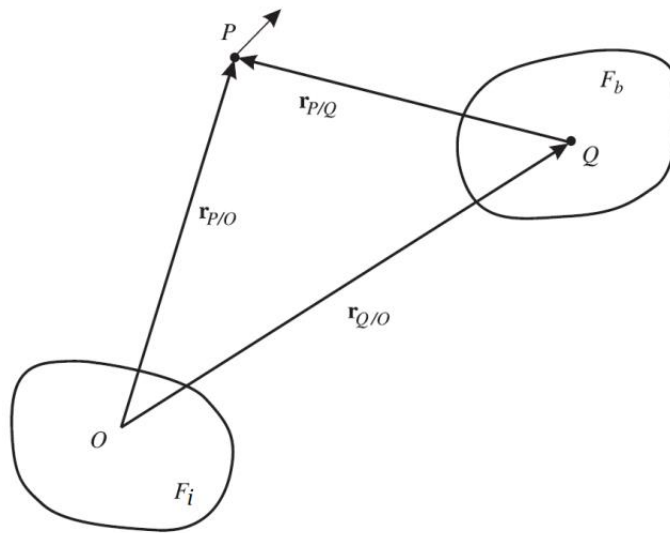
$$\mathbf{r}_{P/O} = \mathbf{r}_{P/Q} + \mathbf{r}_{Q/O} \quad (4.13)$$

Tomando a derivada de (4.13) em relação à F^i e aplicando a Equação de Coriolis descrita em (4.12), tem-se:

$$\mathbf{v}_{P/i} = \mathbf{v}_{P/b} + \mathbf{v}_{Q/i} + \omega_{b/i} \times \mathbf{r}_{P/Q} \quad (4.14)$$

onde $\mathbf{v}_{m/n}$ representa a velocidade do ponto m medida de um observador em n .

Figura 4.2 – Ponto em movimento em dois sistemas de referência em movimento.



Fonte: Adaptado de Stevens e Lewis (2003).

Tomando a derivada de (4.14) em relação à F^i e aplicando-se novamente a equação de Coriolis, tem-se:

$$\mathbf{a}_{P/i} = \mathbf{a}_{P/b} + \mathbf{a}_{Q/i} + \frac{d}{dt} \omega_{b/i} \times \mathbf{r}_{P/Q} + \omega_{b/i} \times (\omega_{b/i} \times \mathbf{r}_{P/Q}) + 2\omega_{b/i} \times \mathbf{v}_{P/b} \quad (4.15)$$

onde $\mathbf{a}_{m/n}$ representa a aceleração do ponto m medida de um observador em n .

4.2 Dinâmica

O movimento mais geral de um corpo pode ser considerado como uma combinação entre um movimento de rotação e um de translação. Ambos os tipos de movimentos serão considerados nesta seção.

4.2.1 Movimento de Translação

Sendo $\mathbf{v}$ a velocidade do quadricóptero e aplicando-se as leis de Newton referente ao movimento translacional sobre referenciais inerciais, tem-se:

$$m \frac{d\mathbf{v}}{dt_i} = \mathbf{f} \quad (4.16)$$

onde m é a massa do quadricóptero, $\mathbf{f}$ é a força aplicada ao quadricóptero e $\frac{d}{dt_i}$ é a derivada no tempo no referencial inercial. Da Equação de Coriolis (4.12) tem-se:

$$m \frac{d\mathbf{v}}{dt_i} = m \left(\frac{d\mathbf{v}}{dt_b} + \boldsymbol{\omega}_{b/i} \times \mathbf{v} \right) = \mathbf{f} \quad (4.17)$$

Segundo Beard (2012), expressando a Equação (4.17) nas coordenadas do referencial fixo ao corpo, onde $\mathbf{v}^b \triangleq (u, v, w)^T$ e $\boldsymbol{\omega}_{b/i}^b \triangleq (p, q, r)^T$, tem-se:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - rv \\ qu - pv \end{bmatrix} + \frac{1}{m} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - rv \\ qu - pv \end{bmatrix} + \frac{1}{m} \mathbf{f}^b \quad (4.18)$$

onde $\mathbf{f}^b \triangleq [f_x, f_y, f_z]^T$.

4.2.2 Movimento de Rotação

Para o movimento de rotação, a segunda lei de Newton diz que

$$\frac{d\mathbf{h}^b}{dt_i} = \mathbf{m} \quad (4.19)$$

onde $\mathbf{h}$ é o momento angular e $\mathbf{m}$ é o torque aplicado. Aplicando a Equação de Coriolis (4.12) em (4.19), tem-se:

$$\frac{d\mathbf{h}}{dt_i} = \frac{d\mathbf{h}}{dt_b} + \boldsymbol{\omega}_{b/i} \times \mathbf{h} = \mathbf{m} \quad (4.20)$$

Solucionando a Equação (4.20) através do uso das coordenadas fixa ao corpo, onde $\mathbf{h}^b = J\boldsymbol{\omega}_{b/i}^b$, a matriz constante de inércia J é dada por:

$$J = \begin{bmatrix} \int (y^2 + z^2) dm & -\int xy dm & -\int xz dm \\ -\int xy dm & \int (x^2 + z^2) dm & -\int yz dm \\ -\int xz dm & -\int yz dm & \int (x^2 + y^2) dm \end{bmatrix} \triangleq \begin{bmatrix} J_{xx} & -J_{xy} & -J_{xz} \\ -J_{xy} & J_{yy} & -J_{yz} \\ -J_{xz} & -J_{yz} & J_{zz} \end{bmatrix} \quad (4.21)$$

Pelo fato de a matriz J ser simétrica e possuir todos seus elementos reais, sua inversa é dada por:

$$J^{-1} = \begin{bmatrix} k_{xx} & k_{xy} & k_{xz} \\ k_{xy} & k_{yy} & k_{yz} \\ k_{xz} & k_{yz} & k_{zz} \end{bmatrix} \quad (4.22)$$

Para encontrar os momentos de inércia no quadrrrotor, as massas de cada componente foram individualmente configuradas no CAD 3D do quadrrrotor apresentado na subseção 3.2. Como o software CAD utilizado possui um recurso de propriedade de massa de peças e montagens que retorna tanto a posição do centro de massa quanto a matriz de momento de inércia, a mesma foi utilizada para encontrar a matriz do quadrrrotor desenvolvido. Este método também foi utilizado no trabalho de Kiyashko, Sholmov e Efremov (2017). A captura de tela do recurso aplicado encontra-se no Apêndice A. A matriz momento de inércia retornada foi:

$$J = \begin{bmatrix} 0,013577 & -0,000003 & 0,000011 \\ -0,000003 & 0,013993 & -0,000022 \\ 0,000011 & -0,000022 & 0,024380 \end{bmatrix} kg \cdot m$$

que invertendo se torna:

$$J^{-1} = \begin{bmatrix} 73,65400 & 0,015740 & -0,033218 \\ 0,015740 & 71,464400 & 0,064481 \\ -0,033218 & 0,064481 & 41,017300 \end{bmatrix} (kg \cdot m)^{-1}$$

Definindo $\mathbf{m}^b \triangleq (\tau_\varphi, \tau_\theta, \tau_\psi)$, onde $(\tau_\varphi, \tau_\theta, \tau_\psi)$ representam os torques nos três eixos de coordenadas de rotação, a Equação (4.20) pode ser reescrita em coordenadas do corpo como:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} k_{xx} & k_{xy} & k_{xz} \\ k_{xy} & k_{yy} & k_{yz} \\ k_{xz} & k_{yz} & k_{zz} \end{bmatrix} \left(\begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \begin{bmatrix} J_{xx} & -J_{xy} & -J_{xz} \\ -J_{xy} & J_{yy} & -J_{yz} \\ -J_{xz} & -J_{yz} & J_{zz} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \begin{bmatrix} \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \right) \quad (4.23)$$

4.3 Forças e Momentos no Quadrrrotor

As forças e momentos presentes no quadrrrotor são ocasionados principalmente devido à gravidade e aos quatro propulsores. Conforme foi apresentado na Figura 1.7, cada conjunto hélice-rotor produz uma força F para cima e um torque τ contrário ao sentido de rotação da hélice. Desta forma, o total de forças produzido pelo quadrrrotor é dado por:

$$F = F_1 + F_2 + F_3 + F_4 \quad (4.24)$$

No caso mostrado na Figura 1.7, os rotores 1 e 3 rotacionam no sentido anti-horário e os rotores 2 e 4 no sentido horário. O torque de rolagem é produzido pelas forças dos rotores 2 e 4 e o de arfagem pelos rotores 1 e 3. Logo:

$$\tau_\phi = l(F_2 - F_4) \quad (4.25)$$

$$\tau_\theta = l(F_1 - F_3) \quad (4.26)$$

Devido à terceira lei de Newton, os arrastos atuando nas hélices dos propulsores produzem um torque de guinada no corpo do quadricóptero. Logo:

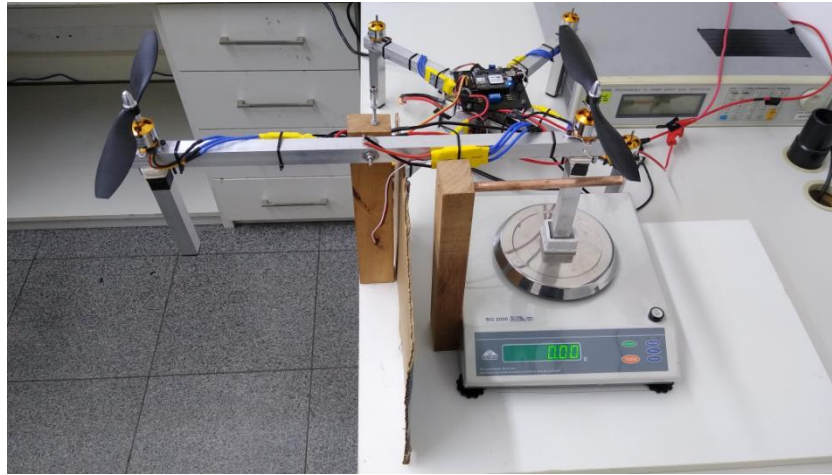
$$\tau_\psi = \tau_2 + \tau_4 - \tau_1 - \tau_3 \quad (4.27)$$

A sustentação e o arrasto produzidos pelos propulsores são proporcionais ao quadrado da velocidade angular. Para determinar a relação entre as forças e torques gerados pelos rotores e os comandos de modulação em largura de pulso, os experimentos apresentados nas subseções 4.3.1 e 4.3.2 foram desenvolvidos. Esta determinação é necessária para que o controlador seja capaz de calcular os comandos que devem ser aplicados aos motores para que a atitude desejada seja executada.

4.3.1 Relação entre força gerada e comando aplicado ao motor

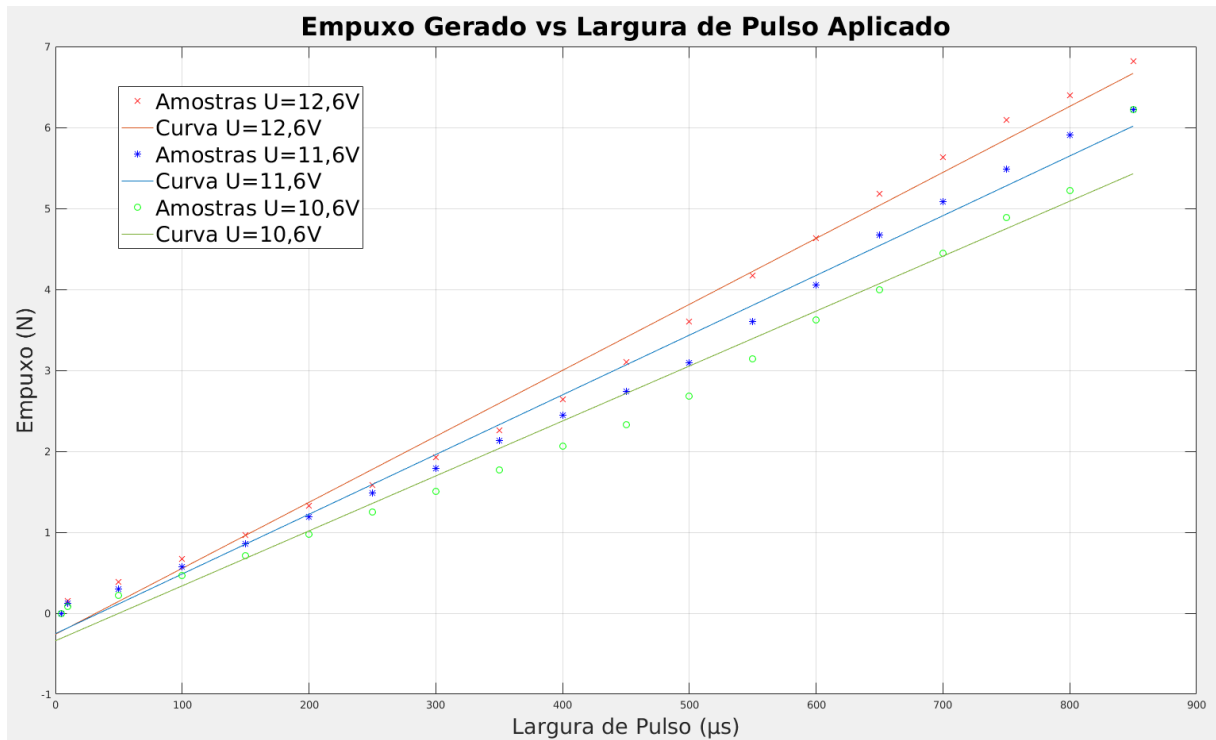
A Figura 4.3 mostra a estrutura utilizada para a medição do fator de empuxo, K_F . A estrutura se consiste de um braço do quadricóptero com movimentação livre em relação ao eixo de rolagem por meio de um eixo passante composto por um movimento rolamentado a fim de reduzir o atrito. Para a realização de tal medida, foi utilizada uma balança de precisão e apenas um dos rotores foi colocado em operação (rotor oposto à balança). A PCI de controle enviou sinais de comando PWM para o ESC deste rotor e uma varredura na largura de pulso permitida para o controle do ESC foi realizada. As varreduras foram efetuadas sob tensões constantes e três varreduras foram efetuadas: em 10,6V, 11,6V e 12,6V. As curvas que representam as medidas de empuxos gerados nas três varreduras podem ser vistas na Figura 4.4. Já na Figura 4.5 é possível visualizar as correntes drenadas pelo motor em função das larguras de pulso aplicadas, e conseqüente, dos empuxos gerados. Estes valores de tensões foram utilizados já que este será o *range* de tensão de bateria utilizado para a operação do quadricóptero: de 10,6V a 12,6V.

Figura 4.3 – Estrutura desenvolvida para medição da constante de empuxo.



Fonte: Elaboração do próprio autor

Figura 4.4 – Empuxo gerado pelo conjunto rotor-hélice vs largura de pulso aplicada.



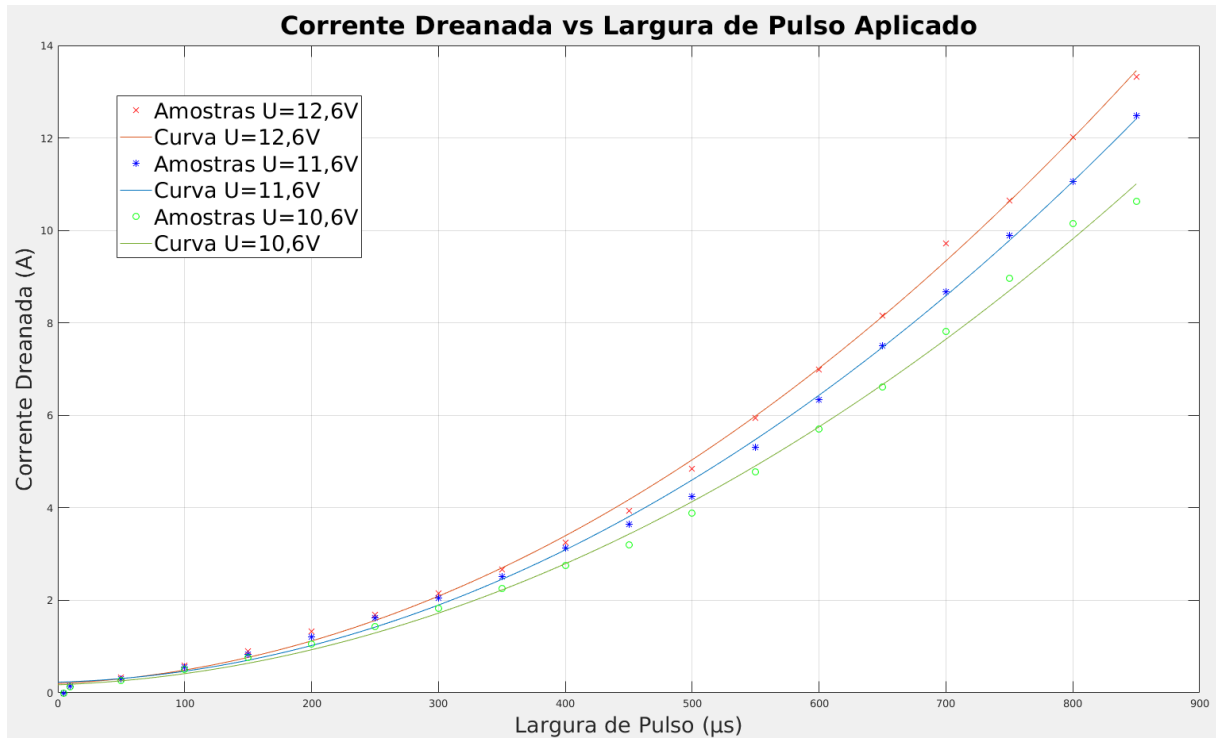
Fonte: Elaboração do próprio autor

Através da análise da Figura 4.4, é possível notar que as medidas realizadas se aproximam de retas. Porém, também é possível notar que a inclinação da reta varia dependendo da tensão de entrada. Logo, K_F não é constante e depende da tensão de entrada. Desta forma, a seguinte simplificação foi adotada:

$$K_F = K_{F_0} U \quad (4.28)$$

onde K_{F_0} é uma constante e U é a tensão da bateria.

Figura 4.5 – Corrente drenada pelo motor vs largura de pulso aplicado.



Fonte: Elaboração do próprio autor

Portanto, a força de empuxo produzida pelo rotor i pode ser expressa por:

$$F_i = K_F \delta_i \quad (4.29)$$

onde δ_i é a largura do sinal de comando aplicado ao ESC i e i varia entre 1 e 4.

Para a determinação do valor da constante K_{F_0} , uma reta foi traçada entre os pontos medidos para cada uma das tensões através de uma aproximação linear de ordem 1. Desta forma, foi encontrado que $K_{F_0} = 0,641 \cdot 10^{-3} \text{ N/V}\mu\text{s}$. O valor de g utilizado para a conversão da medida da balança em força foi de $9,81 \text{ m/s}^2$.

Através da análise da Figura 4.5, é possível notar que a corrente drenada pelo rotor possui uma relação quadrática com a largura de pulso aplicada e, consequentemente, com o empuxo gerado. Isso é esperado visto que o ESC controla o empuxo gerado pelo motor que é proporcional ao quadrado da velocidade de rotação do rotor. Através da curva da corrente, também é possível notar que quanto menor a largura de pulso utilizada, menor o consumo de corrente, o que faz com que trabalhar nesta região de menor consumo seja desejado para aumento da autonomia de voo. No caso do quadrirrotor projetado neste trabalho, que possui uma massa total de 1,298kg, cada rotor estará consumindo cerca de 4A em voo pairado, ou seja, o conjunto total consumirá na ordem de 16A, o que faz com que a

autonomia de voo para a bateria utilizada seja de no máximo 19 minutos e meio. Também é importante notar que, segundo a curva de eficiência do motor apresentada na Figura 3.18, esta corrente é a que oferece maior eficiência energética. Logo, os motores tendem a trabalhar em seus melhores regimes de eficiência durante voos pairados.

Outro ponto importante a se notar é que, embora a curva de empuxo se aproxime de uma reta, ela não é uma reta. Desta forma, o desempenho dos controladores desenvolvidos pode ser afetado devido a esta não linearidade dos motores utilizados. O mesmo é válido para o torque, que é abordado na seção seguinte.

4.3.2 Relação entre torque gerado e comando aplicado ao motor

Para a medição do fator de torque, k_t , a estrutura apresentada na Figura 4.6 foi utilizada. A estrutura é muito semelhante à utilizada para a medição de K_F . Porém, a movimentação se dá em relação ao eixo de guinada. Para a realização das medidas, os mesmos passos utilizados para medições do fator de empuxo foram adotados. A Figura 4.7 apresenta as curvas que representam as medidas dos torques gerados nas três varreduras.

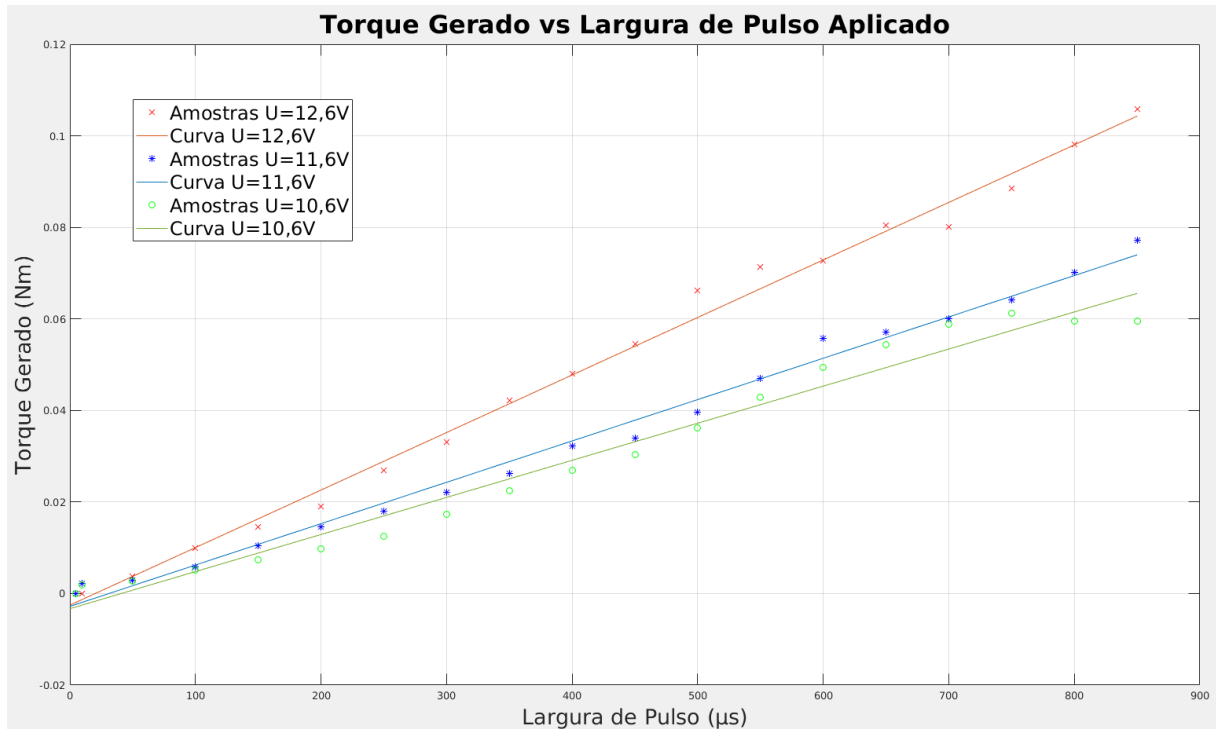
Figura 4.6 – Estrutura desenvolvida para medição da constante de torque.



Fonte: Elaboração do próprio autor

Através da análise da Figura 4.7, também é possível notar que as medidas realizadas se aproximam de retas e que, assim como ocorre no fator de empuxo, o

Figura 4.7 – Torque gerado pelo conjunto rotor-hélice vs largura de pulso aplicado.



Fonte: Elaboração do próprio autor

fator de torque também varia dependendo da tensão de entrada. Logo, k_τ não é constante e também depende da tensão de entrada, porém é proporcional ao quadrado da tensão de entrada. Desta forma, a seguinte simplificação foi adotada:

$$k_\tau = K_{\tau_0} U^2 \quad (4.30)$$

onde K_{τ_0} é uma constante e U é a tensão da bateria.

Portanto, o torque produzido pelo rotor i pode ser expresso por:

$$\tau_i = K_\tau \delta_i \quad (4.31)$$

Para a determinação do valor da constante K_{τ_0} , o mesmo método utilizado para determinar K_{F_0} foi utilizado, o que resultou em $K_{\tau_0} = 0,728 \cdot 10^{-6} \text{ Nm/V}^2 \mu\text{s}$.

4.3.3 Modelagem em função das forças atuantes

A partir da combinação das Equações (4.25) à (4.27), (4.29) e (4.31), as forças e torques existentes no quadrirrotor podem ser escritas em forma de matriz como:

$$\begin{bmatrix} F \\ \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} k_F & k_F & k_F & k_F \\ 0 & lk_F & 0 & -lk_F \\ lk_F & 0 & -lk_F & 0 \\ -k_\tau & k_\tau & -k_\tau & k_\tau \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = \triangleq \mathfrak{M} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} \quad (4.32)$$

As estratégias de controle a serem adotadas irão especificar forças e torques. Desta forma, o comando que será aplicado aos motores pode ser expresso por

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = \mathfrak{M}^{-1} \begin{bmatrix} F \\ \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \quad (4.33)$$

Levando em consideração também a ação da gravidade, chegam-se as seguintes equações que representam a dinâmica do quadrrorotor:

$$\begin{bmatrix} \dot{p}_e \\ \dot{p}_n \\ \dot{p}_u \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & \sin\varphi\sin\theta\cos\psi - \cos\varphi\sin\psi & \cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi \\ \cos\theta\sin\psi & \sin\varphi\sin\theta\sin\psi + \cos\varphi\cos\psi & \cos\varphi\sin\theta\sin\psi - \sin\varphi\cos\psi \\ -\sin\theta & \sin\varphi\cos\theta & \cos\varphi\cos\theta \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (4.34)$$

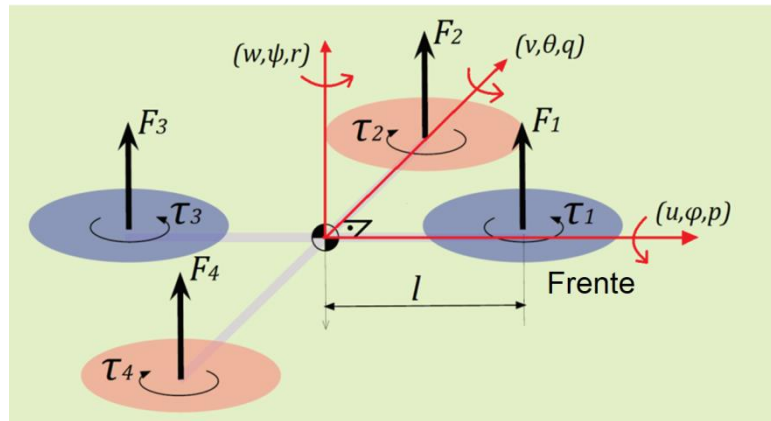
$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - rv \\ qu - pw \end{bmatrix} + \begin{bmatrix} -g \sin\theta \\ g \cos\theta \sin\varphi \\ -g \cos\theta \cos\varphi \end{bmatrix} + \frac{1}{m} \begin{bmatrix} 0 \\ 0 \\ F \end{bmatrix} \quad (4.35)$$

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\varphi \tan\theta & \cos\varphi \tan\theta \\ 0 & \cos\theta & -\sin\varphi \\ 0 & \sin\varphi \sec\theta & \cos\varphi \sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4.36)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} k_{xx} & k_{xy} & k_{xz} \\ k_{xy} & k_{yy} & k_{yz} \\ k_{xz} & k_{yz} & k_{zz} \end{bmatrix} \left(\begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \begin{bmatrix} J_{xx} & -J_{xy} & -J_{xz} \\ -J_{xy} & J_{yy} & -J_{yz} \\ -J_{xz} & -J_{yz} & J_{zz} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \begin{bmatrix} \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \right) \quad (4.37)$$

Todas as componentes que representam a dinâmica do quadrrorotor relacionados ao referencial do corpo podem ser vistos na Figura 4.8.

Figura 4.8 – Esquema de variáveis de estado de um quadrrorotor.



Fonte: Adaptado de Mahony, Kumar e Corke (2012).

4.4 Modelo Simplificado

As equações (4.34) à (4.37) representam as equações do movimento do quadrrorotor. Porém, a fim de simplificar o desenvolvimento dos controladores, algumas simplificações serão adotadas.

Ao analisar a matriz de momentos de Inércia J apresentada ao final da subseção 4.2, é possível notar que os elementos da diagonal são muito mais representativos que os demais. O mesmo é válido para a sua inversa, J^{-1} . Desta forma, a simplificação adotada foi desprezar os produtos de inércia e definir a matriz J e J^{-1} como matrizes diagonais. Desta forma, elas podem ser reescritas como:

$$J = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \quad (4.38)$$

$$J^{-1} = \begin{bmatrix} \frac{1}{J_{xx}} & 0 & 0 \\ 0 & \frac{1}{J_{yy}} & 0 \\ 0 & 0 & \frac{1}{J_{zz}} \end{bmatrix} \quad (4.39)$$

Desta forma, a Equação (4.37) pode ser reescrita como:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{J_{yy}-J_{zz}}{J_{xx}} qr \\ \frac{J_{zz}-J_{xx}}{J_{yy}} pr \\ \frac{J_{xx}-J_{yy}}{J_{zz}} pq \end{bmatrix} + \begin{bmatrix} \frac{1}{J_{xx}} \tau_{\varphi} \\ \frac{1}{J_{yy}} \tau_{\theta} \\ \frac{1}{J_{zz}} \tau_{\psi} \end{bmatrix} \quad (4.40)$$

4.5 Modelo Linearizado

A fim de possibilitar o desenvolvimento de controladores lineares, como é o caso do PID e do LQR, o modelo simplificado foi linearizado através de algumas simplificações.

Limitando os ângulos φ e θ a valores pequenos, a Equação (4.36) pode ser simplificada a:

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4.41)$$

Assumindo que o quadrrrotor rotacione em somente um eixo de coordenada por vez, os termos de Coriolis qr , pr e pq são pequenos. Desta forma, a Equação (4.40) pode ser simplificada a:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{1}{J_{xx}} \tau_{\varphi} \\ \frac{1}{J_{yy}} \tau_{\theta} \\ \frac{1}{J_{zz}} \tau_{\psi} \end{bmatrix} \quad (4.42)$$

Combinando as Equações (4.41) e (4.42), obtém-se:

$$\begin{bmatrix} \ddot{\varphi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{1}{J_{xx}} \tau_{\varphi} \\ \frac{1}{J_{yy}} \tau_{\varphi} \\ \frac{1}{J_{zz}} \tau_{\psi} \end{bmatrix} \quad (4.43)$$

Para a simplificação da posição inercial, será assumido que no ponto de equilíbrio, o ângulo ψ é irrelevante. Desta forma, a Equação (4.34) se torna:

$$\begin{bmatrix} \dot{p}_e \\ \dot{p}_n \\ \dot{p}_u \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\varphi\sin\theta & \cos\varphi\sin\theta \\ 0 & \cos\varphi & -\sin\varphi \\ -\sin\theta & \sin\varphi\cos\theta & \cos\varphi\cos\theta \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (4.44)$$

Novamente, assumindo que o quadrrrotor rotacione em somente um eixo de coordenada por vez, os termos de Coriolis qr , pr e pq são pequenos. Desprezando os termos de Coriolis e inserindo a Equação (4.35) em (4.44), tem-se:

$$\begin{bmatrix} \ddot{p}_e \\ \ddot{p}_n \\ \ddot{p}_u \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \begin{bmatrix} -\cos\theta \sin\varphi \\ \sin\varphi \\ \cos\varphi\cos\theta \end{bmatrix} \frac{F}{m} \quad (4.45)$$

A partir das equações encontradas nesta seção, referentes ao modelo dinâmico do quadrrrotor projetado, é possível projetar os controladores capazes de realizar a estabilização de voo desta aeronave. Os projetos destes controladores, utilizando diferentes técnicas, são apresentados na próxima seção.

5 Controle Contínuo

Nesta seção são apresentadas técnicas de controle contínuo para o controle de estabilização de voo do quadrirrotor. Os controladores discretos apresentados na seção 6 farão uso destes controladores, discretizando-os para que seja possível implementá-los em software para poderem ser utilizados no modelo real.

Como o foco deste trabalho está no controle de estabilização de voo, os controladores aqui projetados têm como objetivo o controle dos ângulos de Euler φ e θ . As técnicas de controle utilizadas para o controle destes ângulos foram PID, LQR e *Backstepping* e são apresentadas nas subseções 5.1, 5.2 e 5.4, respectivamente. Visto que, segundo a modelagem apresentada, o quadrirrotor é simétrico, apenas o controle do ângulo φ foi desenvolvido podendo ser igualmente aplicado ao controle de θ apenas ajustando seus os valores.

Já para o controle do ângulo de Euler ψ , por possuir uma dinâmica mais lenta e ser menos relevante do que os demais ângulos no que diz respeito à estabilidade de voo, uma técnica de controle menos sofisticada foi utilizada e é apresentada na subseção 5.3. Ele foi apresentado antes do controlador *Backstepping* porque o mesmo faz uso de todos os ângulos em seu projeto.

Todos os controladores aqui apresentados são adaptativos, onde seus ganhos variam em função da tensão da bateria. A forma com a qual estes ganhos variam em cada controlador é discutida nas subseções seguintes.

5.1 PID

Segundo Ogata (2010), a utilidade dos controles PID está na sua aplicabilidade geral à maioria dos sistemas de controle. Quando o modelo matemático da planta não é conhecido e, portanto, métodos de projeto analítico não podem ser utilizados, controles PID se mostram bastante interessantes.

Na área dos sistemas de controle de processos, é sabido que os controladores PID provaram sua utilidade provendo resultados satisfatórios, embora eles possam não proporcionar um controle ótimo. Segundo Elbasueny et al. (2019), a implementação deste tipo de controlador é simples e aplicável à maioria dos sistemas.

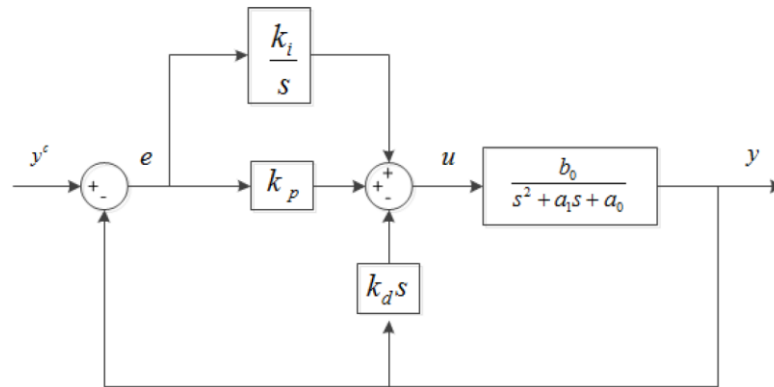
A estrutura padrão de um controlador PID é composta por três termos cuja função de transferência é dada por:

$$C(s) = (k_p + \frac{k_i}{s} + k_d s) \quad (5.1)$$

onde k_p , k_i e k_d são, respectivamente, o ganho proporcional, integral e derivativo do controlador. Segundo Karahan e Kasnakoglu (2019), o termo k_p , que fornece uma ação proporcional ao sinal de entrada do controlador, diminui o tempo de subida, mas não remove erro de regime permanente. O termo k_i , que podem eliminar erros em regime permanente, pode causar uma resposta transitória ruim. Por fim, o termo k_d reduz as oscilações transitórias e pode melhorar a estabilidade e diminuir o sobressinal.

Segundo Alves (2012), um controlador PID aplicado a um sistema de segunda ordem pode ser representado pelo diagrama de blocos da Figura 5.1, onde y^c é o valor de referência, e é o erro do sistema que se deseja anular, u é a saída do controlador e y é a saída do sistema. Esta topologia não é a topologia padrão.

Figura 5.1 – Controlador PID aplicado a um sistema de segunda ordem.



Fonte: Alves(2012).

Para o projeto do controlador PID, o modelo simplificado para o controle do ângulo φ apresentado na Equação (4.43) foi utilizado. Porém, ele representa o comportamento do sistema no tempo e, para o projeto dos controladores, o domínio da frequência é utilizado. Desta forma, a Equação (4.43) pode ser representada no domínio da frequência através da transformação de Laplace por:

$$\varphi(s) = \frac{1}{J_{xx} s^2} \tau_\varphi(s) \quad (5.2)$$

Neste caso, a entrada do sistema é o torque gerado pelos rotores 2 e 4 e a saída do sistema o ângulo φ gerado por este torque. Unindo as Equações (4.25) e (4.29), é possível chegar à Equação (5.3) que representa o torque em φ em função dos comandos de largura de pulso, dada por:

$$\tau_\varphi = l(K_F\delta_2 - K_F\delta_4) = 2lK_F\Delta\delta_\varphi \quad (5.3)$$

onde $\Delta\delta_\varphi = \frac{\delta_2 - \delta_4}{2}$ é a variação do comando que será calculada pelo controlador e será somada ao motor 2 e subtraída do motor 4.

Desta forma, a equação (5.2) pode ser reescrita como:

$$\varphi(s) = \frac{b_x}{s^2} \Delta\delta_\varphi(s) \quad (5.4)$$

onde $b_x = \frac{2l}{J_{xx}} k_F$.

Portanto, o controlador PID aqui projetado tem como função calcular a variação do comando PWM necessário para que o ângulo φ desejado seja alcançado. No caso do diagrama de blocos apresentado na Figura 5.1, y^c representa o ângulo φ desejado, e a diferença entre o ângulo desejado e o ângulo atual, u a variação de comando na largura de pulso ($\Delta\delta_\varphi$) e y o ângulo φ corrente.

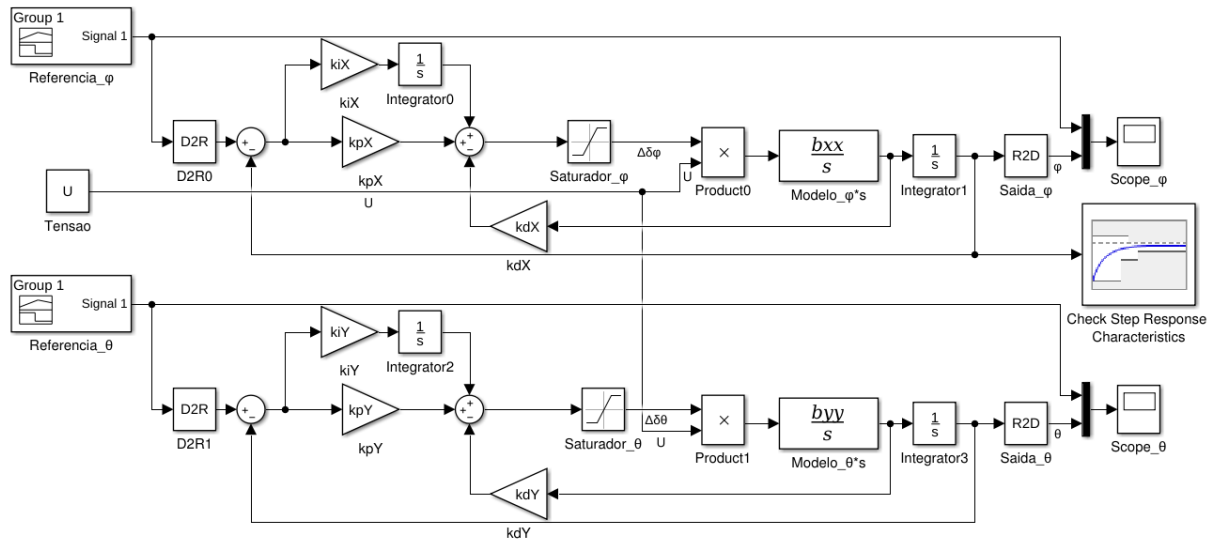
Antes de iniciar o ajuste dos ganhos do controlador, é preciso encontrar o valor de b_x . Substituindo os valores:

$$b_x = \frac{2l}{J_{xx}} k_F = \frac{2 \cdot 0,22}{0,013577} \cdot 0,641 \cdot 10^{-3} \cdot U = 20,77 \cdot 10^{-3} U$$

Desta forma, é possível notar que a planta varia em função da tensão de entrada. Para o ajuste dos ganhos do controlador, um bloco de otimização de parâmetros do *Simulink* (*Signal Constraint*) foi ligado à saída do sistema, conforme mostrado na Figura 5.2. Os requisitos impostos ao bloco otimizador foram: tempo de subida (critério 90%) igual a 0,7s, tempo para acomodação a 1% da referência igual 1,2s e sobressinal máximo de 8%. Estes valores foram escolhidos a fim de resultar no menor tempo para acomodação do sinal até o atingir o valor de referência. Inicialmente foi definido $U = 12,6V$.

O bloco saturador mostrado na Figura 5.2 foi introduzido no sistema para que a saída do controlador não ultrapasse os limites do sistema real. Seu valor foi ajustado para produzir uma saída entre -50 e +50, o que representa $-50\mu s \leq \Delta\delta_\varphi \leq 50\mu s$. O mesmo vale para o saturador de $\Delta\delta_\theta$.

Figura 5.2 – Controlador PID utilizado para o ajuste dos ganhos.



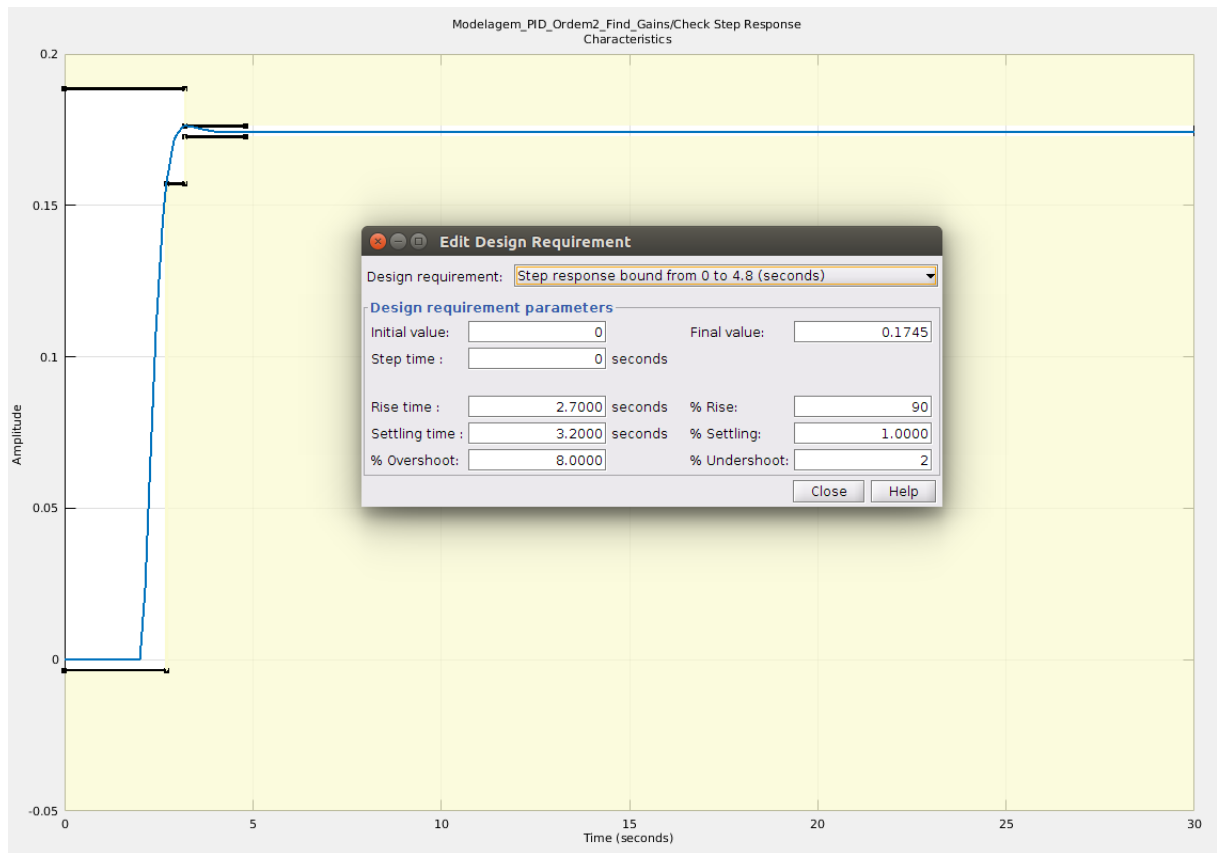
Fonte: Elaboração do próprio autor.

Para a realização da otimização, o sinal de referência de φ foi configurado como um degrau de amplitude 10 graus e os demais sinais de referência foram configurados com amplitude de zero. A curva gerada na ferramenta após as configurações dos ajustes mencionados anteriormente pode ser vista na Figura 5.3. Após a realização dos ajustes com este nível de tensão, foram realizados novos ajustes variando U de 12,6V a 10,6V a passos de 0,5V a fim de analisar os valores gerados pelo otimizador. Os valores retornados podem ser vistos nas curvas presentes na Figura 5.4. O mesmo procedimento foi realizado para o controle de θ .

Analisando os pontos presentes na Figura 5.4, é possível notar que as funções dos ganhos em relação à tensão de entrada se aproximam de retas e que os ganhos são inversamente proporcionais ao valor da tensão. Para determinar as funções que relacionam a tensão de entrada com os ganhos, uma aproximação linear de primeira ordem foi realizada com uso da ferramenta MATLAB. O script utilizado pode ser visto no Código 5.1.

Assim sendo, o controlador PID desenvolvido se trata de um controlador PID adaptativo onde os valores de seus ganhos variam em função do nível de tensão de entrada. O diagrama de blocos completo do controlador PID aplicado ao modelo simplificado pode ser visto na Figura 5.5. O conteúdo do interior do bloco do controlador PID pode ser visto na Figura 5.6.

Figura 5.3 – Ferramenta *Signal Constraint* aplicada ao ajuste dos ganhos do controlador PID aplicado ao controle de φ .



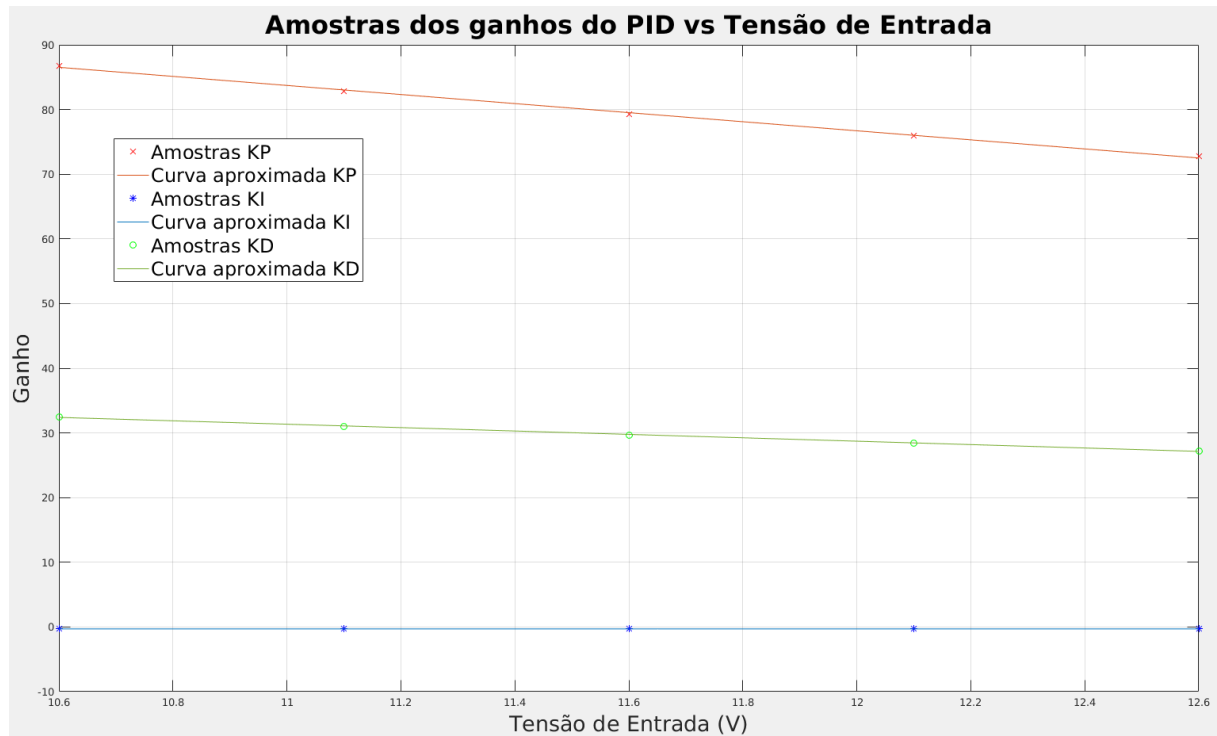
Fonte: Elaboração do próprio autor.

Código 5.1: Código utilizado para determinação das funções das curvas dos ganhos do controlador PID aplicado ao controle de φ .

```
u = [10.6 11.1 11.6 12.1 12.6];
kp = [86.8110, 82.8554, 79.2398, 75.9229, 72.7552];;
ki = [-0.3037, -0.3029, -0.3021, -0.3015, -0.3008];
kd = [32.5072, 31.0240, 29.6683, 28.4247, 27.2381];

polynomial_kp = polyfit(u, kp, 1);
polynomial_ki = polyfit(u, ki, 1);
polynomial_kd = polyfit(u, kd, 1);
kp_1 = polynomial_kp(1);
kp_0 = polynomial_kp(2);
ki_1 = polynomial_ki(1);
ki_0 = polynomial_ki(2);
kd_1 = polynomial_kd(1);
kd_0 = polynomial_kd(2);
```

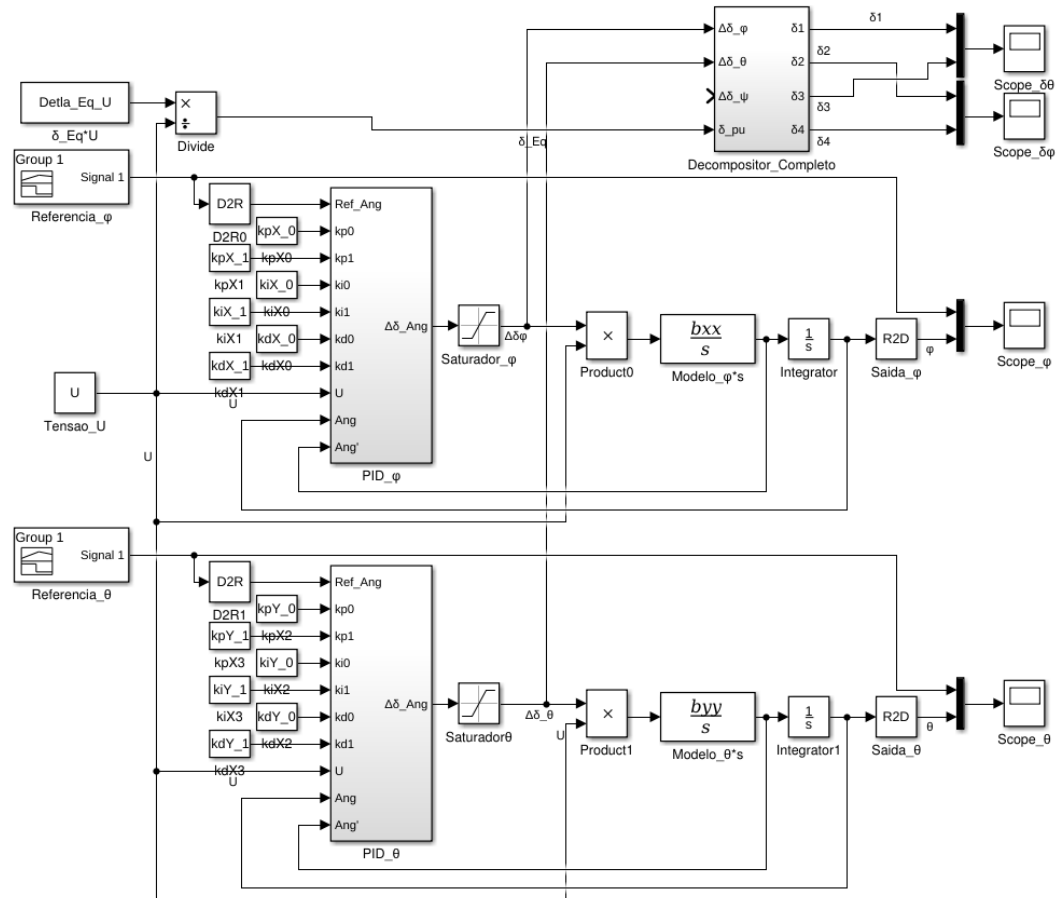
Figura 5.4 – Ganhos do controlador PID aplicado a φ vs tensão de entrada.



Fonte: Elaboração do próprio autor.

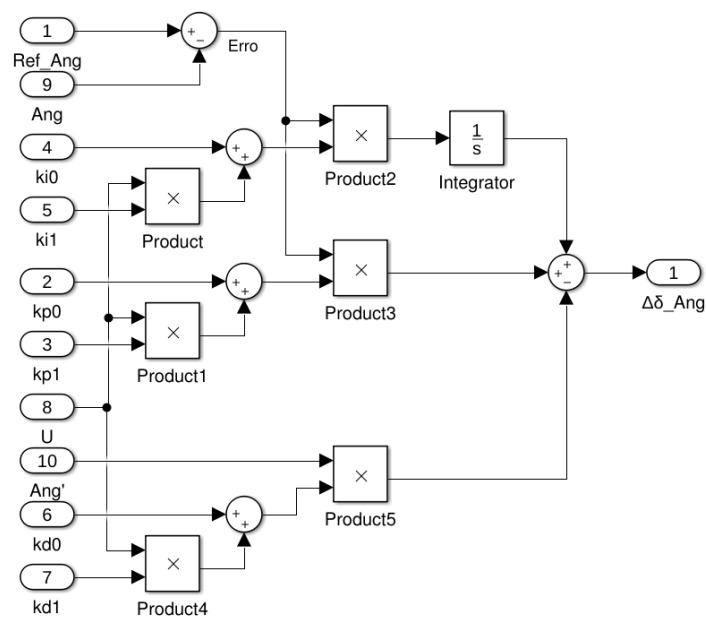
O bloco “Decompositor_Completo” tem como função decompor as saídas dos controladores para os comandos aplicados aos motores individualmente. Sua entrada δ_{pu} representa o comando necessário para manter a altitude do quadricóptero. Como não foi desenvolvido um controlador de altitude neste trabalho, uma entrada em malha aberta foi inserida para manter a altitude do quadricóptero quando o mesmo se encontra em repouso em um ponto fixo no espaço (sem movimentações laterais). O interior deste bloco pode ser visto na Figura 5.7.

Figura 5.5 – Controlador PID aplicado ao controle de φ e θ (modelo simplificado).



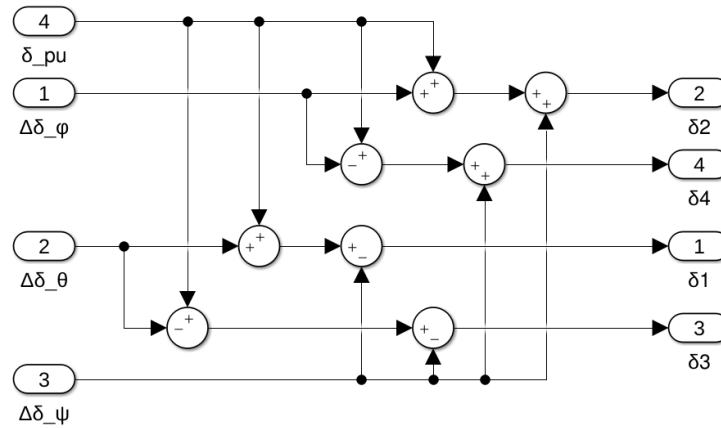
Fonte: Elaboração do próprio autor.

Figura 5.6 – Interior do bloco controlador PID.



Fonte: Elaboração do próprio autor.

Figura 5.7 – Interior do bloco do Decompositor Completo.



Fonte: Elaboração do próprio autor.

5.2 LQR

Um sistema linear invariante no tempo pode ser descrito na forma de variáveis de estado, segundo Okyere et al. (2018), por:

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

onde $x \in \mathbf{R}^n$ e $u \in \mathbf{R}^m$, n representa o número de estados e m o número de entradas.

Assumindo que todos os estados são observáveis, é possível encontrar um controle de realimentação de estados tal que

$$u = -Kx$$

Desta forma, o sistema de malha fechada que utiliza este controle é dado por:

$$\dot{x} = (A - BK)x = A_c x \quad (5.5)$$

onde A_c é a matriz da planta em malha fechada.

O problema do LQR se consiste em encontrar uma lei de controle $u(t)$ que minimize o índice de desempenho quadrático dado por $J = f(x, u)$ tal que

$$J = \frac{1}{2} \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (5.6)$$

onde as matrizes Q e R representam as matrizes de ponderação.

Substituindo a Equação (5.5) em (5.6), encontra-se

$$J = \frac{1}{2} \int_0^{\infty} (x^T (Q + K^T R K) x) dt \quad (5.7)$$

Para otimizar o sistema, é necessário encontrar K que minimize o índice de desempenho J .

Existem vários métodos numéricos na literatura para encontrar K . No caso deste projeto, a função presente na ferramenta MATLAB, $lqr(A,B,Q,R)$, foi utilizada. Para isso, deve-se determinar os valores das matrizes A , B , Q , R para serem utilizadas na função para que K seja retornado e inserido na malha de controle.

Assim como foi utilizado no projeto do controlador PID, o sistema a ser controlado pelo controlador LQR também é o sistema definido pela Equação (4.43) que é representado no domínio da frequência em Laplace pela Equação (5.4). A entrada do sistema também é o torque gerado pelos rotores 2 e 4 e a saída também é o ângulo gerado por este torque.

Reescrevendo a Equação (5.4) na forma de espaço de estados, tem-se:

$$f(x, u) = \begin{bmatrix} \dot{\varphi} \\ \ddot{\varphi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 \\ b_x \end{bmatrix} \Delta\delta_\varphi \quad (5.8)$$

Desta forma, as matrizes A , B são definidas por $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ e $B = \begin{bmatrix} 0 \\ b_x \end{bmatrix}$.

Para analisar a controlabilidade do sistema, é necessário encontrar o posto da matriz M_c , dada por:

$$M_c = [A \ B] = \begin{bmatrix} 0 & b_x \\ b_x & 0 \end{bmatrix}$$

Como o posto da matriz M_c é igual a dois, o mesmo valor de n , o sistema é controlável.

Deve-se agora selecionar as matrizes Q e R positivas definidas para ser possível determinar K . Uma das maneiras é através do uso da regra de Bryson, descrito por Oral, Çetin e Uyar (2010), ou seja:

$$Q_{ii} = \frac{1}{x_{i_max}^2}, i \in \{1, \dots, n\} \quad (5.9)$$

$$R_{jj} = \frac{1}{u_{j_max}^2}, j \in \{1, \dots, m\} \quad (5.10)$$

onde as matrizes Q e R são diagonais.

Para o caso do quadrirrotor, foi definido que o erro máximo de ângulo permitido é de $\frac{\pi}{6}$ rad (30°) e que a velocidade angular máxima é de $\frac{\pi}{2}$ rad/s (90°/s).

Desta forma, $x_{11_max} = \frac{\pi}{6} - (-\frac{\pi}{6}) = \frac{\pi}{3}$ rad e $x_{22_max} = \frac{\pi}{2} - (-\frac{\pi}{2}) = \pi$ rad/s.

Já para a determinação do valor de R , é necessário saber a variação máxima na entrada do controlador. Esta variação se trata da máxima diferença na largura de pulso que pode ser aplicada ao ESC. Assim como foi feito para o controlador PID, esta variação máxima foi definida em $50 - (-50) = 100 \mu s$. Logo, $u_{11_max} = 100$.

$$\text{Desta forma, } Q = \begin{bmatrix} \frac{1}{(\frac{\pi}{3})^2} & 0 \\ 0 & \frac{1}{\pi^2} \end{bmatrix} \text{ e } R = \begin{bmatrix} \frac{1}{100^2} \end{bmatrix}.$$

Agora que estão definidas as matrizes A , B , Q , R , é possível determinar K através do método `lqr(A,B,Q,R)` do MATLAB que retorna um vetor na forma $K = [K_1, K_2]$. Logo, a entrada de controle u é dada por:

$$u = [K_1, K_2] \begin{bmatrix} \varphi \\ \dot{\varphi} \end{bmatrix} \quad (5.11)$$

Para atingir o equilíbrio do sistema em um estado diferente de zero, o estado desejado é acrescentado ao estado do sistema como

$$x = \begin{bmatrix} x_1 - x_1^d \\ x_2 - x_2^d \end{bmatrix} \quad (5.12)$$

Visto que o valor desejado para a posição angular é φ^d e para a velocidade angular é zero, a entrada de controle u pode ser dada por:

$$u = [K_1, K_2] \begin{bmatrix} \varphi - \varphi^d \\ \dot{\varphi} \end{bmatrix} \quad (5.13)$$

Assim como abordado no projeto do controlador PID, os ajustes dos ganhos do controlador dependem do valor da tensão de entrada U através do termo b_x presente na matriz B . Novamente, foram efetuadas otimizações com alguns valores de entradas e os resultados foram plotados em uma curva a fim de ser possível analisar seus comportamentos em função da tensão de entrada. Foi constatado que o ganho K_1 não é interferido pelo valor de tensão.

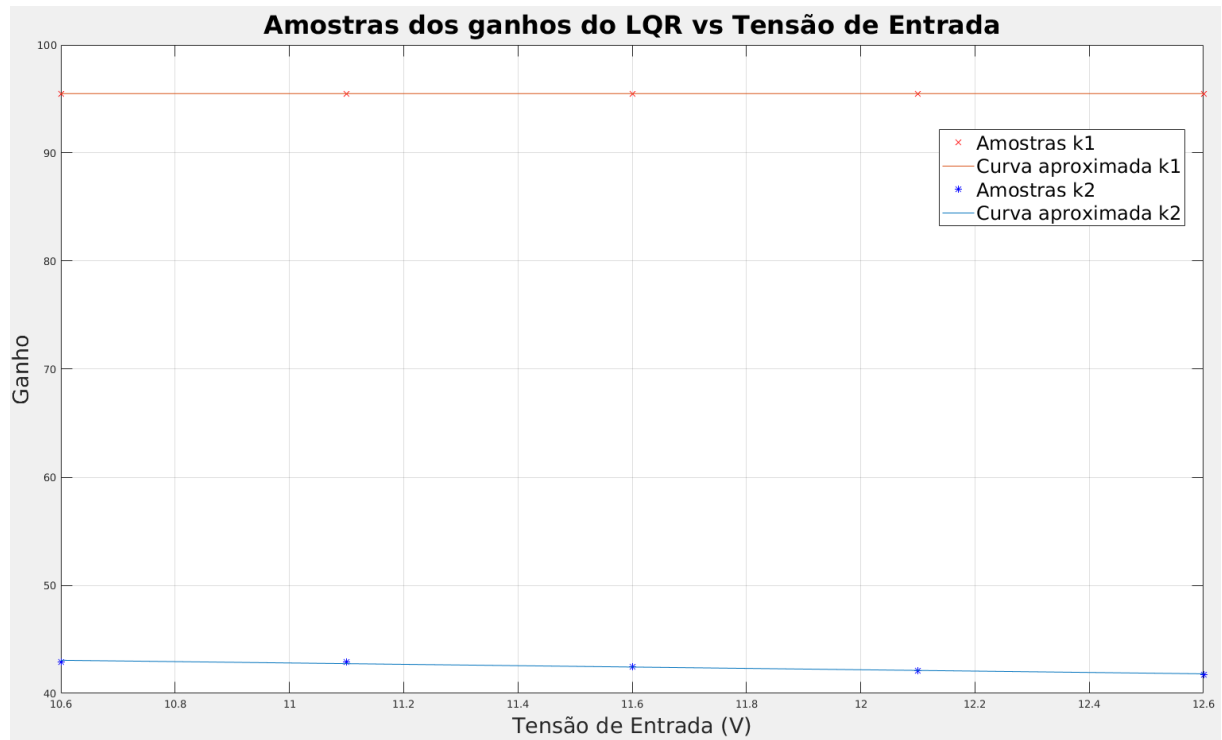
O mesmo não acontece pra K_2 e seu comportamento pode ser visto na Figura 5.8. Para determinar a função que relaciona a tensão de entrada com este ganho, uma aproximação linear de primeira ordem foi realizada com uso da ferramenta MATLAB. Para determinar os coeficientes da função que representa tal curva, o Código 5.2 foi utilizado na ferramenta MATLAB.

Logo, o controlador LQR desenvolvido se trata de um controlador adaptativo, onde os valores de seus ganhos variam em função do nível de tensão de entrada.

Código 5.2: Código utilizado para determinação da função da curva do ganho K_2 do controlador LQR aplicado ao controle de φ .

```
u_s = [10.6 11.1 11.6 12.1 12.6];
k2_s = [42.9122, 42.9122, 42.4942, 42.1071, 41.7475];
polynomial = polyfit(u_s, k2_s, 1);
k2_1 = polynomial(1);
k2_0 = polynomial(2);
```

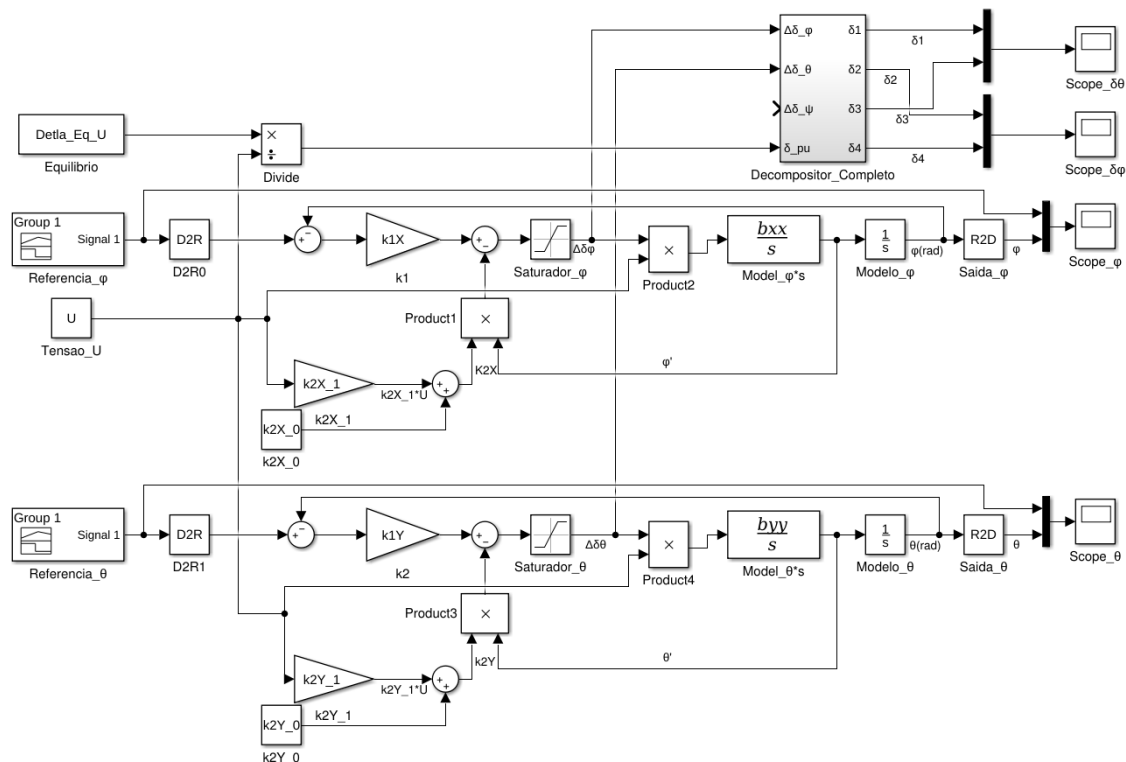
Figura 5.8 – Ganhos do controlador LQR aplicado a φ vs tensão de entrada.



Fonte: Elaboração do próprio autor.

O diagrama de blocos do controlador projetado pode ser visto na Figura 5.9.

Figura 5.9 – Controlador LQR aplicado ao controle de φ e θ (modelo simplificado).



Fonte: Elaboração do próprio autor.

5.3 Controle de ψ

Os ângulos de Euler φ e θ , conforme mencionado anteriormente, são os principais responsáveis pela estabilização de voo do quadricóptero, sendo o ângulo ψ responsável apenas pela direção de voo do mesmo. Entretanto, o controle do ângulo ψ é necessário visto que sem ele, o quadricóptero poderia rotacionar de forma desordenada em torno deste ângulo, o que tornaria o controle da estrutura inviável.

Para o projeto do controlador, é mais fácil que o controle seja realizado sobre a velocidade angular de ψ , e não sobre ângulo ψ em si, visto que a medida da posição angular sobre este eixo não é tão precisa e teria que ser feita via uso do magnetômetro. Já a medida da velocidade angular pode ser realizada via o uso do sensor giroscópico, fornecendo assim uma maior precisão. Portanto, neste trabalho o controle de ψ é realizado sobre sua velocidade angular.

O modelo utilizado para o projeto do controlador da velocidade de guinada é o descrito pela Equação (4.43) que pode ser representado no domínio da frequência pela transformada de Laplace por:

$$\psi'(s) = \frac{1}{J_{zz}} \tau_{\psi}(s) \quad (5.14)$$

Neste caso, a entrada do sistema é o torque gerado pelos dois pares de rotores e a saída do sistema a velocidade angular ψ' gerada por estes torques. Unindo as Equações (4.27) e (4.30), é possível chegar à Equação (5.15) que representa o torque em ψ em função dos comandos de largura de pulso.

$$\tau_{\psi} = K_{\tau} \delta_2 + K_{\tau} \delta_4 - K_{\tau} \delta_1 - K_{\tau} \delta_3 = 4K_{\tau} \Delta\delta_{\psi} \quad (5.15)$$

onde $\Delta\delta_{\psi} = \frac{\delta_2 + \delta_4 - \delta_1 - \delta_3}{4}$ é a variação do comando que será calculada pelo controlador e será somada ao par de motores 2 e 4 e subtraída do par de motores 1 e 3.

Desta forma, a Equação (5.15) pode ser reescrita como:

$$\psi'(s) = \frac{b_z}{s} \Delta\delta_{\psi}(s) \quad (5.16)$$

onde $b_z = \frac{4K_{\tau}}{J_{zz}}$.

Como a dinâmica deste ângulo não é tão rápida e crucial para estabilidade de voo, um controlador menos sofisticado pode ser utilizado. Neste caso, um controlador proporcional foi utilizado, assim como no trabalho de Alves (2012).

Portanto, o controlador P projetado tem como função calcular a variação do comando PWM necessário para que a velocidade angular ψ' desejada seja alcançada. A equação de controle é dada por

$$u = k_p(\psi'^d - \psi') \quad (5.17)$$

onde $u = \Delta\delta_\psi(s)$.

Desta forma, a função de transferência do sistema em malha fechada é dada por:

$$\frac{\psi'}{\psi'^d} = \frac{b_z k_p}{s + b_z k_p} \quad (5.18)$$

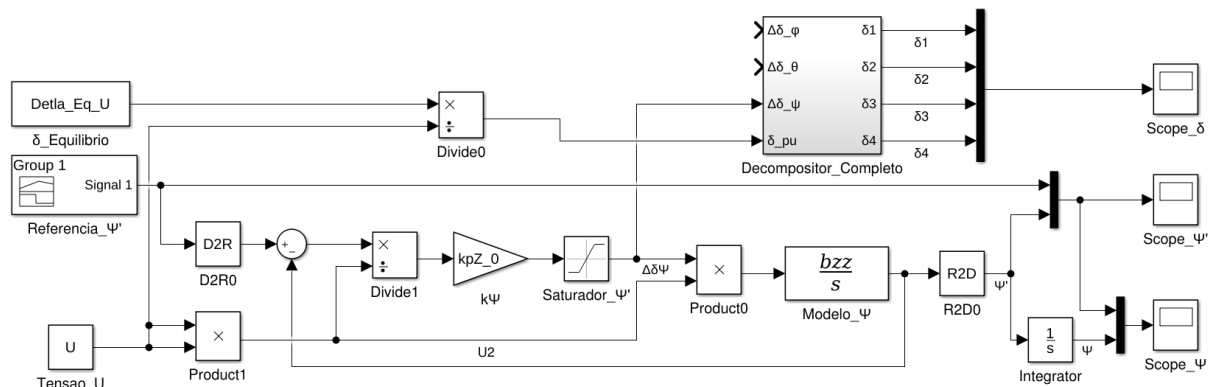
Portanto, para que o sistema seja estável, basta que k_p seja positivo. Porém, antes de determinar o seu valor, é necessário determinar o valor de b_z para que seja possível a realização de simulações para análise de eficiência do controlador.

Substituindo os valores:

$$b_z = \frac{4k_\tau}{J_{zz}} = \frac{4 \cdot 0,728 \cdot 10^{-6}}{0,024380} \cdot U^2 = 1,195 \cdot 10^{-4} \cdot U^2$$

Novamente, é possível notar que o sistema varia em função da tensão de entrada. Seguindo o mesmo método utilizado pra determinação da relação dos ganhos do controlador em função da tensão de entrada no projeto do controlador PID, foi possível determinar que a dependência trata-se de uma função de segunda ordem. O diagrama de blocos do controlador projetado utilizado para realização das simulações pode ser visto na Figura 5.10. A saída do controlador é o erro dividido pelo quadrado da tensão da bateria multiplicado pelo ganho k_{pz_0} . Desta forma, quanto menor o nível de tensão da bateria, maior será a saída do controlador.

Figura 5.10 – Controlador P aplicado ao controle de ψ' (modelo simplificado).



Fonte: Elaboração do próprio autor

O bloco saturador mostrado na Figura 5.10 foi introduzido no sistema para que a saída do controlador não ultrapasse os limites do sistema real. Seu valor foi ajustado para produzir uma saída entre -100 e +100, o que representa $-100\mu s \leq \Delta\delta_\psi \leq 100\mu s$.

5.4 Backstepping

Visto que a dinâmica do quadrirotor é não linear, um controle não linear pode se mostrar mais eficaz. A técnica de controle não linear adotada neste trabalho foi a *Backstepping* (Zhou e Wen, 2008). Para o projeto deste controlador, foi levado em consideração o modelo simplificado não linearizado descrito pelas Equações (4.36) e (4.40). Assumindo que os ângulos θ e φ são pequenos e substituindo a entrada do sistema para comando PWM ao invés de torque, o modelo pode ser descrito na forma de espaço de estados por:

$$\begin{bmatrix} \dot{\varphi} \\ \ddot{\varphi} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{J_{yy}-J_{zz}}{2J_x} \dot{\psi} & 0 & \frac{J_{yy}-J_{zz}}{2J_x} \dot{\theta} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{J_{zz}-J_{xx}}{2J_y} \dot{\psi} & 0 & 0 & 0 & \frac{J_{zz}-J_{xx}}{2J_y} \dot{\varphi} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{J_{xx}-J_{yy}}{2J_{zz}} \dot{\theta} & 0 & \frac{J_{xx}-J_{yy}}{2J_{zz}} \dot{\varphi} & 0 & 0 \end{bmatrix} \begin{bmatrix} \varphi \\ \dot{\varphi} \\ \theta \\ \dot{\theta} \\ \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_x & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_y & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & b_z \end{bmatrix} \begin{bmatrix} 0 \\ \Delta\delta_\varphi \\ 0 \\ \Delta\delta_\theta \\ 0 \\ \Delta\delta_\psi \end{bmatrix} \quad (5.19)$$

Assim como no controlador PID, o controlador aqui projetado tem como função calcular a variação da largura do comando PWM necessária para alcançar os ângulos φ e θ desejados. Os passos aqui adotados para a determinação do controlador Backstepping foram extraídos do trabalho de Alves (2012).

Para o primeiro passo, considerando apenas o ângulo φ , o erro do sistema é dado por:

$$z_1 = x_1^d - x_1 \quad (5.20)$$

A função de Lyapunov escolhida deve ser positiva definida e sua derivada no tempo negativa definida. Esta função é dada por

$$V_1(z_1) = \frac{1}{2} z_1^2 \quad (5.21)$$

e portanto

$$\dot{V}_1(z_1) = z_1 \dot{z}_1 = z_1 (\dot{x}_1^d - \dot{x}_1) \quad (5.22)$$

Para garantir a estabilidade de z_1 , uma entrada de controle virtual x_2 é introduzida no sistema, onde

$$\dot{x}_2 = \dot{x}_1^d + \alpha_1 z_1, \quad \alpha_1 > 0$$

que substituindo na equação (5.22) resulta em:

$$\dot{V}(z_1) = \alpha_1 z_1^2 \quad (5.23)$$

Aplicando mudança de variável, é possível definir z_2 como:

$$z_2 = x_2 - \dot{x}_1^d + \alpha_1 x_1 \quad (5.24)$$

Para o segundo passo, é considerado a função quadrática de Lyapunov dada por:

$$V_2(z_1, z_2) = \frac{1}{2}(z_1^2 + z_2^2) \quad (5.25)$$

$\dot{V}_2(z_1, z_2)$ pode ser descrita por:

$$\dot{V}_2(z_1, z_2) = z_1 \dot{z}_1 + z_2 \dot{z}_2 \quad (5.26)$$

$$\dot{V}(z_1, z_2) = z_2 \left(\frac{J_y - J_z}{J_x} x_4 x_6 + u_1 \right) - z_2 \left(\ddot{x}_1^d - \alpha_1 (z_2 + z_1 \alpha_1) \right) - z_1 z_2 - \alpha_1 z_1^2 \quad (5.27)$$

Considerando $\ddot{x}_{1,2,3}^d = 0$ e $\dot{V}(z_1, z_2) < 0$, tem-se:

$$u_1 = (z_1 - \frac{J_y - J_z}{J_x} x_4 x_6 - \alpha_1 (z_2 + z_1 \alpha_1) - \alpha_2 z_2) \quad (5.28)$$

Desta forma, pode-se reescrever $\dot{V}(z_1, z_2)$ como

$$\dot{V}(z_1, z_2) = -\alpha_1 z_1^2 - \alpha_2 z_2^2 \quad (5.29)$$

O termo α_2 , que é maior do que zero, é adicionado para garantir a estabilização de z_2 . Substituindo z_1 e z_2 na equação anterior que representa u_1 , tem-se:

$$u_1 = ((1 + \alpha_1 \alpha_2)(\dot{x}_1^d - x_1) - \frac{J_{yy} - J_{zz}}{J_{xx}} x_4 x_6 - (\alpha_1 + \alpha_2)x_2) \quad (5.30)$$

Definindo β_1 e β_2 como

$$\beta_1 = 1 + \alpha_1 \alpha_2 \quad (5.31)$$

$$\beta_2 = \alpha_1 + \alpha_2 \quad (5.32)$$

o objetivo é encontrar os valores de β_1 e β_2 que otimizem a resposta do sistema.

De forma similar, pode-se encontrar u_2 :

$$u_2 = ((1 + \alpha_3 \alpha_4)(\dot{x}_3^d - x_3) - \frac{J_{yy} - J_{zz}}{J_{xx}} x_2 x_6 - (\alpha_3 + \alpha_4)x_4) \quad (5.33)$$

onde

$$z_3 = x_3^d - x_3 \quad (5.34)$$

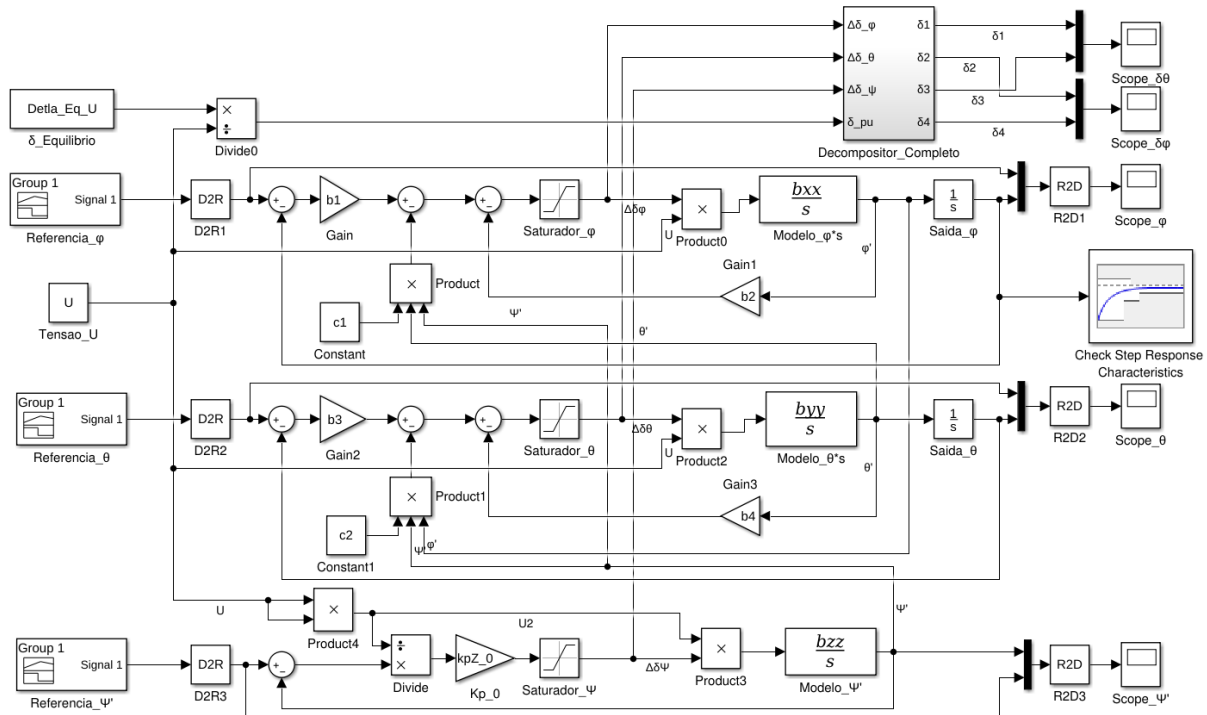
$$z_4 = x_4 - \dot{x}_3^d - \alpha_3 z_3 \quad (5.35)$$

$$\beta_3 = 1 + \alpha_3 \alpha_4 \quad (5.36)$$

$$\beta_4 = \alpha_3 + \alpha_4 \quad (5.37)$$

Para o ajuste dos ganhos do controlador, um bloco de otimização de parâmetros do *Simulink* (*Signal Constraint*) foi ligado à saída do sistema, conforme mostrado na Figura 5.11. Visto que a dinâmica de ψ é lenta, o controlador apresentado na seção anterior foi utilizado para o controle deste ângulo, ficando a cargo da técnica *Backstepping* apenas o controle de φ e θ . Os requisitos impostos ao bloco otimizador foram os mesmos impostos para o controlador PID.

Figura 5.11 – Controlador *Backstepping* utilizado no *Simulink* para ajuste de ganhos.

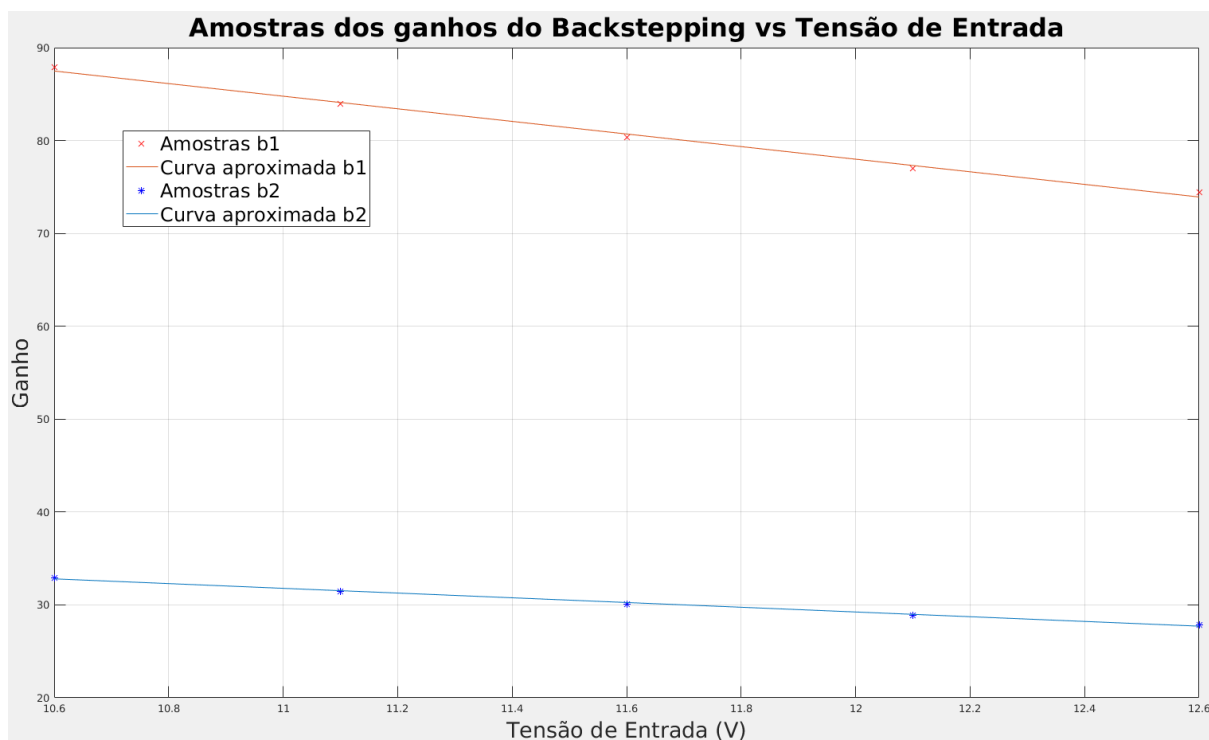


Fonte: Elaboração do próprio autor.

Para a realização da otimização, o mesmo procedimento utilizado no controlador PID foi utilizado. Os valores retornados podem ser vistos nas curvas presentes na Figura 5.12.

Analisando os pontos presentes na Figura 5.12, é possível notar que as funções dos ganhos em relação à tensão de entrada se aproximam de retas. Para determinar as funções que relacionam a tensão de entrada com os ganhos, uma aproximação linear de primeira ordem foi realizada com uso da ferramenta MATLAB. O script utilizado pode ser visto no Código 5.3.

Figura 5.12– Ganhos do controlador *Backstepping* aplicado a φ vs tensão de entrada.



Fonte: Elaboração do próprio autor.

Código 5.3: Código utilizado para determinação da função das curvas dos ganhos do controlador Backstepping aplicado ao controle de φ .

```
Jx = 0.013577;
Jy = 0.013993;
Jz = 0.024380;

c1 = (Jy-Jz)/Jx;
c2 = (Jx-Jz)/Jy;

U = 12.6;
Kp_0 = 60*12.6;

u = [10.6 11.1 11.6 12.1 12.6];

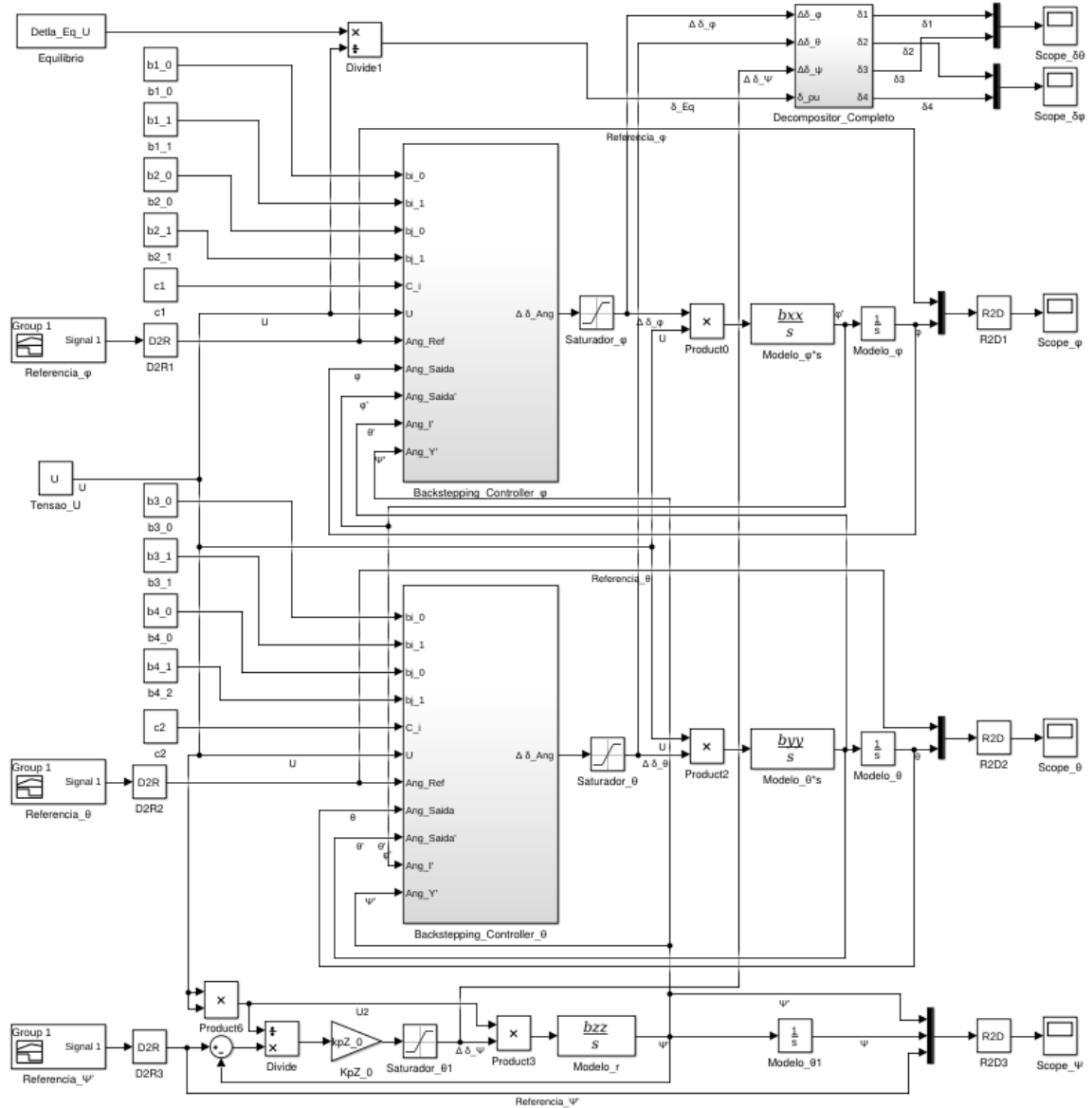
b1_array = [28.30 28.52 28.76 28.99 29.21];
b2_array = [18.45 18.35 18.24 18.13 18.02];

polynomial_b1 = polyfit(u, b1_array, 1);
polynomial_b2 = polyfit(u, b2_array, 1);

b1_1 = polynomial_b1(1);
b1_0 = polynomial_b1(2);
b2_1 = polynomial_b2(1);
b2_0 = polynomial_b2(2);
```

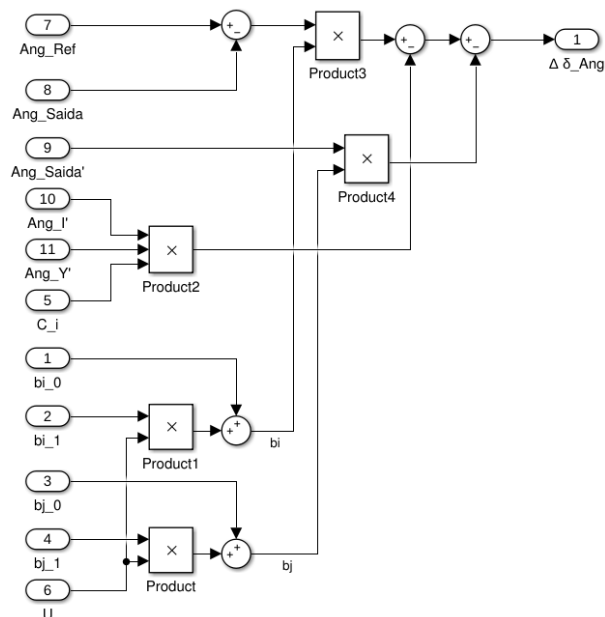
Assim sendo, o controlador *Backstepping* desenvolvido se trata de um controlador *Backstepping* adaptativo onde os valores de seus ganhos variam em função do nível de tensão de entrada. O diagrama de blocos do controlador projetado utilizado para realização das simulações pode ser visto na Figura 5.13, que engloba as Equações (5.31) e (5.33). O conteúdo do bloco do controlador *Backstepping* projetado pode ser visto na Figura 5.14.

Figura 5.13 - *Backstepping* aplicado ao controle de φ e θ (modelo simplificado).



Fonte: Elaboração do próprio autor.

Figura 5.14 - Interior do bloco do controlador *Backstepping*.



Fonte: Elaboração do próprio autor.

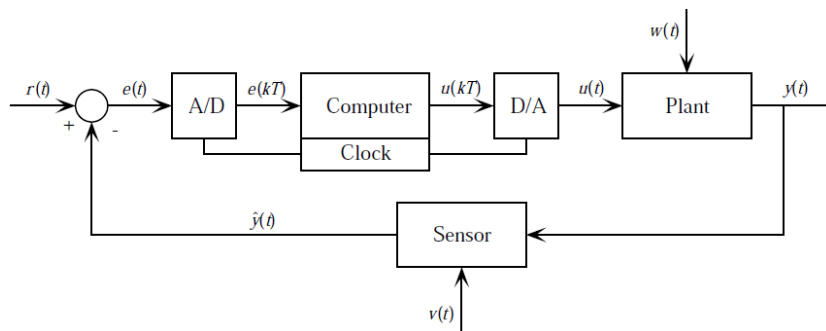
A partir dos controladores contínuos projetados nesta seção, é possível encontrar seus correspondentes no domínio discreto. Os controladores no domínio discreto são necessários, visto que o controle do quadricóptero será executado sobre um computador digital. A discretização destes controladores é apresentada na próxima seção.

6 Controle Discreto

Os projetos dos controladores discretos têm como objetivo tornar possível a implementação dos mesmos em um computador digital. Visto que o modelo real utiliza um computador digital, o controlador que será embarcado no mesmo deve ser discreto.

Segundo Starr (2006), um sistema de controle digital pode ser ilustrado pela Figura 6.1, onde um processador digital toma a forma do bloco de controle responsável por realizar a compensação e atuar sobre a planta a ser controlada e T representa o tempo de amostragem do sistema.

Figura 6.1 – Sistema de controle digital.



Fonte: Starr (2006).

Para realizar a conversão dos controladores contínuos apresentados na seção 5, a transformada Z foi utilizada. Para a aproximação da derivada, a técnica conhecida como “*Backward Difference*” foi utilizada. Para a utilização desta técnica, basta realizar a seguinte substituição de variáveis:

$$s = \frac{(1-z^{-1})}{T} \quad (6.1)$$

Já para a aproximação da integral, a técnica conhecida como “Transformação Bilinear” ou “Transformação TUSTIN” foi utilizada. Para a utilização desta técnica, basta realizar a seguinte substituição de variáveis:

$$s = \frac{2}{T} \frac{(1-z^{-1})}{(1+z^{-1})} \quad (6.2)$$

As discretizações dos controladores contínuos apresentadas na seção 5 são apresentadas nas subseções seguintes. Após encontrar as respectivas equações de

diferenças de cada controlador, os mesmos foram convertidos em código e embarcados no quadrrirrotor.

Também será apresentada na subseção 6.4 a estrutura de testes giroscópica desenvolvida para a validação da eficiência dos controladores discretos aplicados ao quadrrirrotor.

A taxa de amostragem T utilizada pelos controladores discretos neste projeto foi $T = 0,8s$. Esta taxa foi encontrada experimentalmente e representa a menor taxa de resposta do conjunto ESC, rotor e hélice utilizado.

6.1 PID

Segundo o diagrama de blocos apresentado na Figura 5.1, o controlador PID contínuo aplicado ao controle de φ é descrito por:

$$U(s) = k_p E(s) + \frac{k_i}{s} E(s) - k_d s \varphi(s) \quad (6.3)$$

Aplicando as substituições de variáveis apresentadas nas equações (6.1) e (6.2) a esta função de transferência, é possível concluir que o controlador PID discreto pode ser descrito pela seguinte equação:

$$U(z) = k_p E(z) + k_i \frac{T}{2} \frac{(1+z^{-1})}{(1-z^{-1})} E(z) - k_d \frac{(1-z^{-1})}{T} \Phi(z) \quad (6.4)$$

$$U(z) = k_p E(z) + k_i h_1 \frac{(1+z^{-1})}{(1-z^{-1})} E(z) + k_d h_2 (1-z^{-1}) \Phi(z) \quad (6.5)$$

onde $h_1 = \frac{T}{2}$ e $h_2 = \frac{1}{T}$.

Desenvolvendo a Equação (6.5), chega-se em:

$$U(z) = \frac{k_p(1-z^{-1})E(z) + k_i h_1 (1+z^{-1})E(z) + k_d h_2 (1-z^{-1})^2 \Phi(z)}{1-z^{-1}} \quad (6.6)$$

O que resulta em:

$$U(z) = \frac{(k_p + k_i h_1)E(z) + (-k_p + k_i h_1)E(z)z^{-1} - k_d h_2 \Phi(z) + 2k_d h_2 \Phi(z)z^{-1} - k_d h_2 \Phi(z)z^{-2}}{1-z^{-1}} \quad (6.7)$$

Que pode ser simplificado a:

$$U(z) = \frac{b_0 E(z) + b_1 E(z)z^{-1} - b_2 \Phi(z) + 2b_2 \Phi(z)z^{-1} - b_2 \Phi(z)z^{-2}}{1-z^{-1}} \quad (6.8)$$

onde $b_0 = k_p + k_i h_1$, $b_1 = -k_p + k_i h_1$ e $b_2 = k_d h_2$.

Desenvolvendo a Equação (6.8):

$$U(z) = b_0 E(z) + b_1 E(z)z^{-1} - b_2 \Phi(z) + 2b_2 \Phi(z)z^{-1} - b_2 \Phi(z)z^{-2} + U(z)z^{-1} \quad (6.9)$$

Logo, a equação de diferenças que descreve um controlador PID discreto é dada por:

$$u[n] = b_0 e[n] + b_1 e[n-1] - b_2 \varphi[n] + 2b_2 \varphi[n-1] - b_2 \varphi[n-2] + u[n-1] \quad (6.10)$$

Porém, conforme visto na seção 5.1, os ganhos k_p , k_i e k_d variam em função da tensão da bateria. Desta forma, a cada iteração de processamento, o nível de bateria deve ser lido e os termos b_0 , b_1 e b_2 recalculados.

6.2 LQR

Segundo o diagrama de blocos apresentado na Figura 5.9, o controlador LQR contínuo aplicado ao controle de φ é descrito por:

$$U(s) = k_1 E(s) - k_2 \varphi(s)s \quad (6.11)$$

Aplicando as substituições de variáveis apresentadas nas equações (6.1) a esta equação, é possível determinar que a lei de controle do controlador LQR discreto aplicado ao controle de φ pode ser descrito pela equação

$$U(z) = k_1 E(z) - k_2 \Phi(z) \frac{(1-z^{-1})}{T} \quad (6.12)$$

que resulta em:

$$U(z) = k_1 E(z) - \frac{k_2}{T} \Phi(z) + \frac{k_2}{T} \Phi(z) z^{-1} \quad (6.13)$$

Logo, a equação de diferenças que descreve o controlador LQR discreto aplicado ao controle de φ é dada por:

$$u[n] = k_1 e[n] - \frac{k_2}{T} (\varphi[n] - \varphi[n-1]) \quad (6.14)$$

Conforme visto na seção 5.2, o ganho k_2 varia em função da tensão da bateria. Desta forma, a cada iteração de processamento, o nível de bateria deve ser lido e o termo k_2 recalculado.

6.3 Controle de ψ

Foi determinado na seção 5.3 que o controle do eixo de guinada se daria em torno de sua velocidade angular ψ' e que um controlador proporcional seria o suficiente para executar tal controle.

No caso do controlador proporcional, a transformação para equação de diferenças é direta. Logo, a equação de diferenças que descreve o controlador P discreto aplicado ao controle de ψ' é dada por:

$$u[n] = k_p e[n] \quad (6.15)$$

Conforme visto na seção 5.3, o ganho varia k_p em função da tensão da bateria. Desta forma, a cada iteração de processamento, o nível de bateria deve ser lido e o termo k_p recalculado.

6.4 Backstepping

Segundo o diagrama apresentado na Figura 5.13 e na Equação (5.30), o controlador *Bacskstepping* contínuo aplicado ao controle de φ é descrito por:

$$U(s) = b_1 E(s) - b_2 \varphi(s)s - c_1 s \theta(s) \psi'(s) \quad (6.16)$$

Aplicando as substituições de variáveis apresentadas nas equações (6.1) a esta equação, é possível determinar que a lei de controle do controlador *Bacskstepping* discreto aplicado ao controle de φ pode ser descrito pela equação

$$U(z) = b_1 E(z) - b_2 \Phi(z) \frac{(1-z^{-1})}{T} - c_1 \Theta(z) \frac{(1-z^{-1})}{T} \Psi'(z) \quad (6.17)$$

que resulta em:

$$U(z) = b_1 E(z) - \frac{b_2}{T} \Phi(z) + \frac{b_2}{T} \Phi(z) z^{-1} - \frac{c_1}{T} \Theta(z) \Psi'(z) + \frac{c_1}{T} \Theta(z) z^{-1} \Psi'(z) \quad (6.18)$$

Logo, a equação de diferenças que descreve o controlador Backstepping discreto aplicado ao controle de φ é dada por:

$$u[n] = b_1 e[n] - \frac{b_2}{T} (\varphi[n] - \varphi[n-1]) - \frac{c_1}{T} (\theta[n] \psi' - \theta[n-1] \psi'[n]) \quad (6.19)$$

Conforme visto na seção 5.4, os ganhos b_1 e b_2 variam em função da tensão da bateria. Desta forma, a cada iteração de processamento, o nível de bateria deve ser lido e os termos b_1 , b_2 e c_1 recalculados.

A partir das equações de diferenças encontradas nas discretizações dos controladores, é possível descrever as equações em software. Tomando como exemplo a equação de diferenças apresentada na Equação (6.19): $u[n]$ representa a variação de largura de pulso que será aplicada entre os rotores 2 e 4 no n -ésimo intervalo, $e[n]$ representa o erro no n -ésimo intervalo, que é a diferença entre o ângulo φ desejado e o ângulo φ na n -ésima amostra, $\varphi[n]$ representa o ângulo de rolagem no n -ésimo intervalo, $\varphi[n-1]$ o ângulo de rolagem medido no ciclo anterior,

$\theta[n]$ o ângulo de arfagem n -ésimo intervalo, $\theta[n - 1]$ o ângulo de arfagem medido no ciclo anterior e, por fim, $\psi'[n]$ representa a taxa de guinada no n -ésimo intervalo. Para implementação desta equação de diferenças em software, foi necessário criar uma função responsável pela leitura e armazenamento destas variáveis, com os valores correntes e lidos no ciclo anterior. Uma *thread* é responsável por fazer a chamada desta função a cada 0,8 segundo, que foi definido como taxa de amostragem, e é responsável por realizar o cálculo. A partir do resultado do cálculo, a diferença de largura de pulso é então aplicada aos dois ESCs correspondentes ao par de rotor a serem controlados, e no próximo ciclo, a função é executada novamente.

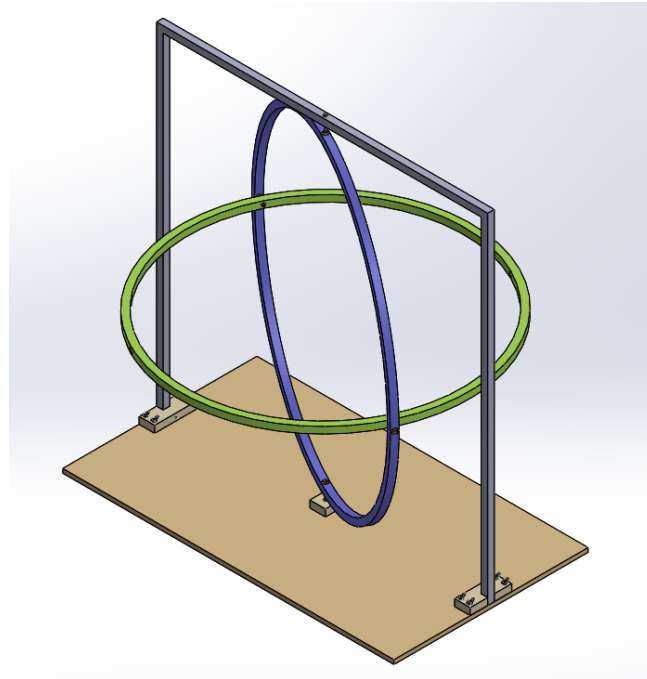
6.5 Estrutura de Testes Giroscópica

A estrutura de testes giroscópica tem como função possibilitar o teste de estabilização de voo sem a necessidade de colocar o quadricóptero em um voo real. Ela é formada por dois aros, uma trave e uma base conforme pode ser visto na Figura 6.2. Todos os movimentos são rolamentados a fim de diminuir o atrito e braços de alumínio foram utilizados para realizar o acoplamento entre o quadricóptero e o aro central. A Figura 6.3 mostra o CAD 3D completo da estrutura com o quadricóptero acoplado e a Figura 6.4 mostra a estrutura real com o quadricóptero.

A estrutura desenvolvida disponibiliza os três graus de liberdade rotativos para o quadricóptero: rotações nos eixos de rolagem, arfagem e guinada. Além de possibilitar a análise dos ângulos de rotação sem interferências, interferências podem se incluídas no sistema para analisar o comportamento dos controladores. Uma possível interferência é a inserção de peso em somente uma das extremidades do quadricóptero. Este tipo de interferência simularia, por exemplo, o vento em uma direção. Rajadas de vento podem ser inseridas também através de um soprador.

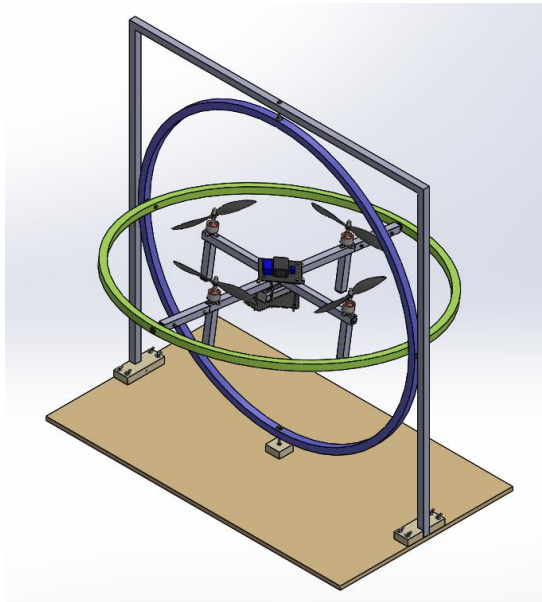
Uma consequência negativa que esta estrutura traz é a inserção de inércia nos movimentos, principalmente nos eixos de rolagem e guinada devido a massa dos aros. Portanto, a medida que mais se aproxima de um voo livre é a medida do eixo de arfagem que não adiciona inércia ao sistema. Outro efeito negativo que esta estrutura traz é o desalinhamento entre o centro de massa do quadricóptero e o eixo da estrutura, o que faz com forças não mapeadas na modelagem dinâmica surjam.

Figura 6.2 – CAD 3D da estrutura de teste giroscópica.



Fonte: Elaboração do próprio autor.

Figura 6.3 – CAD 3D da estrutura de teste giroscópica com o quadrrirrotor acoplado.



Fonte: Elaboração do próprio autor

Figura 6.4 – Estrutura de teste giroscópica com o quadrirrotor acoplado.



Fonte: Elaboração do próprio autor

Na próxima seção serão apresentados os resultados encontrados nas simulações e ensaios aplicados ao quadrirrotor real fixado à estrutura de testes giroscópica.

7 Resultados

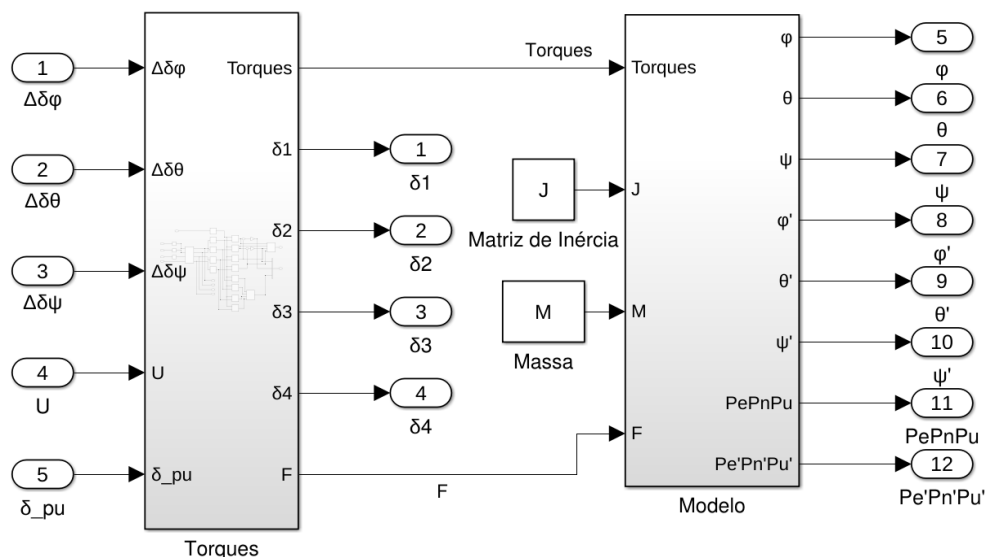
Nesta seção são apresentados e discutidos os resultados das simulações dos controladores contínuos aplicados ao modelo matemático dinâmico encontrado e dos controladores discretos aplicados ao quadrrorotor real fixado à estrutura de testes giroscópica. Para o caso dos controladores contínuos, serão apresentados tanto os resultados das simulações utilizando o modelo linear simplificado quanto utilizando o modelo não linear descrito pelas Equações (4.34) a (4.37).

7.1 Controladores Contínuos

Para os controladores contínuos, foram realizadas tanto simulações aplicadas ao modelo linear simplificado utilizado para a sintonia dos controladores quanto no modelo não linear, visto que o resultado neste tipo de modelo é mais fidedigno ao sistema real.

O modelo não linear utilizado para a realização das simulações pode ser visto na Figura 7.1.

Figura 7.1 – Modelo Não Linear utilizado nas simulações.

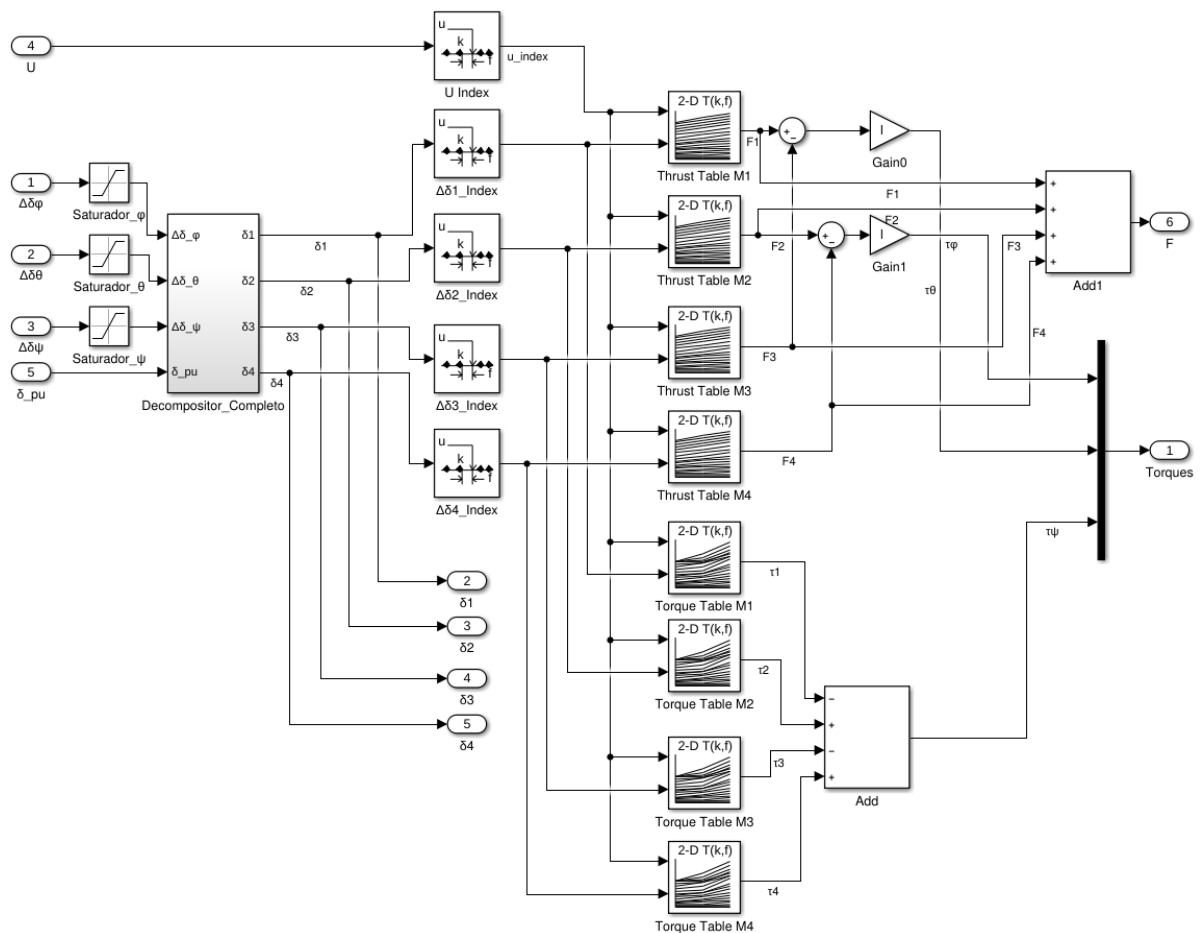


Fonte: Elaboração do próprio autor

O modelo não linear é composto por dois blocos: o bloco “Torques” e o bloco “Modelo”. O bloco “Torques” é responsável por produzir em sua saída os torques τ_φ ,

τ_θ e τ_ψ a partir das entradas $\Delta\delta_\varphi$, $\Delta\delta_\theta$, $\Delta\delta_\psi$ e δ_{pu} . O seu conteúdo pode ser visto na Figura 7.2. O bloco “Decompositor_Completo” é o mesmo exposto na seção 5. As tabelas de empuxo (*Thrust Table Mi*) e torque (*Torque Table Mi*) são alimentadas com os valores encontrados experimentalmente apresentados nas Figuras 4.4 e 4.7, respectivamente. A partir dos comandos de entrada, levando em consideração também a tensão U , este bloco realiza uma interpolação linear e produz os torques gerados pelo sistema. Esta interpolação se faz necessária para estimar os torques gerados a partir dos comandos aplicados aos motores, visto que os mesmos não são lineares. Os saturadores utilizados neste bloco são similares aos utilizados na seção 5. Porém, como no sistema real os comandos de largura de pulso são números inteiros, este saturador realiza este tipo de conversão. Por fim, saída do sistema F é utilizada no bloco “Modelo” para a estimativa de posição.

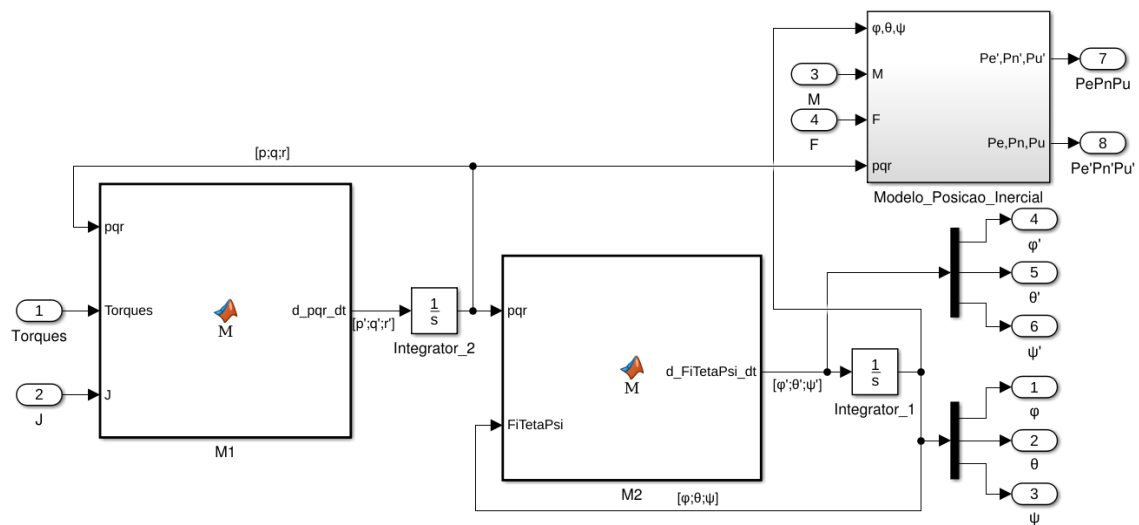
Figura 7.2 – Bloco gerador de torques.



Fonte: Elaboração do próprio autor.

O bloco “Modelo” é responsável por gerar as posições e velocidades angulares e lineares do sistema. Seu conteúdo pode ser visto na Figura 7.3. Este bloco é composto por duas funções e por um terceiro bloco que será discutido posteriormente. A função M1 corresponde à Equação (4.37) e a função M2 à Equação (4.36). Seus conteúdos podem ser vistos nos Códigos 7.1 e 7.2.

Figura 7.3 – Bloco “Modelo” para simulação não linear.



Fonte: Elaboração do próprio autor.

Código 7.1: Código do bloco M1 que representa a Equação (4.37).

```
function d_pqr_dt = M(pqr, Torques, J)

JI = inv(J);
m1 = [ 0      pqr(3) -pqr(2);
      -pqr(3)  0      pqr(1);
      pqr(2) -pqr(1)  0];
d_pqr_dt = JI*(m1*J*pqr + Torques);
```

Código 7.2: Código do bloco M2 que representa a Equação (4.36).

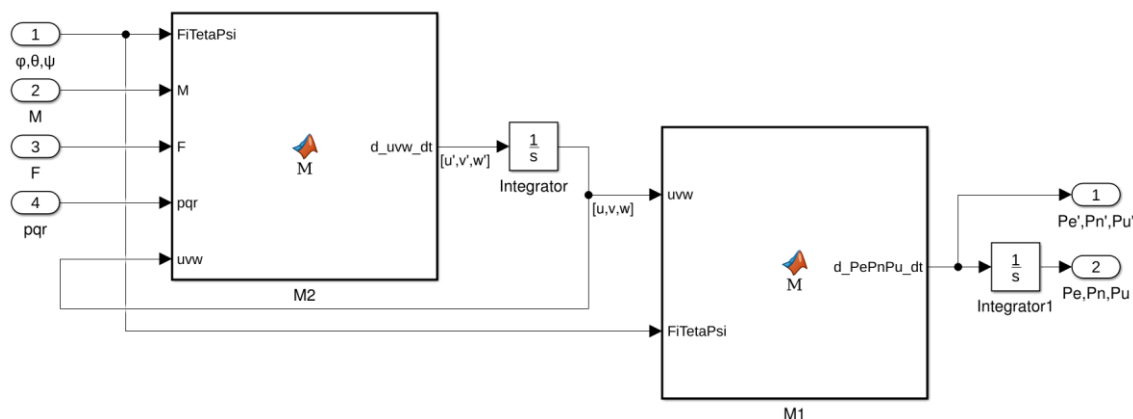
```
function d_FiTetaPsi_dt = M(pqr, FiTetaPsi)

m1 = [1, sin(FiTetaPsi(1,1))*tan(FiTetaPsi(2,1)), cos(FiTetaPsi(1,1))*tan(FiTetaPsi(2,1));
      0, cos(FiTetaPsi(1,1)), (-1)*sin(FiTetaPsi(1,1));      -pqr(3)  0      pqr(1);
      0, sin(FiTetaPsi(1,1))/cos(FiTetaPsi(2,1)), cos(FiTetaPsi(1,1))/cos(FiTetaPsi(2,1))];
d_FiTetaPsi_dt = m1*pqr;
```

Por fim, o bloco “Modelo_Posicao_Inercial” é responsável por determinar as posições e velocidades no referencial inercial. Seu conteúdo pode ser visto na Figura 7.4. Este bloco é composto por duas funções. A função M1 corresponde à

Equação (4.35) e a função M2 à Equação (4.34). Seus conteúdos podem ser vistos nos Códigos 7.3 e 7.4.

Figura 7.4 – Bloco “Modelo_Posicao_Inercial” para simulação não linear.



Fonte: Elaboração do próprio autor.

Código 7.3: Código do bloco M1 que representa a Equação (4.35).

```
function d_PePnPu_dt = M(uvw, FiTetaPsi)

u = uvw(1,1);
v = uvw(2,1);
w = uvw(3,1);

Fi = FiTetaPsi(1,1);
Teta = FiTetaPsi(2,1);
Psi = FiTetaPsi(3,1);

m1 = [ cos(Teta)*cos(Psi), sin(Fi)*sin(Teta)*cos(Psi)-cos(Fi)*sin(Psi),
        cos(Fi)*sin(Teta)*cos(Psi)+sin(Fi)*sin(Psi);
        cos(Teta)*sin(Psi), sin(Fi)*sin(Teta)*sin(Psi)+cos(Fi)*cos(Psi),
        cos(Fi)*sin(Teta)*sin(Psi)-sin(Fi)*cos(Psi);
        (-1)*sin(Teta), sin(Fi)*cos(Teta), cos(Fi)*cos(Teta) ];

JI = inv(J);
m1 = [ 0 pqr(3) -pqr(2);
        -pqr(3) 0 pqr(1);
        pqr(2) -pqr(1) 0 ];
d_pqr_dt = JI*(m1*J*pqr + Torques);
```

Código 7.4: Código do bloco M2 que representa a Equação (4.34).

```
function d_uvw_dt = M(FiTetaPsi, M,F, pqr, uvw)

g = 9.81;
Fi = FiTetaPsi(1,1);
Teta = FiTetaPsi(2,1);
Psi = FiTetaPsi(3,1);

p = pqr(1,1);
q = pqr(2,1);
```

```

r = pqr(3,1);

u = uvw(1,1);
v = uvw(2,1);
w = uvw(3,1);

v0 = [r*v - q*w;
      p*w - r*v;
      q*u - p*w];

v1 = [(-1)*g*sin(Teta);
      g*cos(Teta)*sin(Fi);
      (-1)*g*cos(Teta)*cos(Fi) + F/M];

d uvw dt = v0 + v1;

```

A seguir serão exibidos os resultados encontrados para cada um dos controladores desenvolvidos, tanto aplicados ao modelo linear simplificado quanto ao modelo não linear. Discussões sobre os resultados obtidos são apresentadas ao final de cada subseção.

7.1.1 PID

Os resultados encontrados nas simulações realizadas com os controladores PIDs podem ser vistos nas subseções seguintes.

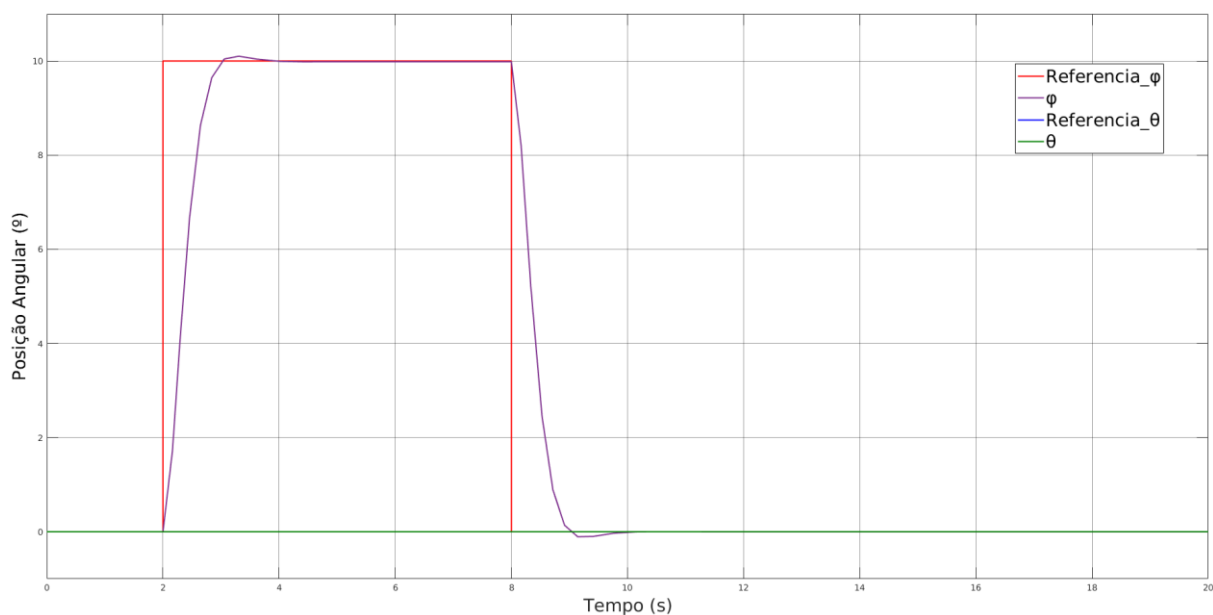
7.1.1.1 Modelo Linear Simplificado

Para os controladores PIDs aplicados ao modelo linear simplificado, foi utilizado o diagrama de blocos apresentado na Figura 5.5. Dele foram extraídas tanto as posições angulares de φ e θ quanto os comandos aplicados aos motores: δ_1 a δ_4 . A entrada do sistema se consiste de um degrau de amplitude 10° , com subida no tempo $t = 2s$ e descida no tempo $t = 8s$.

Duas simulações foram realizadas. Na primeira, o degrau foi aplicado ao ângulo φ e a referência do ângulo θ foi mantida em zero. Já na segunda simulação, o degrau foi aplicado ao ângulo θ e a referência do ângulo φ foi mantida em zero.

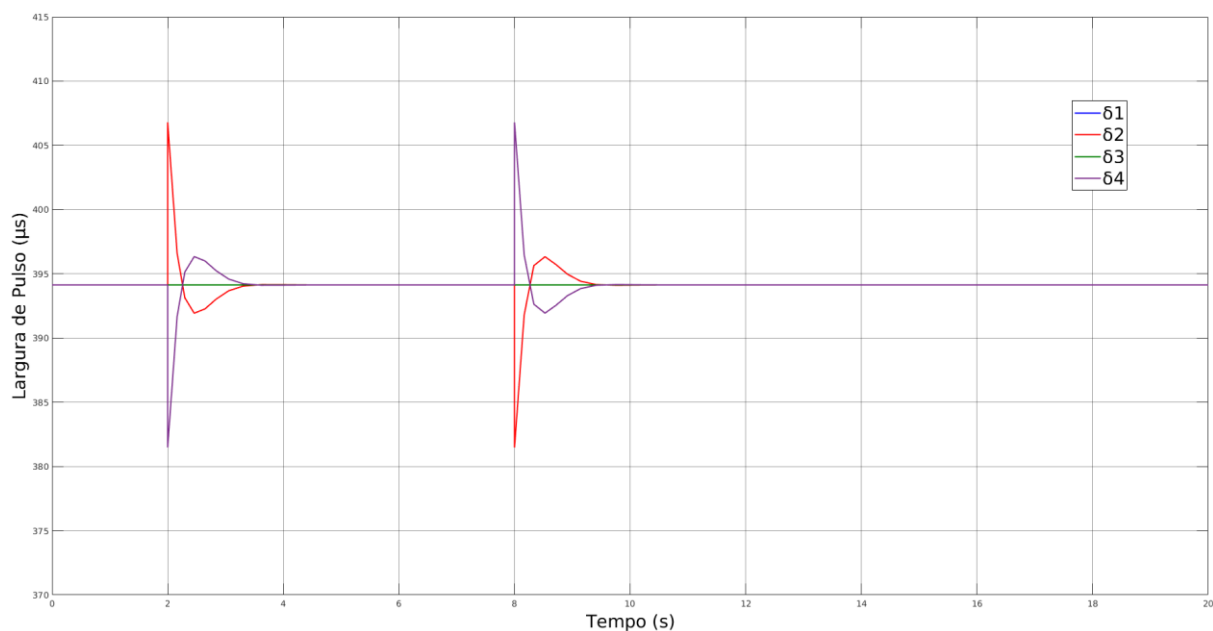
Os resultados encontrados para os ângulos φ e θ nas duas simulações podem ser vistos nas Figuras 7.5 e 7.7. Já os comandos aplicados aos motores nas duas simulações podem ser vistos nas Figuras 7.6 e 7.8. A tabela contendo os valores dos ganhos dos controladores que foram sintonizados na seção 5 pode ser vista na Tabela 7.1.

Figura 7.5 – Simulação 1: Controladores PID's Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).



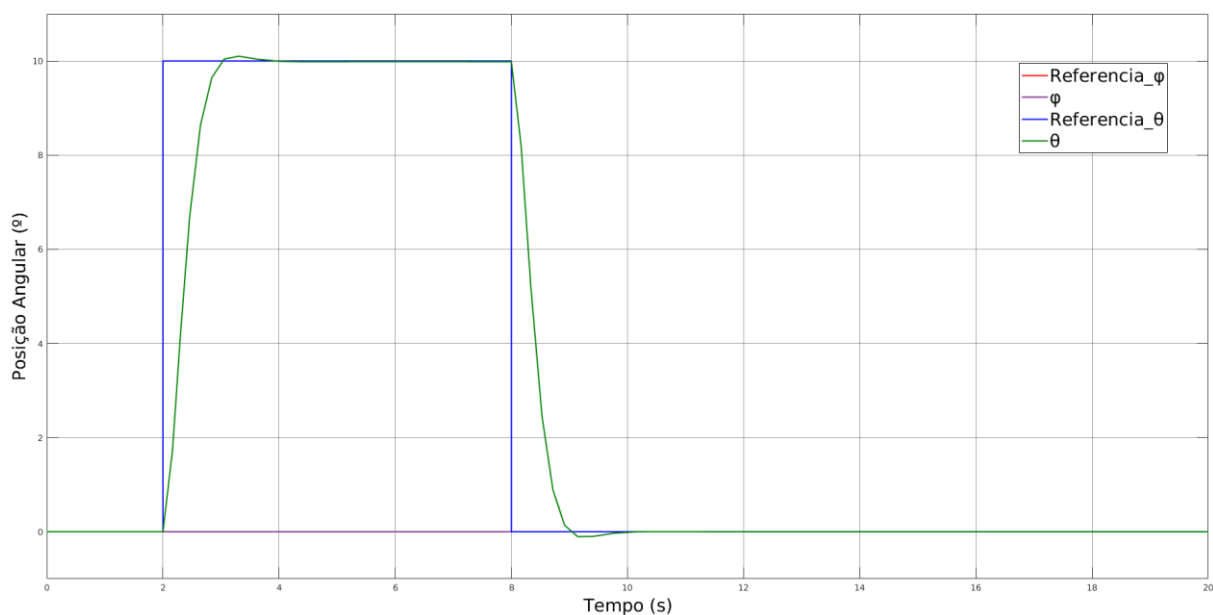
Fonte: Elaboração do próprio autor.

Figura 7.6 – Simulação 1: Saídas δ_1 a δ_4 dos Controladores PID's Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).



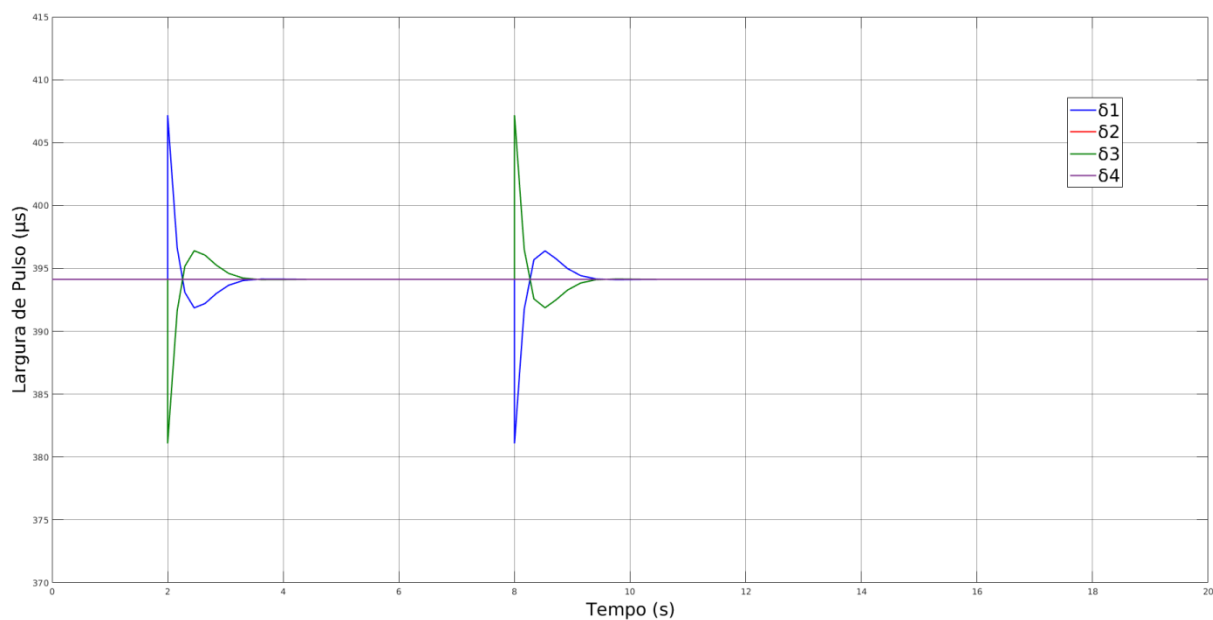
Fonte: Elaboração do próprio autor.

Figura 7.7 – Simulação 2: Controladores PID's Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

Figura 7.8 – Simulação 2: Saídas δ_1 a δ_4 dos Controladores PID's Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

Tabela 7.1 – Ganhos dos controladores PIDs contínuos.

Ganhos do Controlador PID de φ		Ganhos do Controlador PID de θ	
k_{p0}	160,8192	k_{p0}	165,1177
k_{p1}	-7,0088	k_{p1}	-7,1709
k_{i0}	-0,3189	k_{i0}	-0,3188
k_{i1}	0,0014	k_{i1}	0,0014
k_{d0}	60,2515	k_{d0}	61,8685
k_{d1}	-2,6275	k_{d1}	-2,6885

Fonte: Elaboração do próprio autor.

Também foram calculados os erros quadráticos médios nas duas simulações a fim de se obter um índice para um comparativo numérico entre os desempenhos dos controladores desenvolvidos. A Tabela 7.2 apresenta tais índices para os controladores PIDs aplicados ao modelo linear simplificado.

Tabela 7.2 – Erros quadráticos médios obtidos pelos controladores PIDs contínuos aplicados a φ e θ (modelo linear simplificado).

Variável	Simulação 1	Simulação 2
φ	3,1293 °	0,0 °
θ	0,0 °	3,1286 °

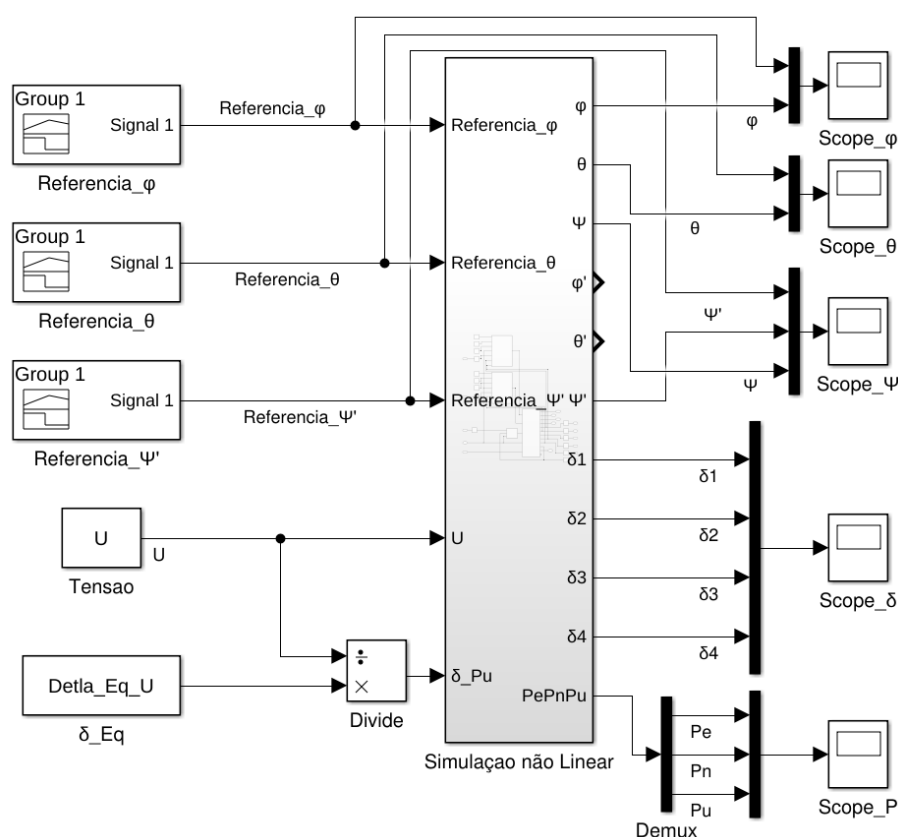
Fonte: Elaboração do próprio autor.

Através da análise dos resultados obtidos, é possível notar que os controladores PIDs desenvolvidos foram capazes de atingir a referência nos dois ângulos controlados. Os erros quadráticos médios obtidos nas duas simulações apresentados na Tabela 7.2 também mostram que os desempenhos dos controladores PIDs aplicados ao controle dos dois ângulos são bem próximos. Também é possível notar que os sobressinais para ambas as simulações são pequenos.

7.1.1.2 Modelo Não Linear

Para a realização das simulações no modelo não linear, foi montado o diagrama de blocos mostrado na Figura 7.9. Este será o mesmo para todos os controladores, mudando apenas os controladores utilizados em seu conteúdo. Seu conteúdo para o uso dos controladores PIDs pode ser visto na Figura 7.10.

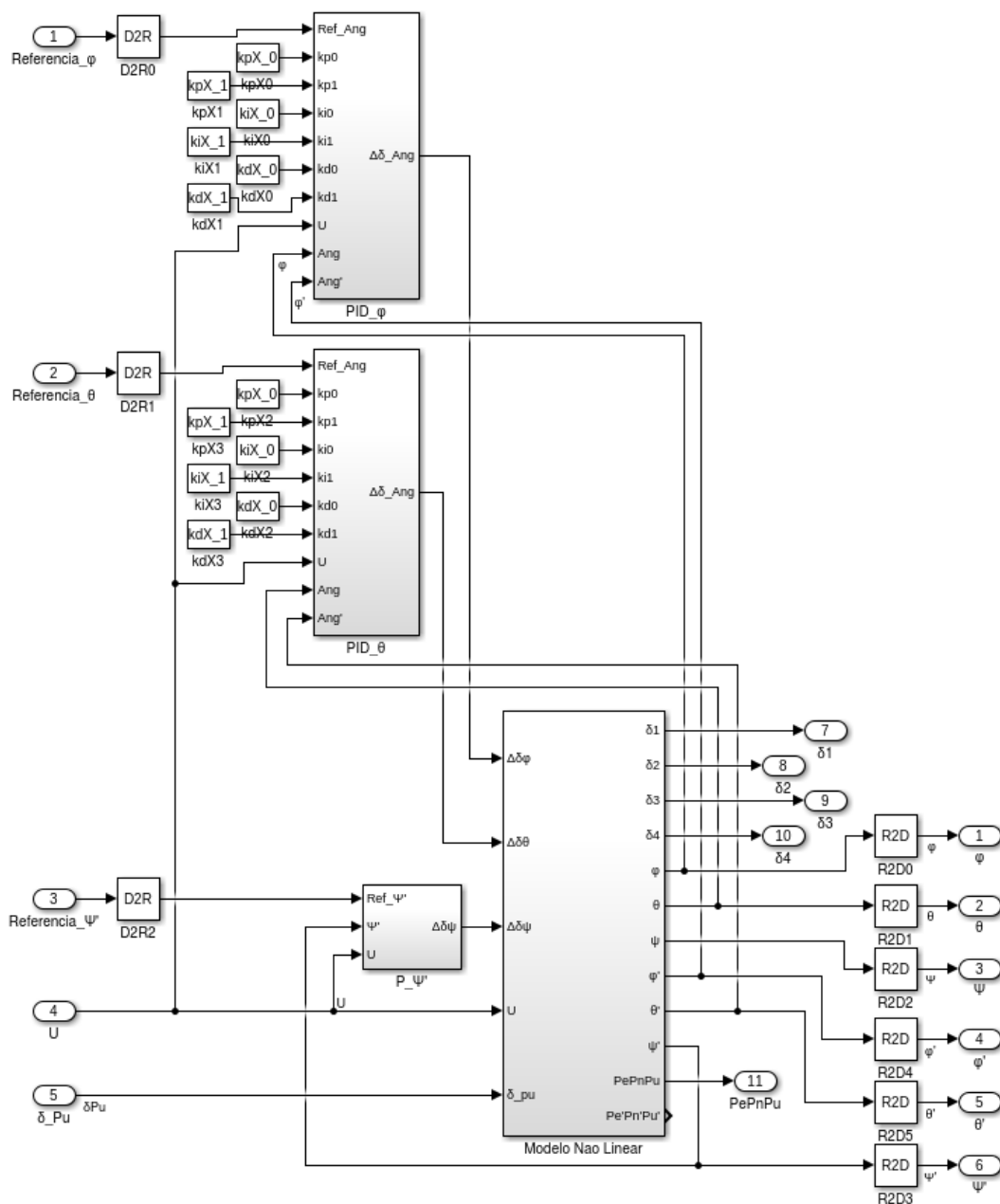
Figura 7.9 – Diagrama de Blocos utilizados para simulações aplicadas ao modelo não linear.



Fonte: Elaboração do próprio autor

Deste modelo de simulação, foram extraídas as posições angulares φ , θ e ψ , a velocidade angular ψ' e os comandos aplicados aos motores: δ_1 a δ_4 . Como no modelo não linear existe acoplamento entre os ângulos de Euler, quatro tipos de entradas foram simulados. Na simulação 1, um degrau de amplitude de 10° , assim como o aplicado no modelo linear simplificado, é aplicado a φ e uma entrada de amplitude zero é aplicado às demais referências. Na simulação 2, um degrau de amplitude 10° é aplicado a θ e as demais referências são mantidas em zero. Na

Figura 7.10 – Diagrama de Blocos utilizados para simulação dos controladores PID's aplicados ao modelo não linear.

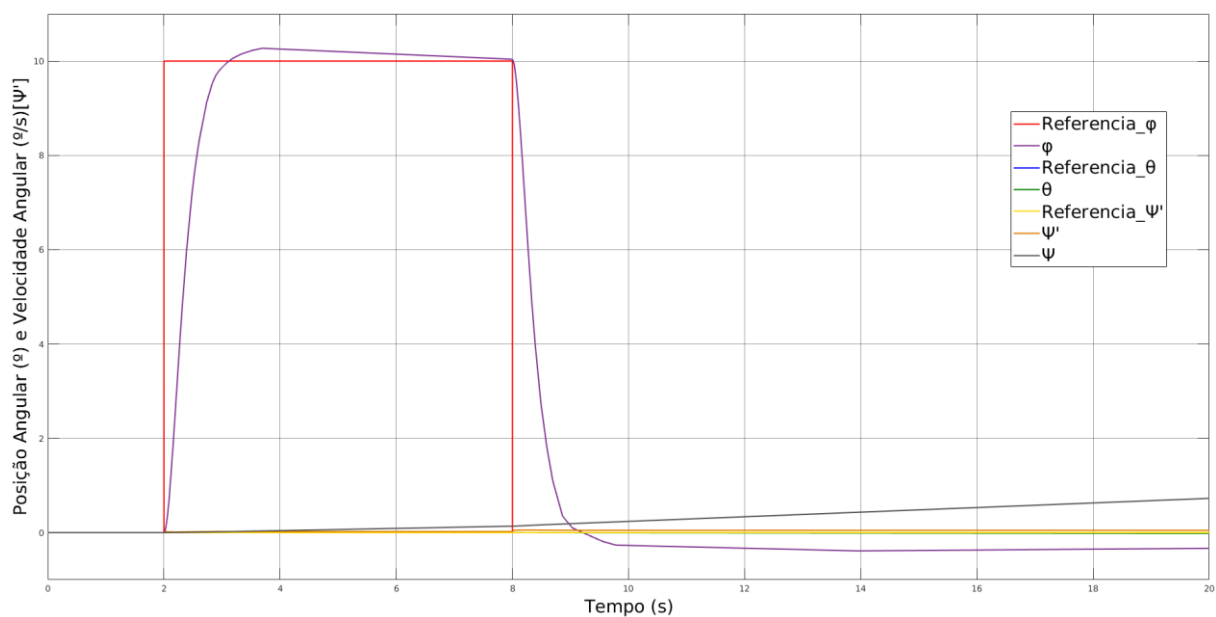


Fonte: Elaboração do próprio autor

simulação 3, ambos os ângulos φ e θ recebem um degrau de amplitude 10° e ψ' recebe zero como entrada. Por fim, na simulação 4, ambos os ângulos φ e θ recebem um degrau de amplitude 10° e ψ' recebe um degrau de amplitude $45^\circ/s$ durante 8 segundos a fim de se completar um ciclo de 360° no ângulo ψ . Este degrau na entrada de ψ' é aplicada em $t = 4s$ e retorna para zero em $t = 12s$.

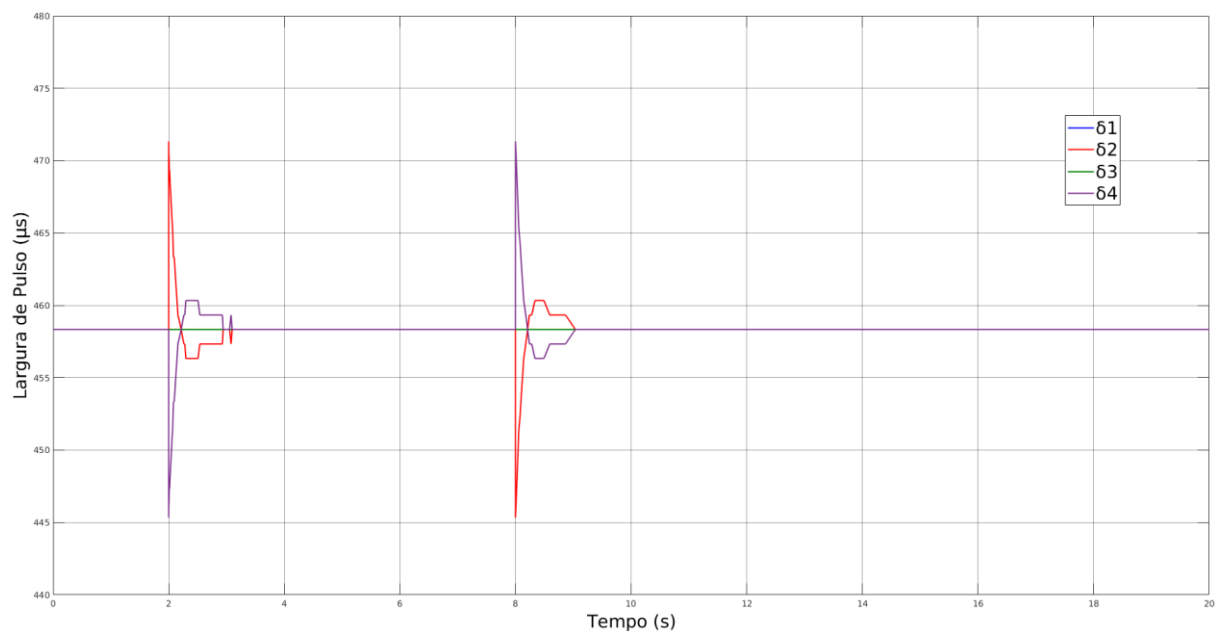
Os resultados destas simulações podem ser vistos nas Figuras 7.11 a 7.19.

Figura 7.11 – Simulação 1: Controladores Contínuos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



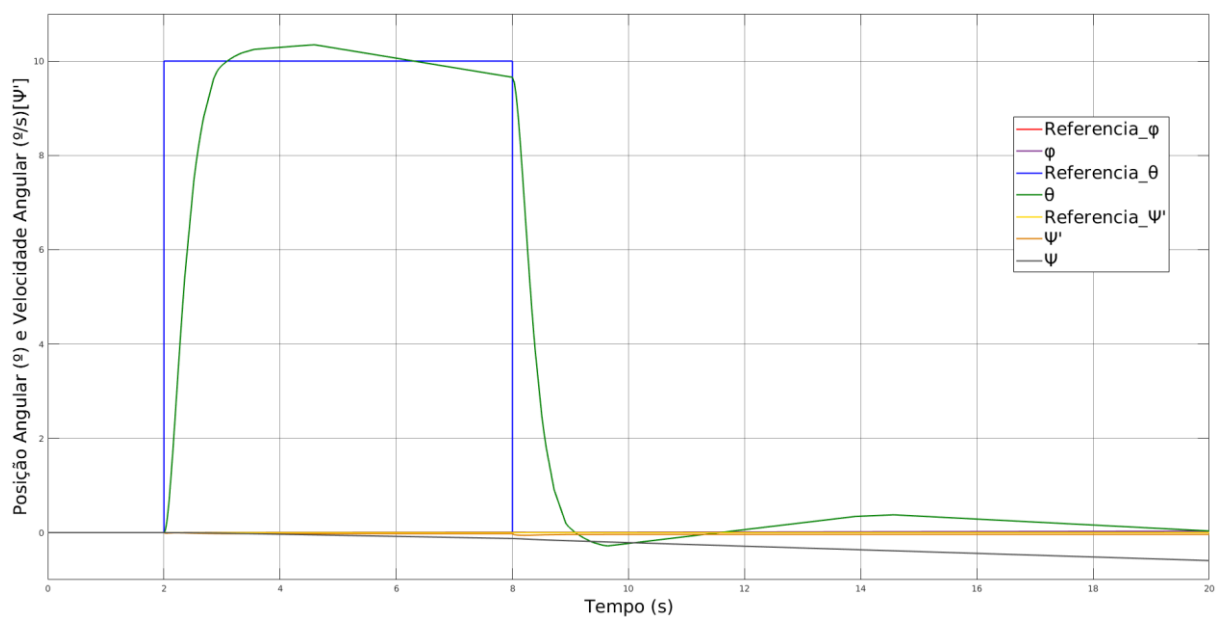
Fonte: Elaboração do próprio autor.

Figura 7.12 – Simulação 1: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos PID's aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



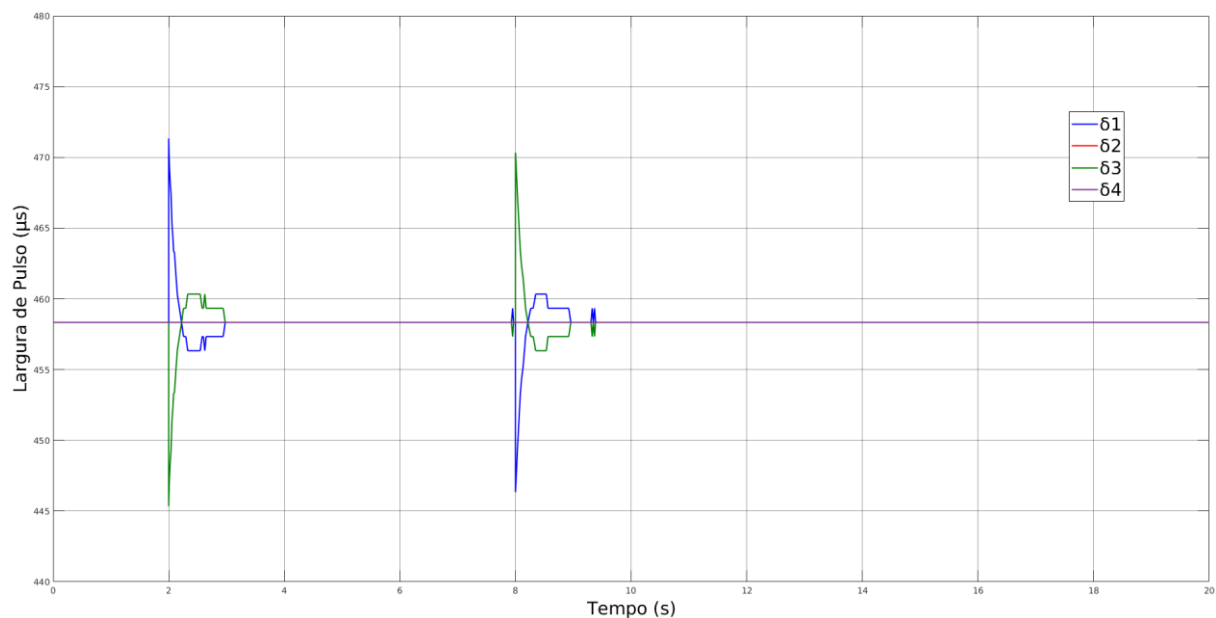
Fonte: Elaboração do próprio autor.

Figura 7.13 – Simulação 2: Controladores Contínuos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



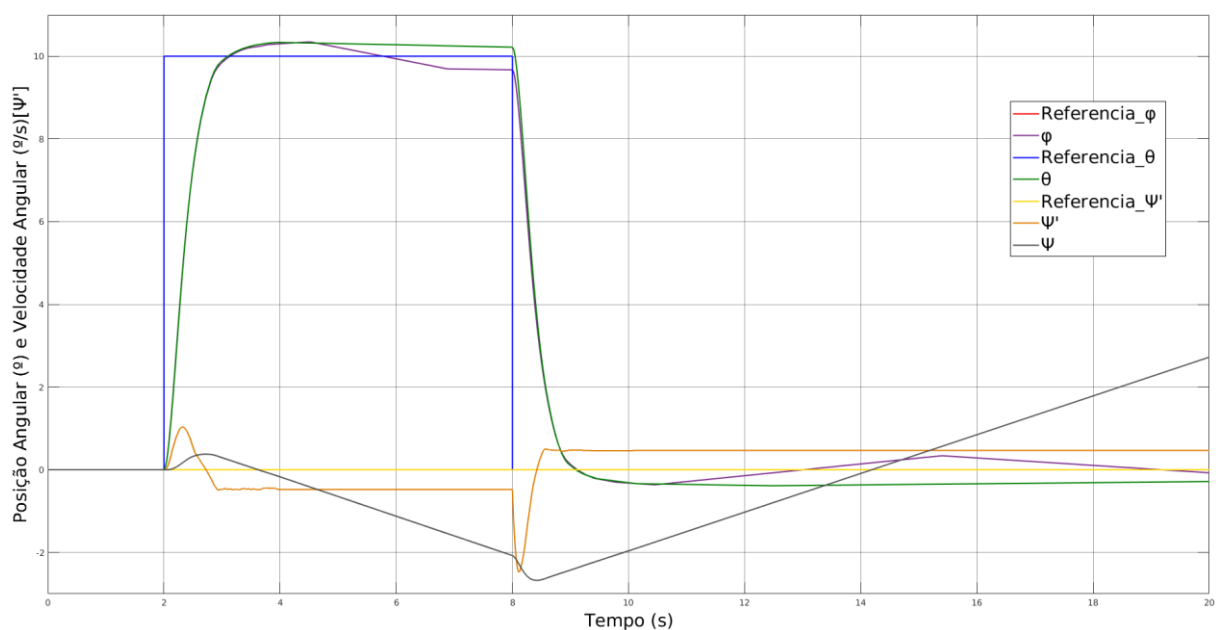
Fonte: Elaboração do próprio autor.

Figura 7.14 – Simulação 2: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos PID's aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



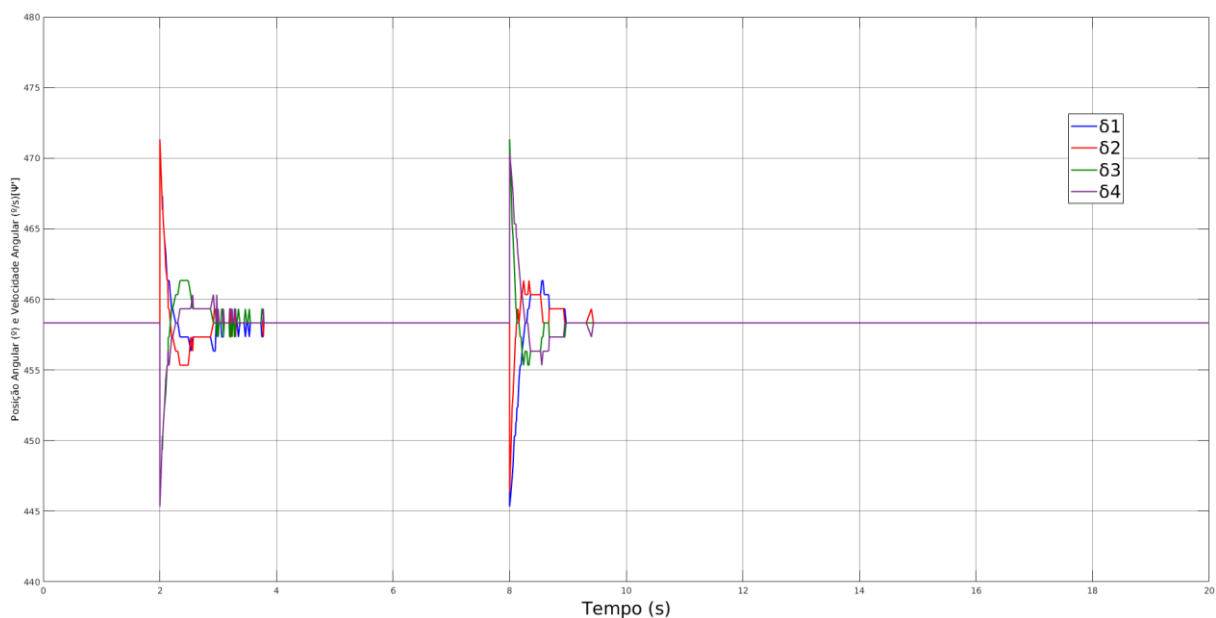
Fonte: Elaboração do próprio autor.

Figura 7.15 – Simulação 3: Controladores Contínuos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



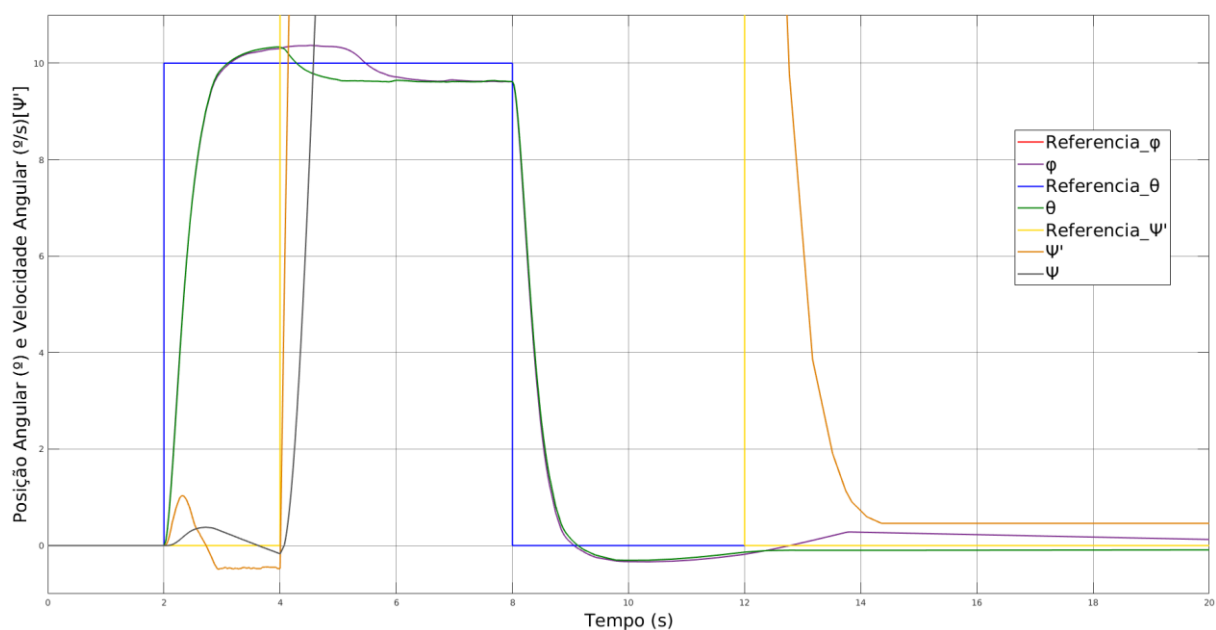
Fonte: Elaboração do próprio autor.

Figura 7.16 – Simulação 3: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos PID's aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



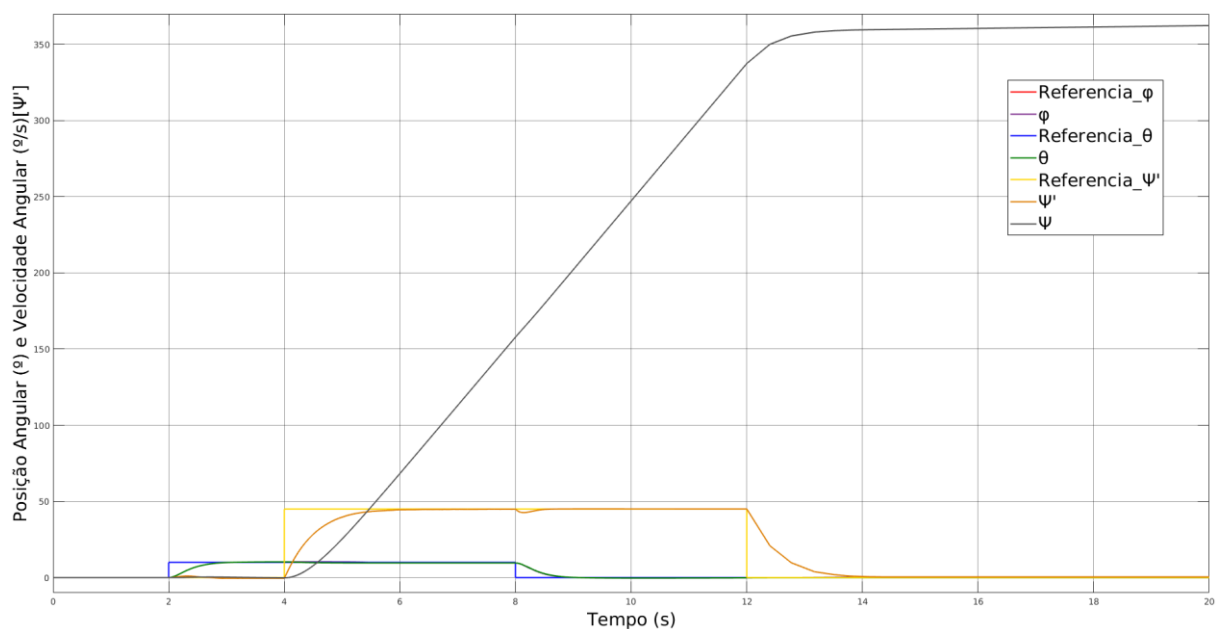
Fonte: Elaboração do próprio autor.

Figura 7.17 – Simulação 4: Controladores Contínuos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear) (zoom em φ e θ).



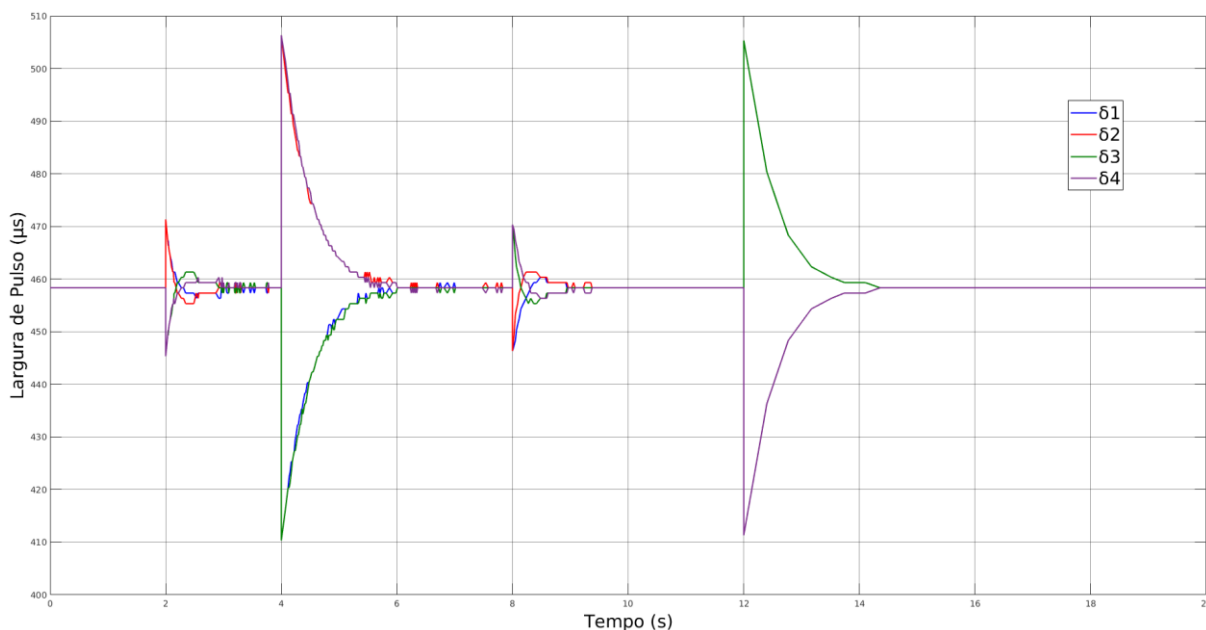
Fonte: Elaboração do próprio autor.

Figura 7.18 – Simulação 4: Controladores Contínuos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



Fonte: Elaboração do próprio autor.

Figura 7.19 – Simulação 4: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos PIDs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



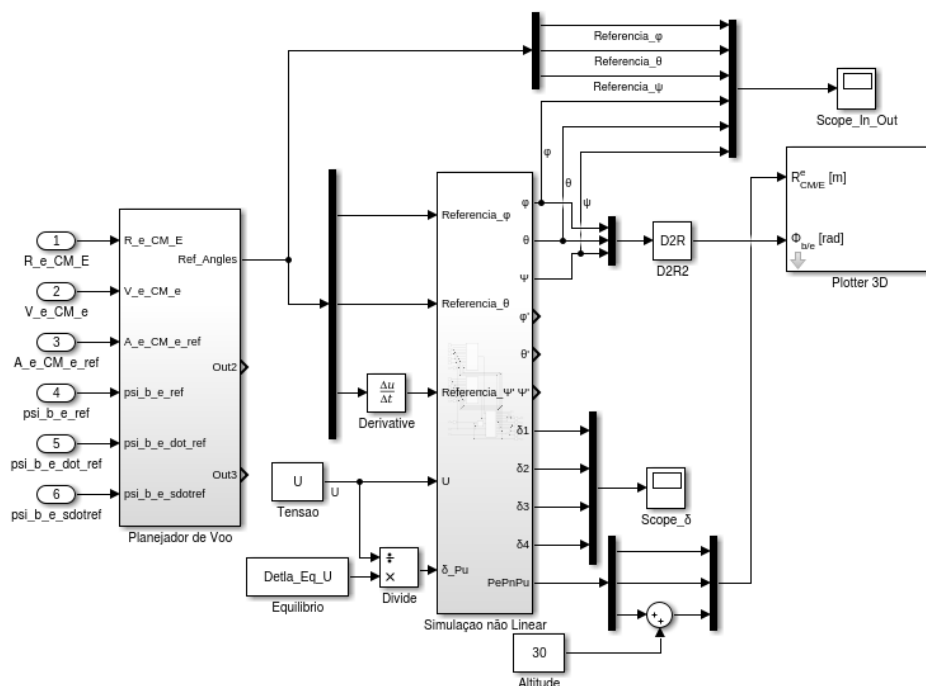
Fonte: Elaboração do próprio autor.

Por fim, também foi realizada uma simulação de voo utilizando um planejador de trajetória, que tem em sua saída os ângulos de referência. O diagrama de blocos que mostra a interação entre o planejador de voo e o sistema pode ser visto na Figura 7.20. Como o controlador de altitude não foi implementado neste trabalho, o planejador manterá uma altitude constante. No caso, a altitude escolhida foi de 30 metros. A trajetória planejada trata-se de uma espiral. O resultado da simulação de voo pode ser vista na Figura 7.21.

O comparativo entre os ângulos de referência e os ângulos obtidos pode ser visto na Figura 7.22. Na Figura 7.23 é dada uma ampliação na parte inferior da curva para facilitar a visualização dos ângulos φ e θ .

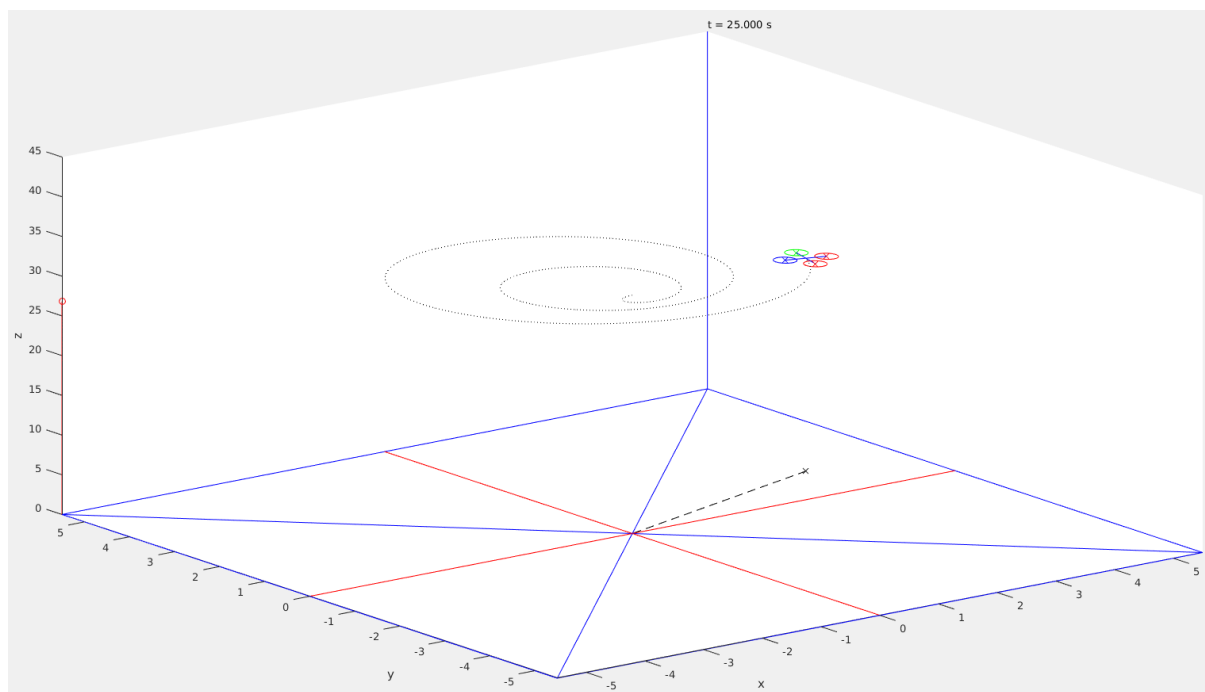
Assim como realizado para as simulações aplicadas ao modelo linear, aqui também foram calculados os erros quadráticos médios nas cinco simulações a fim de se obter um índice para um comparativo numérico entre os desempenhos dos controladores desenvolvidos aplicados ao modelo não linear. A Tabela 7.3 apresenta tais índices para os controladores PIDs aplicados ao controle de φ e θ em conjunto do controlador P aplicado ao controle de ψ' (modelo não linear). Como na simulação de voo a referência em guinada é sobre a posição angular e não sobre a sua velocidade, esta foi a variável utilizada para o cálculo de seu erro quadrático médio.

Figura 7.20 – Planejador de voo acoplado ao sistema.



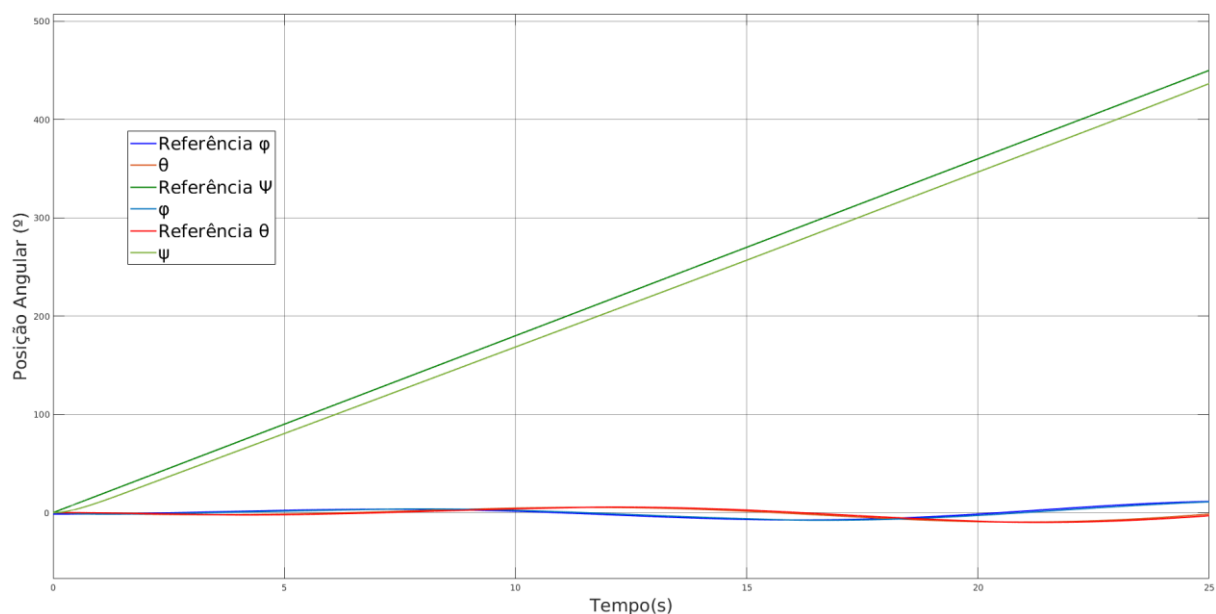
Fonte: Elaboração do próprio autor.

Figura 7.21 – Resultado da simulação de voo utilizando os controladores PID's e P.



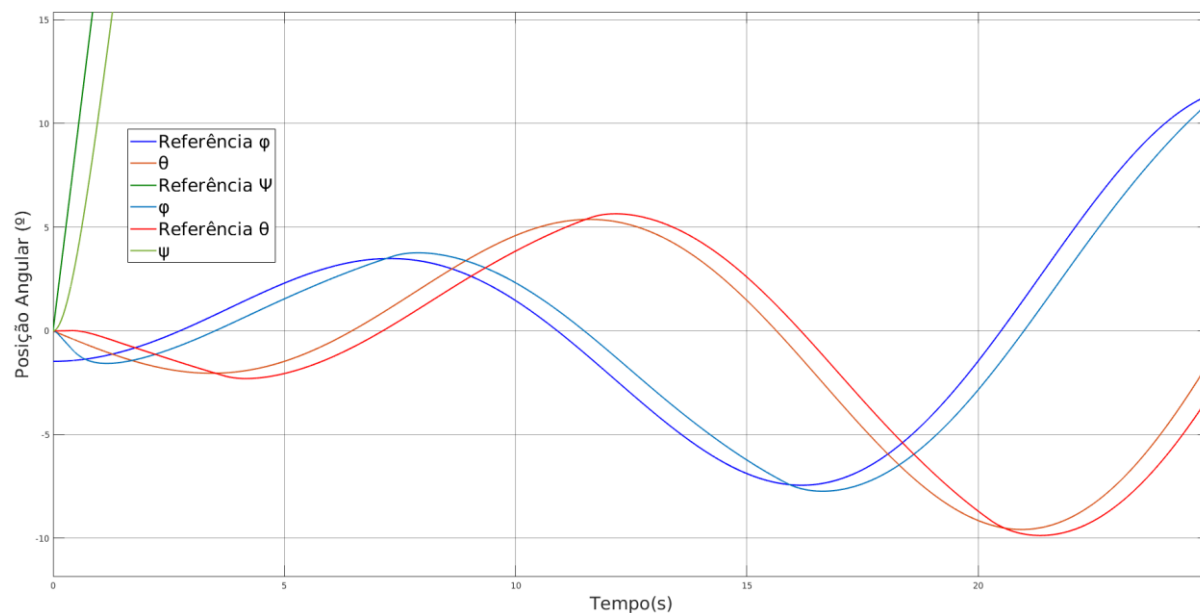
Fonte: Elaboração do próprio autor.

Figura 7.22 – Ângulos de Referência e entrada na simulação de voo utilizando os Controladores Contínuos PIDs e P.



Fonte: Elaboração do próprio autor.

Figura 7.23 – Ângulos de Referência e entrada na simulação de voo utilizando os Controladores Contínuos PIDs e P (zoom).



Fonte: Elaboração do próprio autor.

Tabela 7.3 – Erros quadráticos médios obtidos pelos controladores PIDs contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear).

Variável	Simulação 1	Simulação 2	Simulação 3	Simulação 4	Simulação voo
φ	4,2309 °	0,0117 °	3,5294 °	2,5765 °	0,9369 °
θ	0,0052 °	3,7178 °	3,6342 °	2,5928 °	0,8407 °
ψ'	0,0361 °/s	0,0324 °/s	0,6906 °/s	9,0485 °/s	-
ψ	-	-	-	-	11,6406 °

Fonte: Elaboração do próprio autor.

Analizando as saídas das simulações dos controladores PIDs aplicados ao modelo não linear, é possível notar que nas simulações 1 e 2, os ângulos de referência foram atingidos em um tempo bem próximo ao encontrado na simulação linear. Nestas simulações, o acoplamento dos demais ângulos é pequeno. É possível notar também que no controlador aplicado ao controle de φ , há um erro de regime permanente devido às características dos atuadores. Já na simulação 3, é possível notar que os três ângulos sofrem influência um do outro, inclusive o eixo de guinada. Suas saídas são diferentes das apresentadas nas simulações 1 e 2 e o controlador aplicado ao controle de θ apresenta um erro de regime permanente. Por fim, na simulação 4 é possível notar que a atuação no ângulo de guinada, ψ , também influencia o controle dos demais ângulos. O resultado da simulação de voo se mostrou eficiente visto que o quadricóptero traçou a rota programada e os controladores atingiram os ângulos desejados. Como para o ângulo ψ , o controlador foi projetado para atuar em sua taxa e não em sua posição, um *offset* constante foi encontrado visto que o mesmo possui um tempo de subida para que a taxa programada seja alcançada.

Analizando a Tabela 7.3, é possível notar que no modelo não linear, os erros quadráticos médios podem variar consideravelmente dependendo do tipo de entrada aplicada. Por exemplo, o erro quadrático médio obtido para o ângulo φ na simulação 1 é diferente do obtido das simulações 3 e 4, que possuem o mesmo tipo de referência neste ângulo.

7.1.2 LQR

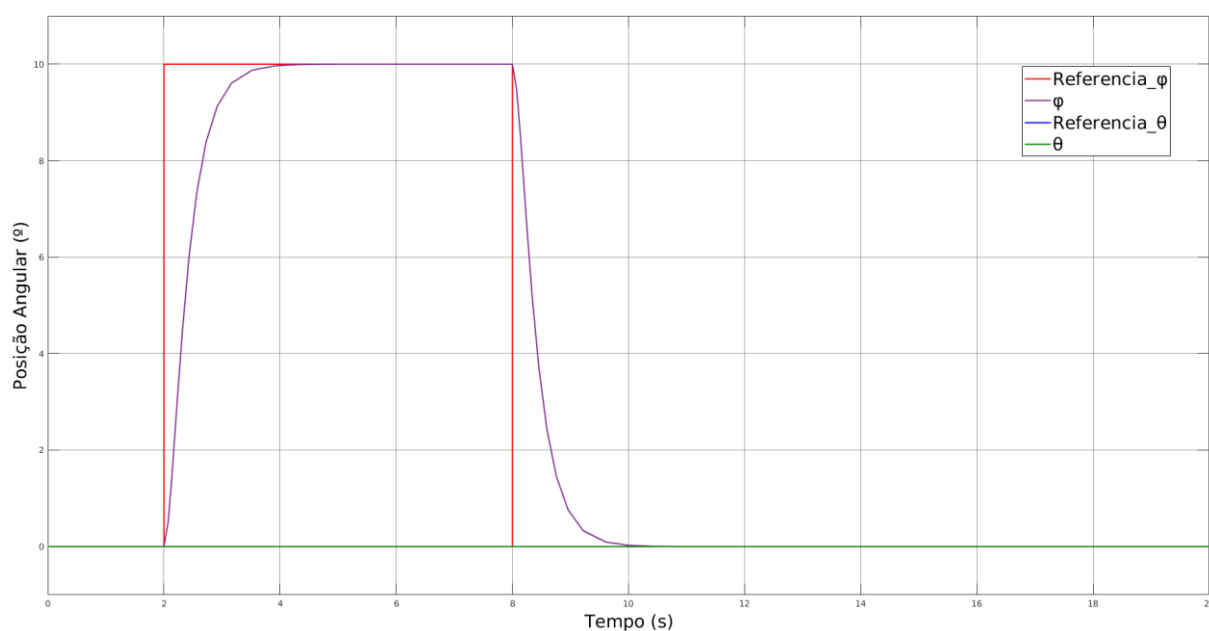
Os resultados encontrados nas simulações realizadas com os controladores LQRs podem ser vistos nas subseções seguintes.

7.1.2.1 Modelo Linear Simplificado

Para os controladores LQRs aplicados ao modelo linear simplificado, foi utilizado o diagrama de blocos apresentado na Figura 5.9. Dele foram extraídas tanto as posições angulares de φ e θ quanto os comandos aplicados aos motores: δ_1 a δ_4 . As entradas e simulações aplicadas ao sistema foram as mesmas aplicadas nos controladores PIDs.

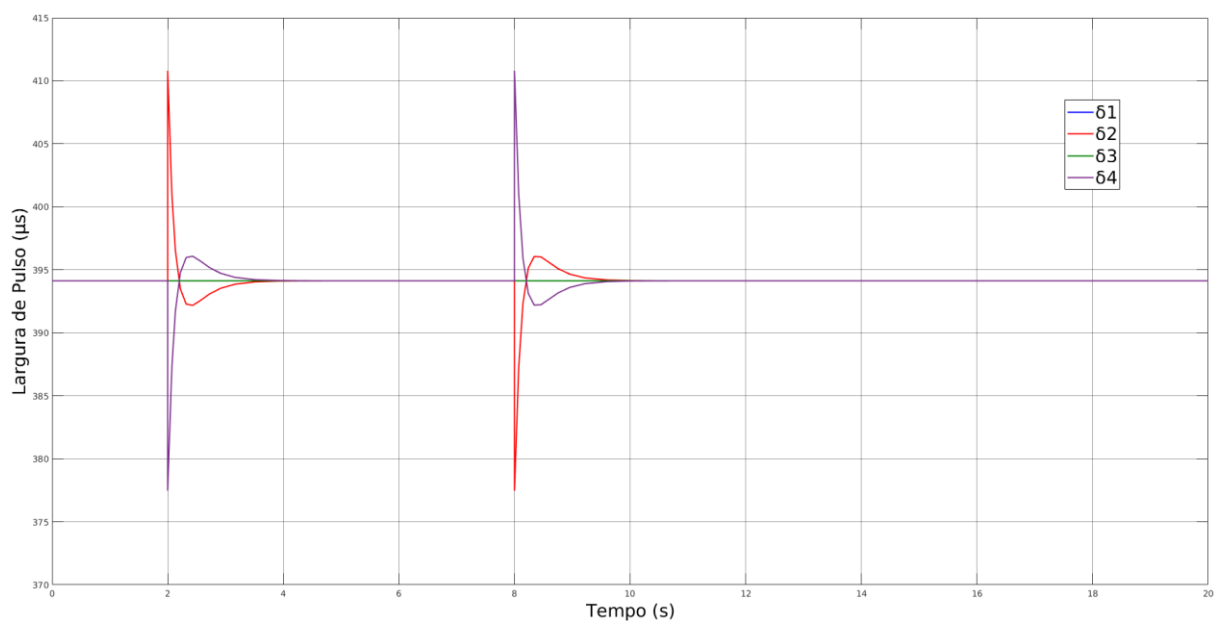
Os resultados encontrados para os ângulos φ e θ nas duas simulações podem ser vistos nas Figuras 7.24 e 7.26. Já os comandos aplicados aos motores nas duas simulações podem ser vistos nas Figuras 7.25 e 7.27. A tabela contendo os valores dos ganhos dos controladores que foram sintonizados na seção 5 pode ser vista na Tabela 7.4.

Figura 7.24 – Simulação 1: Controladores LQRs Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).



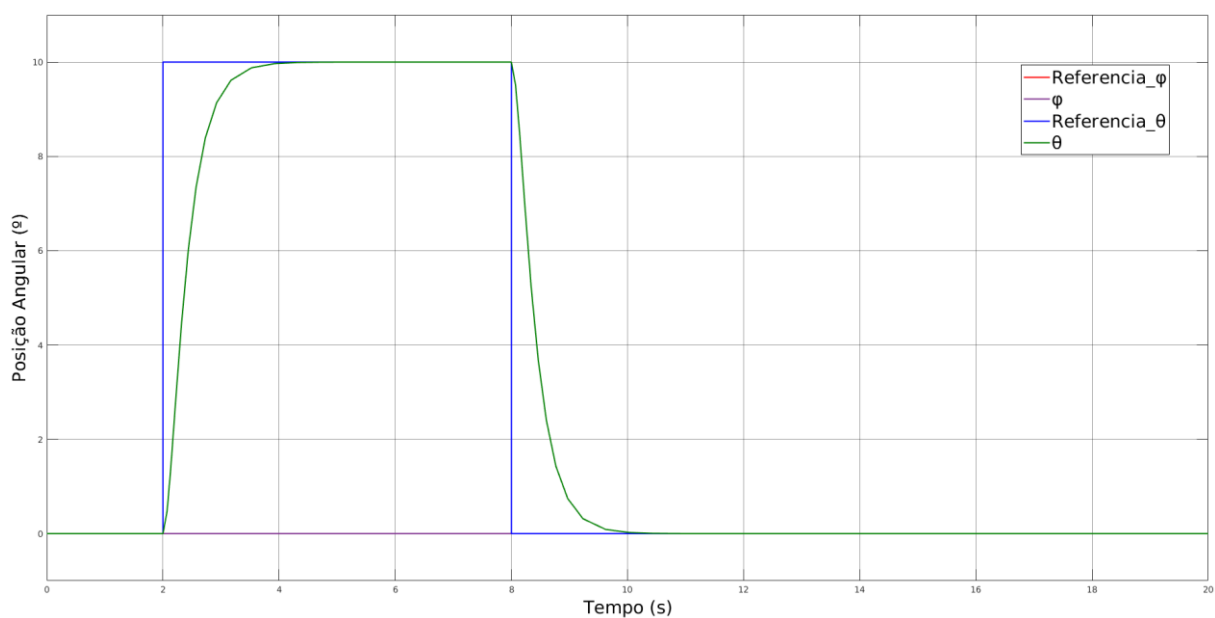
Fonte: Elaboração do próprio autor.

Figura 7.25 – Simulação 1: Saídas δ_1 a δ_4 dos Controladores LQRs Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).



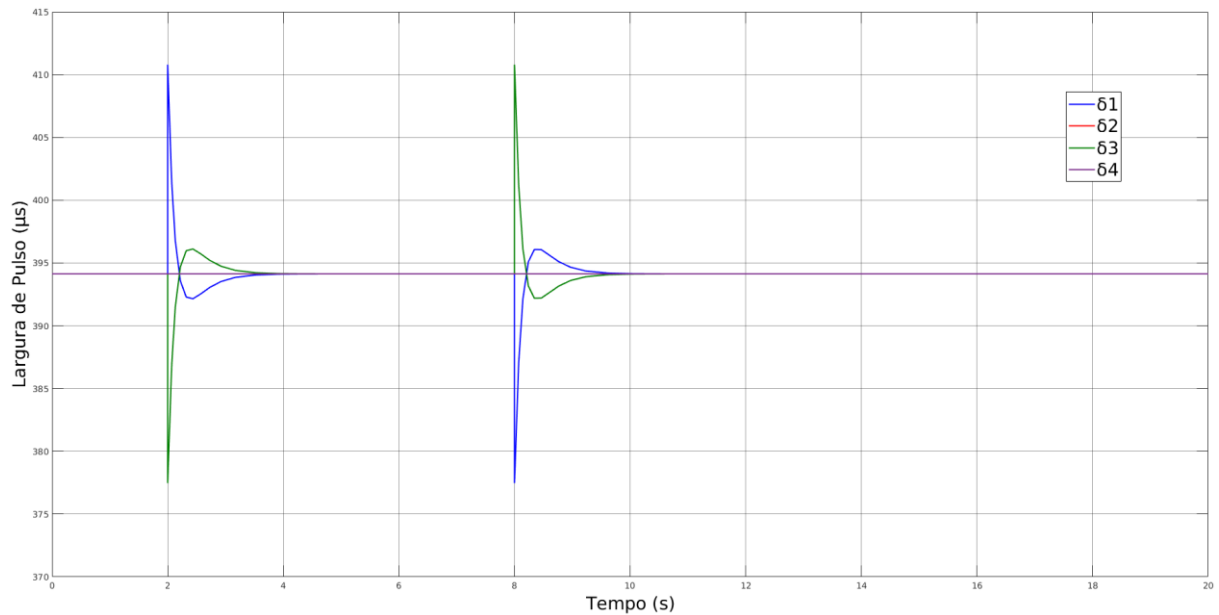
Fonte: Elaboração do próprio autor.

Figura 7.26 – Simulação 2: Controladores LQRs Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

Figura 7.27 – Simulação 2: Saídas δ_1 a δ_4 dos Controladores LQRs Contínuos aplicados ao controle de φ e θ (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

Tabela 7.4 – Ganhos dos controladores LQRs contínuos.

Ganhos Controlador LQR de φ		Ganhos Controlador LQR de θ	
k_1	95.4930	k_1	95.4930
k_{2_0}	49.7067	k_{2_0}	50.1635
k_{2_1}	-0.6269	k_{2_1}	-0.6419

Fonte: Elaboração do próprio autor.

A Tabela 7.5 apresenta os erros quadráticos médios obtidos durante as simulações pelos controladores LQRs aplicados ao modelo linear simplificado.

Tabela 7.5 – Erros quadráticos médios obtidos pelos controladores LQRs contínuos aplicados a φ e θ (modelo linear simplificado).

Variável	Simulação 1	Simulação 2
φ	3,6922 °	0,0 °
θ	0,0 °	3,7012 °

Fonte: Elaboração do próprio autor.

Através da análise dos resultados obtidos, é possível notar que os controladores LQRs desenvolvidos foram capazes de atingir a referência nos dois ângulos controlados. Os erros quadráticos médios obtidos nas duas simulações apresentados na Tabela 7.5 também mostram que os desempenhos dos controladores LQRs aplicados ao controle dos dois ângulos são bem próximos. Também é possível notar que para ambas as simulações a saída não apresentou sobressinal algum, embora o tempo para acomodação seja superior ao encontrado pelos controladores PIDs.

7.1.2.2 Modelo Não Linear

Para a realização da simulação no modelo não linear, foi utilizado o diagrama de blocos apresentado na Figura 7.9. Seu conteúdo para o uso dos controladores LQRs pode ser visto na Figura 7.28. Já o conteúdo do bloco controlador LQR apresentado na Figura 7.28 pode ser visto na Figura 7.29.

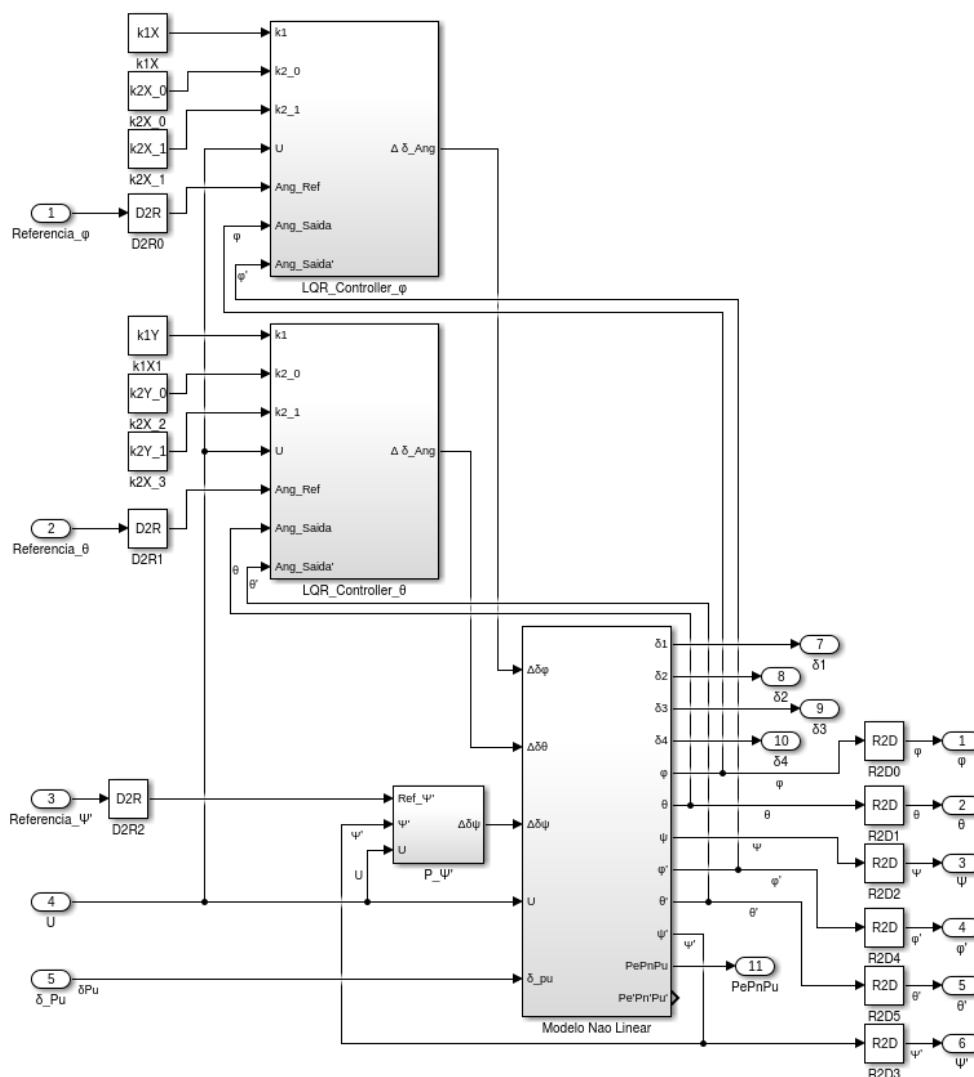
Deste modelo de simulação, foram extraídas as posições angulares φ , θ e ψ , a velocidade angular ψ' e os comandos aplicados aos motores: δ_1 a δ_4 . Os mesmos quatro tipos de simulações aplicados aos controladores PIDs foram aplicados a estes controladores.

Os resultados destas simulações podem ser vistos nas Figuras 7.30 a 7.38.

Por fim, também foi realizada uma simulação de voo utilizando o mesmo planejador de trajetória utilizado com os controladores PIDs mostrado na subseção anterior. O resultado da simulação de voo pode ser vista na Figura 7.39.

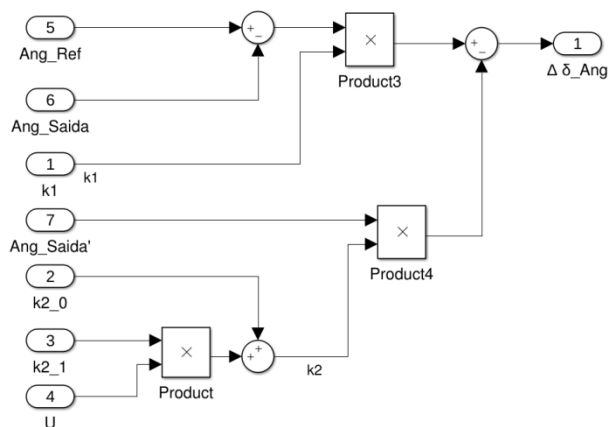
Já comparativo entre os ângulos de referência e os ângulos alcançados nesta simulação de voo pode ser visto na Figura 7.40. Na Figura 7.41 é dada uma ampliação na parte inferior da curva para facilitar a visualização dos ângulos φ e θ .

Figura 7.28 – Diagrama de Blocos utilizados para simulação dos controladores LQRs aplicados ao modelo não linear.



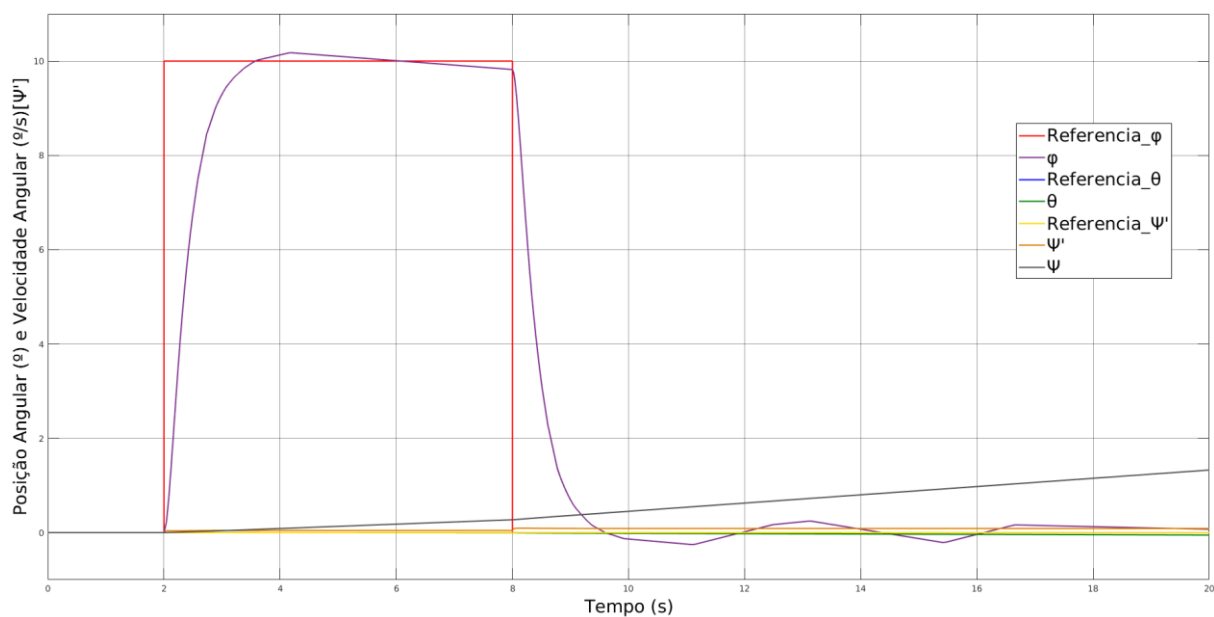
Fonte: Elaboração do próprio autor.

Figura 7.29 – Bloco controlador LQR utilizado para simulação.



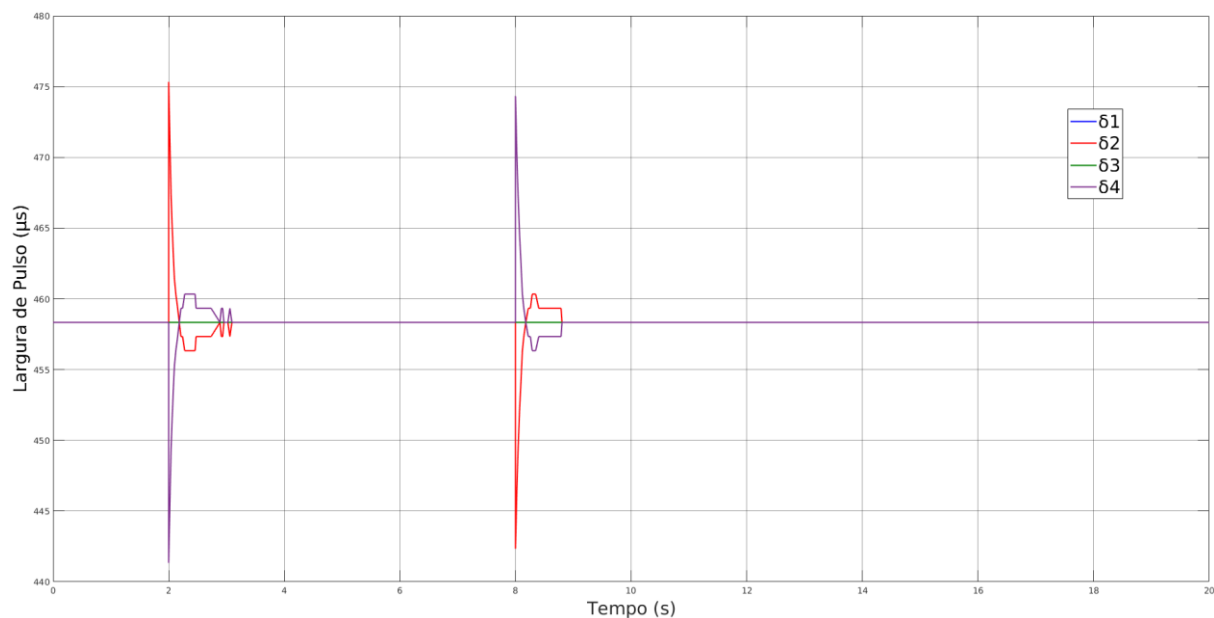
Fonte: Elaboração do próprio autor.

Figura 7.30 – Simulação 1: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



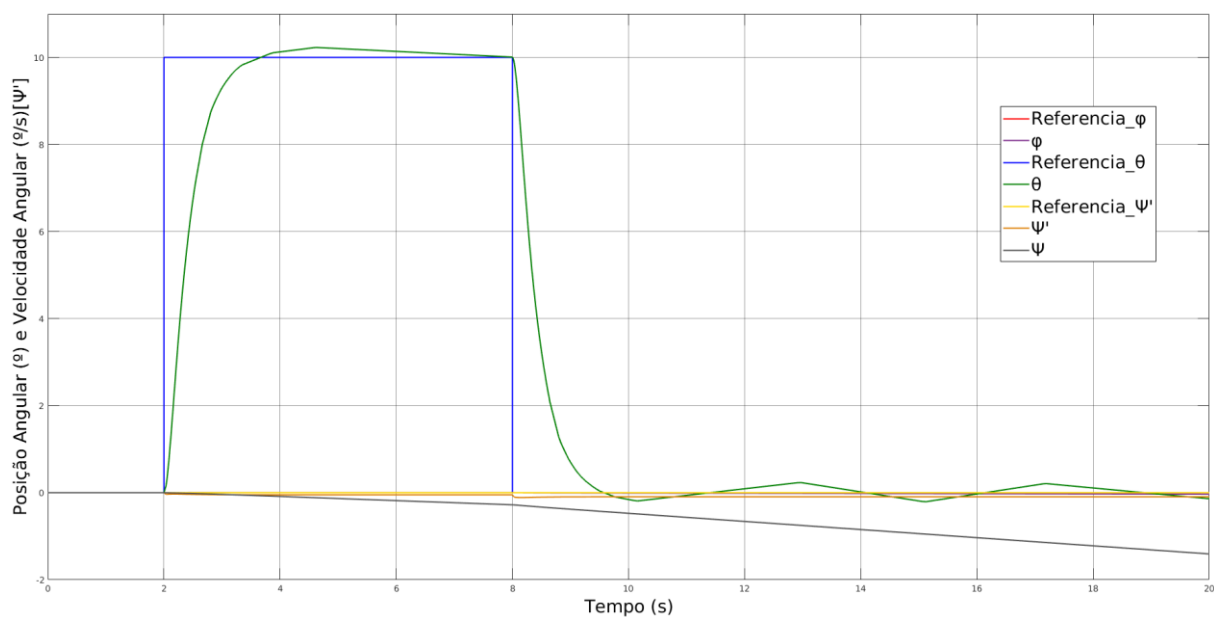
Fonte: Elaboração do próprio autor.

Figura 7.31 – Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos LQRs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



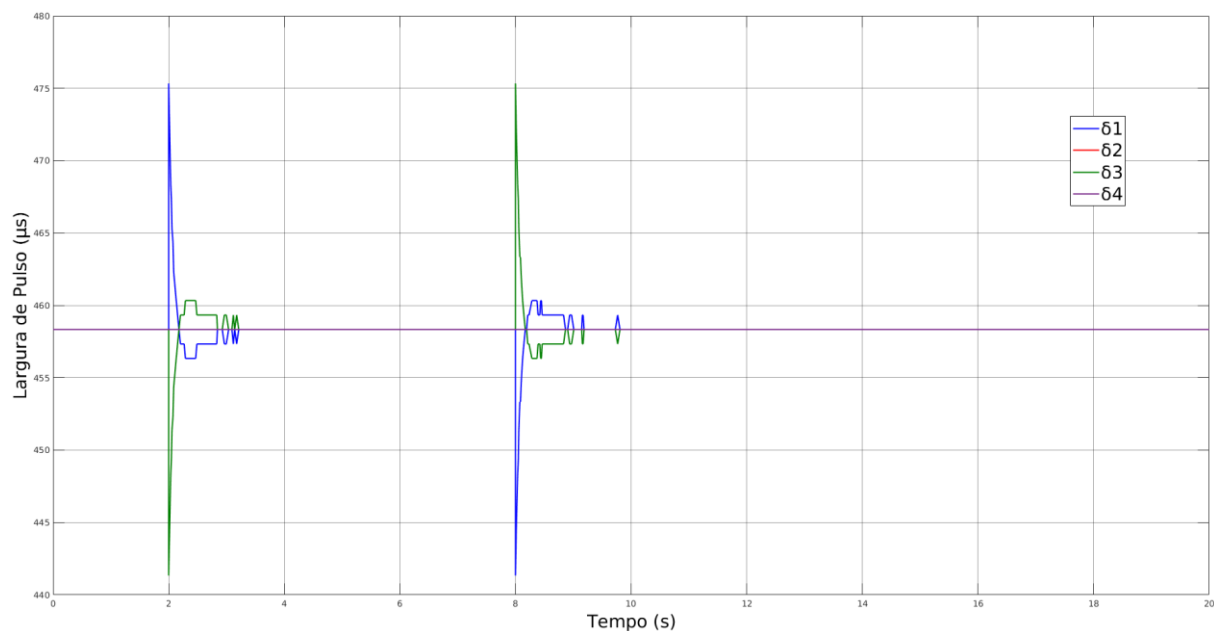
Fonte: Elaboração do próprio autor.

Figura 7.32 – Simulação 2: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



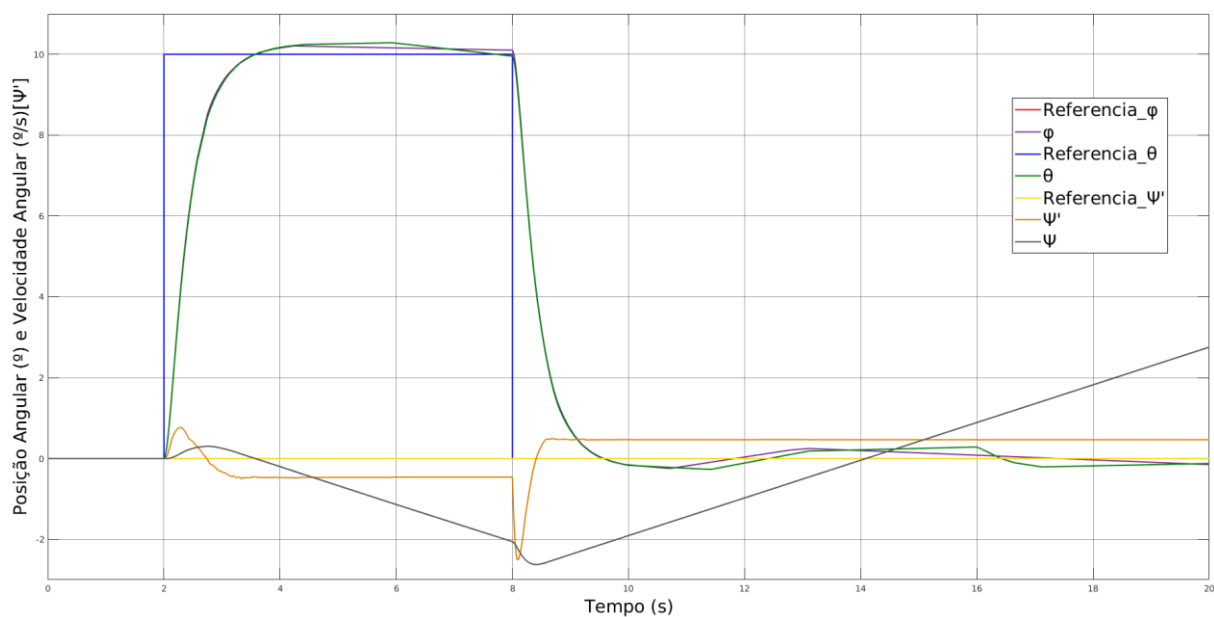
Fonte: Elaboração do próprio autor.

Figura 7.33 – Simulação 2: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos LQRs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



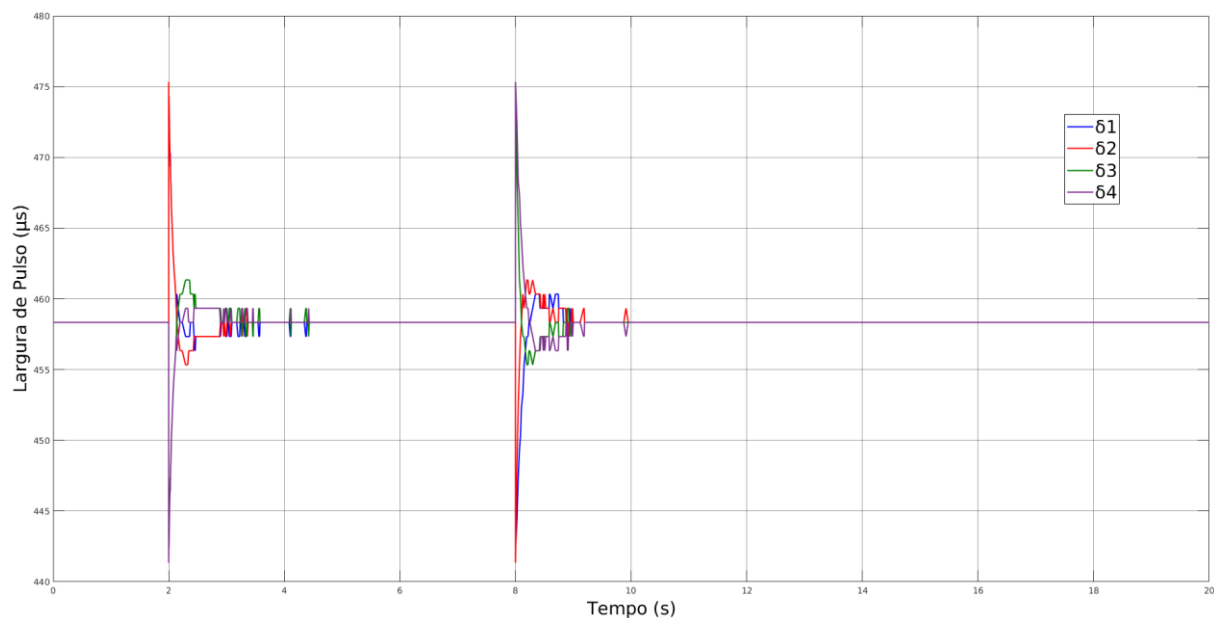
Fonte: Elaboração do próprio autor.

Figura 7.34 – Simulação 3: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



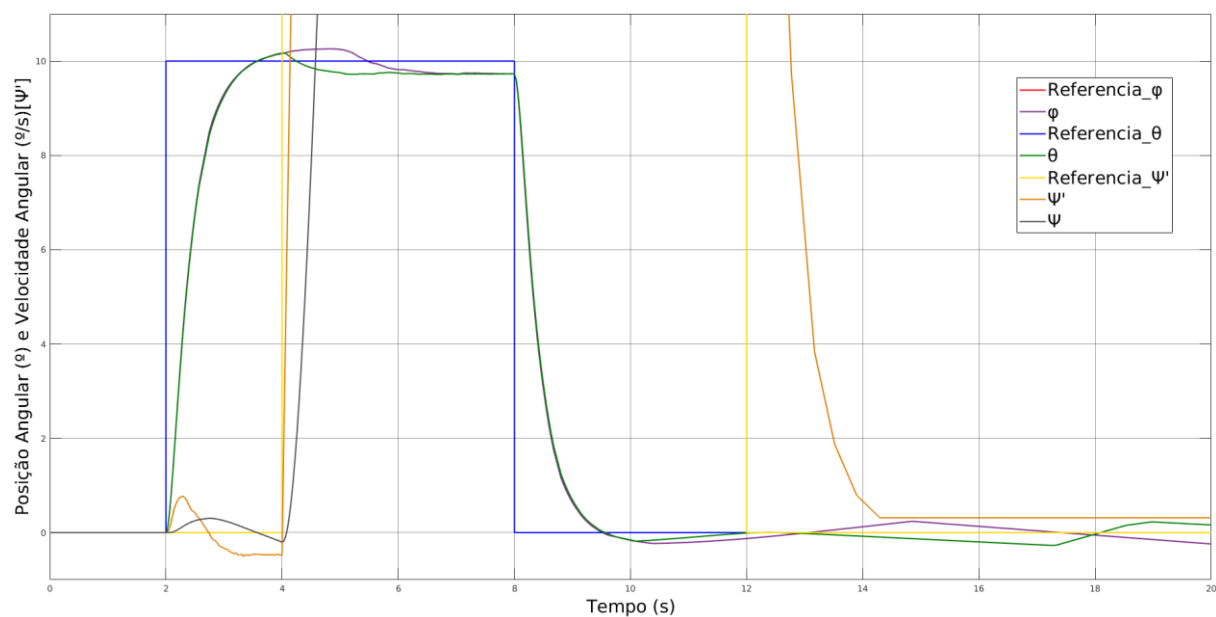
Fonte: Elaboração do próprio autor.

Figura 7.35 – Simulação 3: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos LQRs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



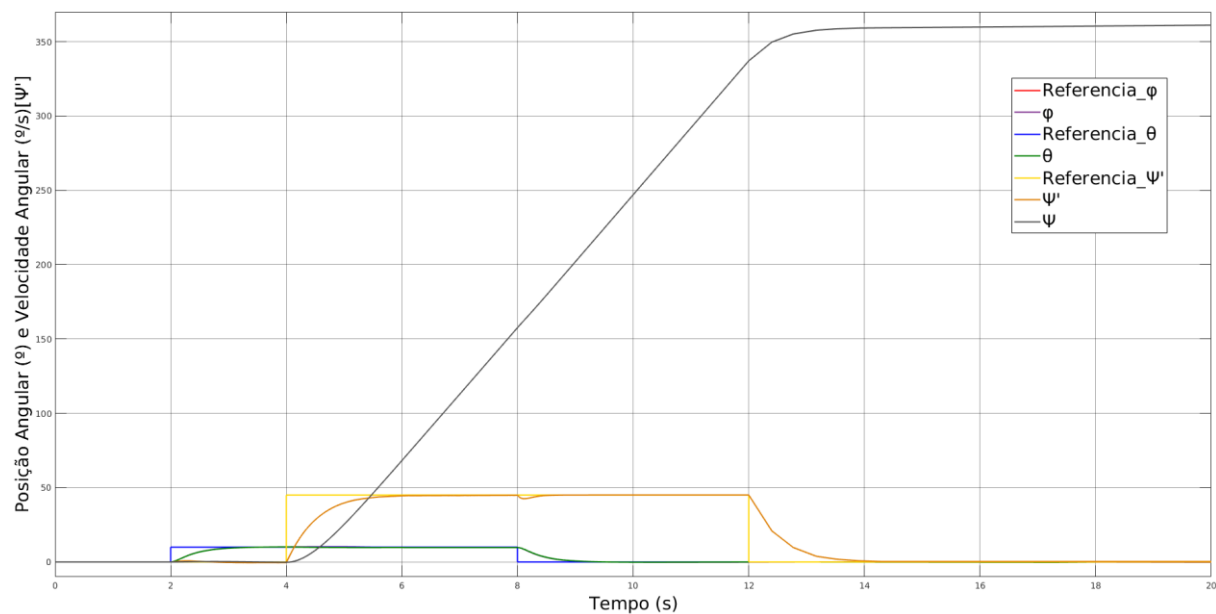
Fonte: Elaboração do próprio autor.

Figura 7.36 – Simulação 4: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear) (zoom em φ e θ).



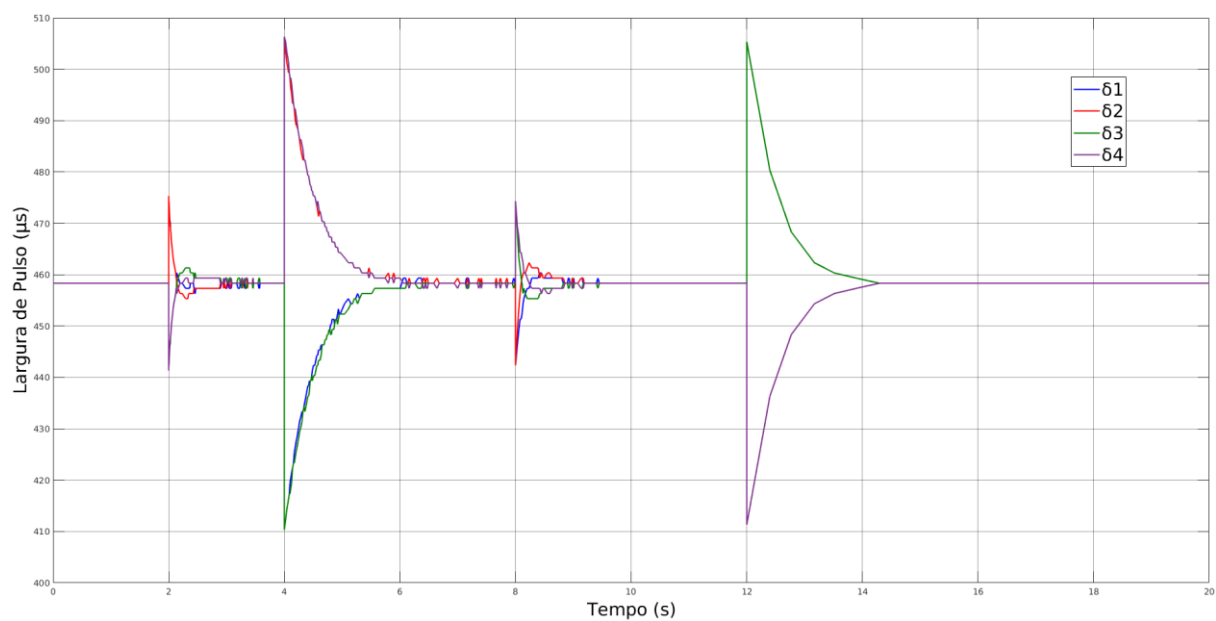
Fonte: Elaboração do próprio autor.

Figura 7.37 – Simulação 4: Controladores Contínuos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



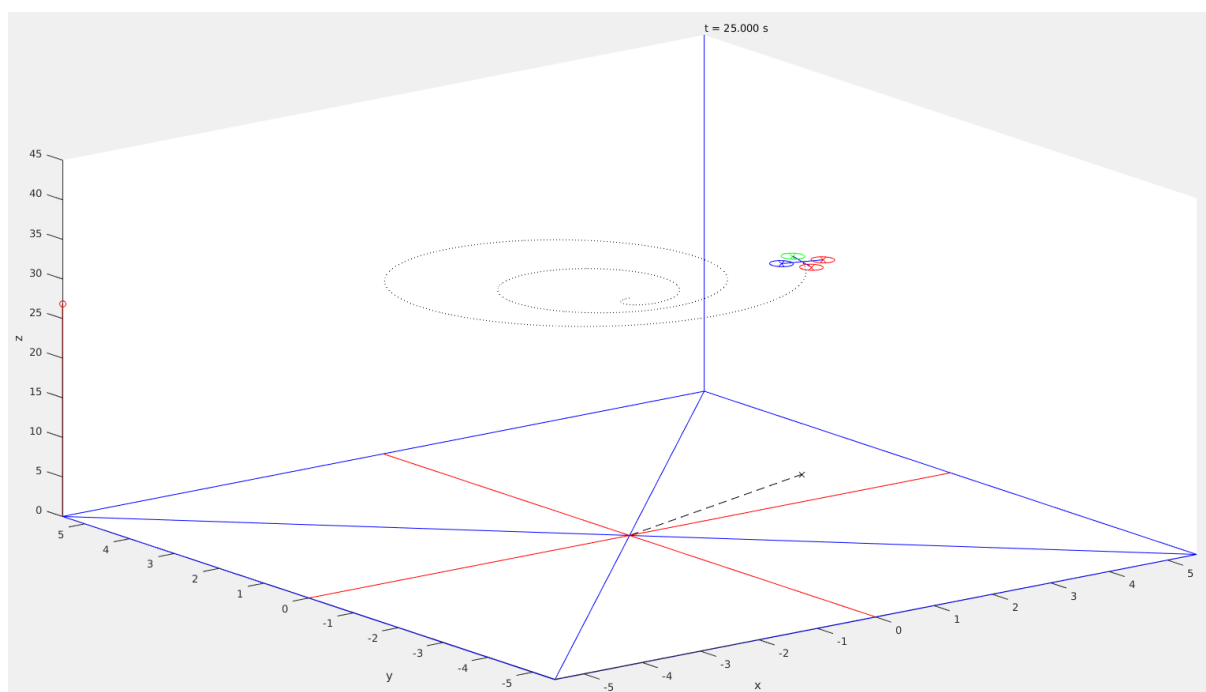
Fonte: Elaboração do próprio autor.

Figura 7.38 – Simulação 4: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos LQRs aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



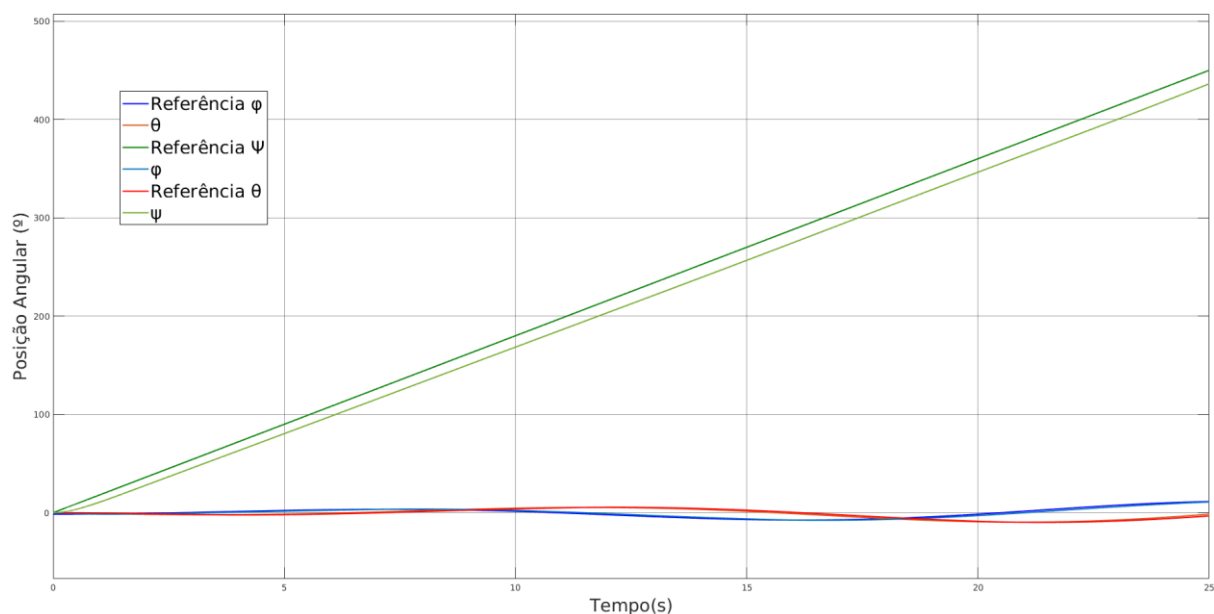
Fonte: Elaboração do próprio autor.

Figura 7.39 – Resultado da simulação de voo utilizando os controladores LQRs e P.



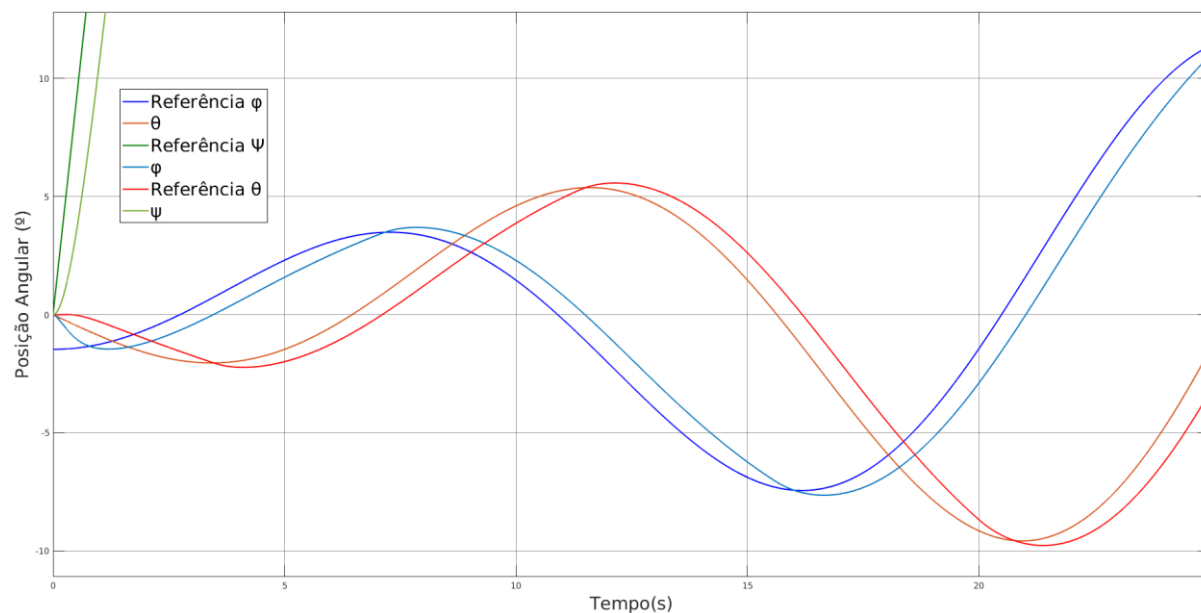
Fonte: Elaboração do próprio autor.

Figura 7.40 – Ângulos de Referência e entrada na simulação de voo utilizando os controladores LQRs e P.



Fonte: Elaboração do próprio autor.

Figura 7.41 – Ângulos de Referência e entrada na simulação de voo utilizando os controladores LQRs e P (zoom).



Fonte: Elaboração do próprio autor.

A Tabela 7.6 apresenta os erros quadráticos médios encontrados nas simulações gerados pelos controladores LQRs aplicados ao controle de ϕ e θ e pelo controlador P aplicado ao controle de ψ' (modelo não linear).

Tabela 7.6 – Erros quadráticos médios obtidos pelos controladores LQRs contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear).

Variável	Simulação 1	Simulação 2	Simulação 3	Simulação 4	Simulação voo
φ	3,4427 °	0,0099 °	3,4348 °	2,8525 °	0,9540 °
θ	0,0173 °	4,0812 °	3,4281 °	2,8691 °	0,8446 °
ψ'	0,0724 °/s	0,0773 °/s	0,5823 °/s	9,4688 °/s	-
ψ	-	-	-	-	11,7488 °

Fonte: Elaboração do próprio autor.

Analizando as saídas das simulações dos controladores LQRs aplicados ao modelo não linear, é possível notar que nas simulações 1 e 2, os ângulos de referência são atingidos em um tempo inferior ao encontrado na simulação linear, porém ainda superiores aos PIDs, e que as diferenças entre as referências e as saídas são menores se comparadas com os controladores PIDs. Para essas duas simulações não há erro de regime permanente, embora haja uma oscilação em torno da referência. Também é possível notar nestas simulações que o controlador P da taxa de guinada sofre mais interferência em conjunto com os controladores LQRs do que com os controladores PIDs. O mesmo ocorre na simulação 3. Nesta simulação, também é possível notar que os três ângulos sofrem forte influência um do outro. Por fim, na simulação 4 é possível notar que a atuação no ângulo de guinada, ψ , também influencia o controle dos demais ângulos, porém menos do que no controlador PID. O resultado da simulação de voo também se mostrou eficiente visto que o quadricóptero traçou a rota programada e os controladores atingiram os ângulos desejados.

Analizando a Tabela 7.6, é possível notar que no modelo não linear, os erros quadráticos médios também podem variar consideravelmente dependendo do tipo de entrada aplicada, assim como ocorrido nos controladores PIDs.

7.1.3 Controle de ψ'

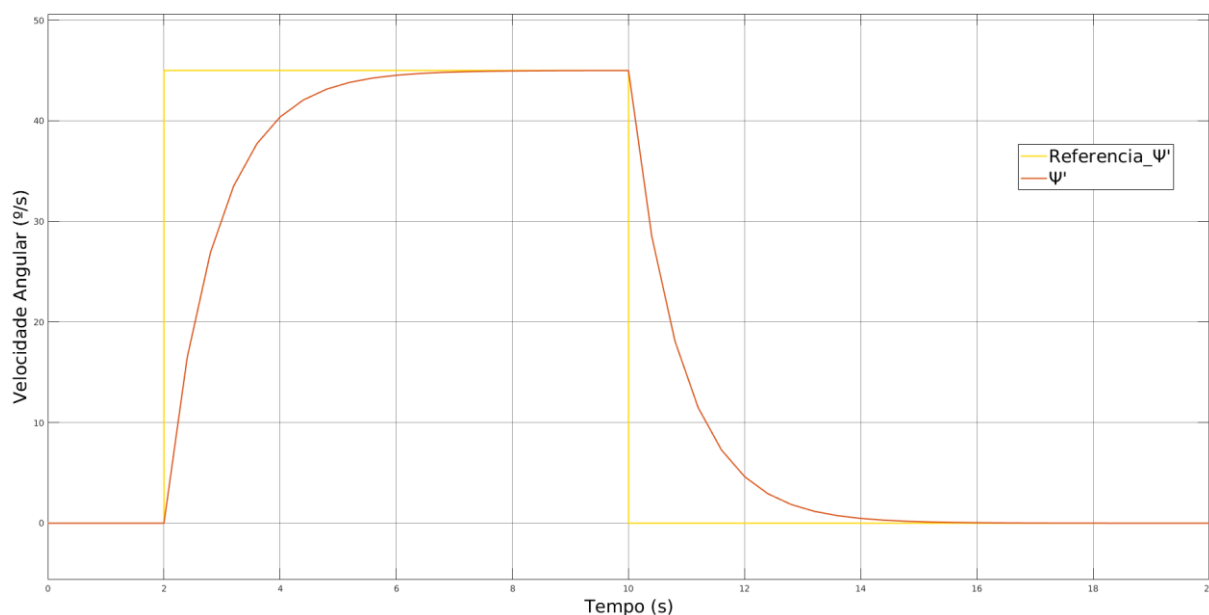
Nesta subseção são exibidos os resultados encontrados nas simulações realizadas com o controlador P aplicado ao modelo linear simplificado. O seu resultado no modelo não linear é exibido em conjunto com cada um dos demais

controladores que fazem seu uso no modelo não linear (subseções 7.1.1, 7.1.2 e 7.1.4).

Para o controlador P aplicado ao modelo linear simplificado, foi utilizado o diagrama de blocos apresentado na Figura 5.10. Dele foram extraídas tanto a velocidade angular ψ' e a posição angular ψ quanto os comandos aplicados aos motores: δ_1 a δ_4 . A entrada do sistema se consiste de um degrau de amplitude 45° , com subida no tempo $t = 2s$ e descida no tempo $t = 10s$. Esta amplitude e duração do degrau foram aplicadas para que quadrirrotor completasse um ciclo de 360° em torno do eixo de guinada.

Os resultados encontrados para o ângulo ψ' e ψ podem ser vistos nas Figuras 7.42 a 7.44.

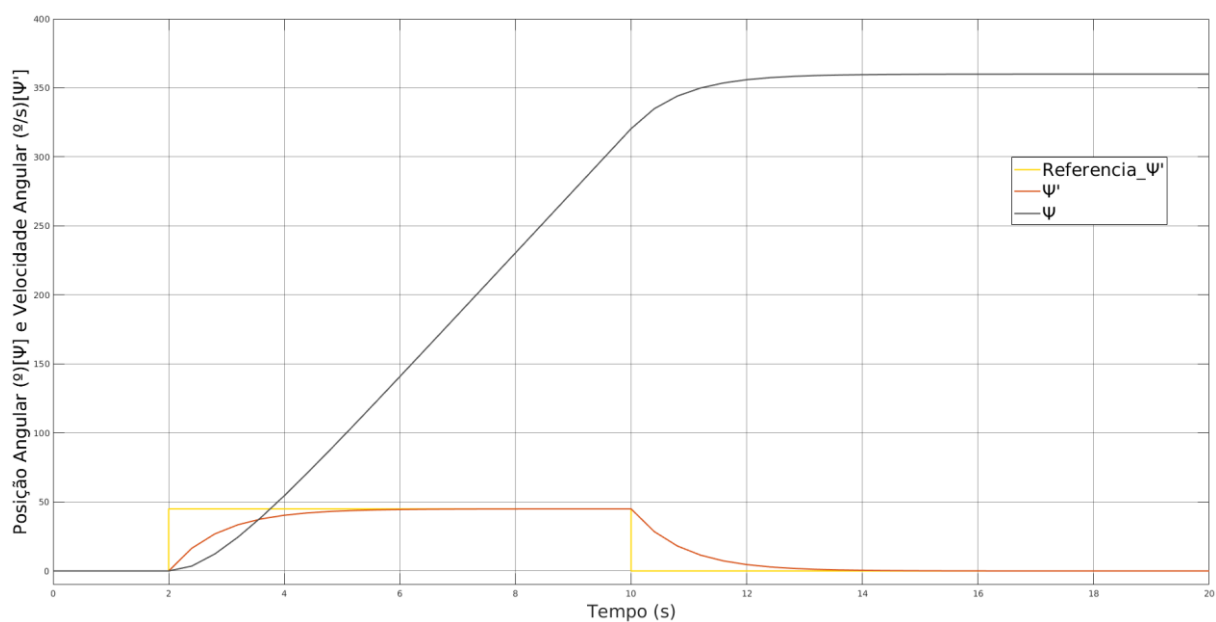
Figura 7.42 – Controlador P Contínuo aplicado ao controle de ψ' (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

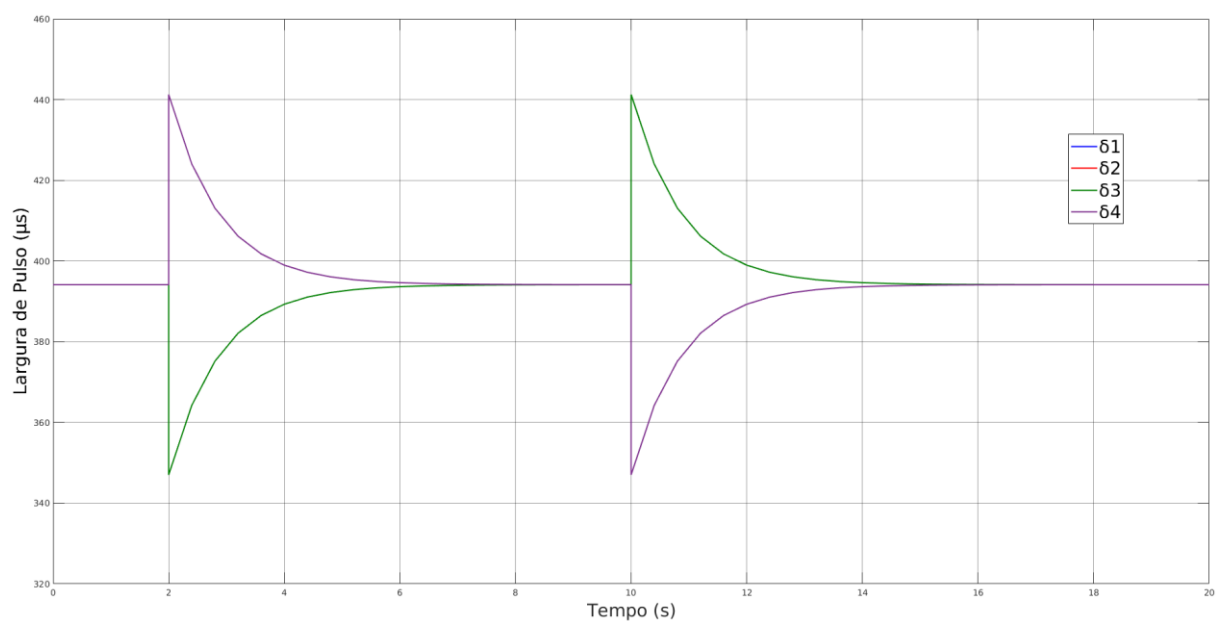
A tabela contendo o valor do ganho do controlador que foi sintonizado na seção 5 pode ser vistos na Tabela 7.7.

Figura 7.43 – Controlador P Contínuo aplicado ao controle de ψ' mais ψ (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

Figura 7.44 – Saídas δ_1 a δ_4 do Controlador P Contínuo aplicado ao controle de ψ' (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

Tabela 7.7 – Ganhos do Controlador P Contínuo.

Ganhos do Controlador P	
k_{pz0}	9.525,6

Fonte: Elaboração do próprio autor.

A Tabela 7.8 apresenta o erro quadrático médio gerado pelo controlador P aplicado ao modelo linear simplificado durante a simulação.

Tabela 7.8 – Erro quadrático médio obtido pelo controlador P contínuo aplicado a ψ' (modelo linear simplificado).

Variável	Simulação 1
ψ'	12,7666 %/s

Fonte: Elaboração do próprio autor.

Através da análise dos resultados obtidos, é possível notar que o controlador P desenvolvido aplicado ao controle de ψ' foi capaz de atingir a referência no ângulo controlado no modelo linear simplificado. Porém, analisando o conjunto de saídas para esta variável de controle em conjunto com os demais controladores nas simulações aplicadas ao modelo não linear, é possível notar que nem sempre este controlador é capaz de atingir a referência, fazendo que o que quadrirrotor continue a rotacionar em torno do eixo de guinada mesmo quando a referência de sua velocidade angular está configurada para zero.

Já o valor encontrado na Tabela 7.8, que é mais elevado do que o encontrado para os controladores aplicados aos ângulos φ e θ , se justifica pelo fato de a amplitude da referência aplicada ser maior e ao fato de a dinâmica para resposta neste eixo ser mais lenta por limitações do sistema.

7.1.4 Backstepping

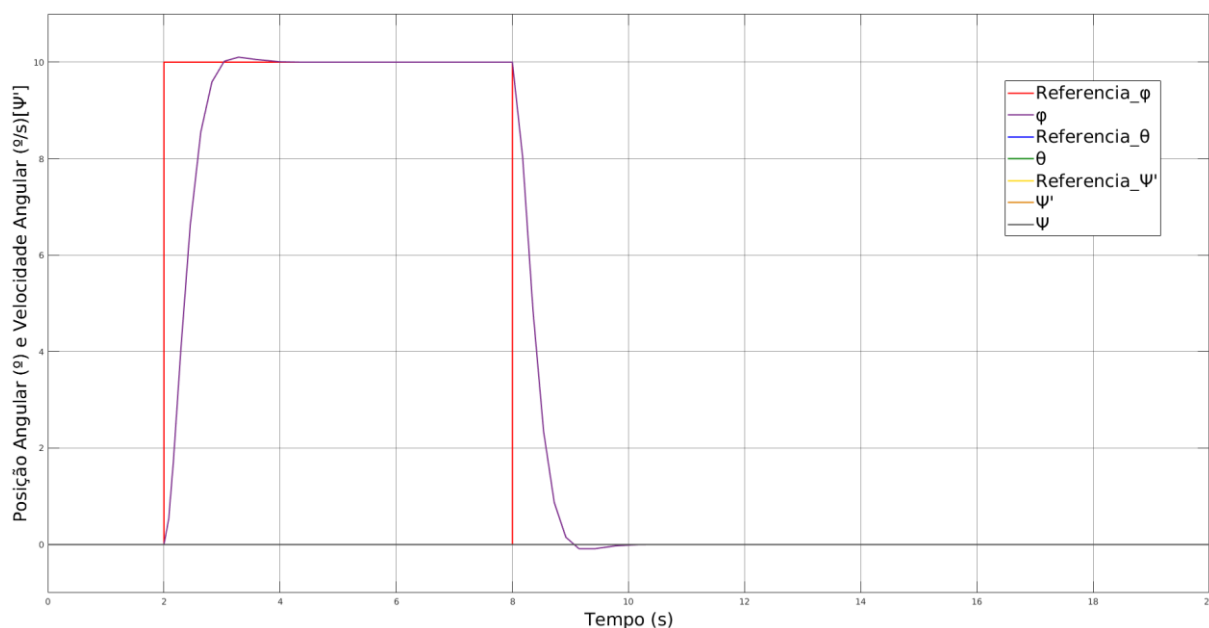
Os resultados encontrados nas simulações realizadas com os controladores *Backsteppings* podem ser vistos nas subseções seguintes.

7.1.4.1 Modelo Linear Simplificado

Para os controladores *Backsteppings* aplicados ao modelo linear simplificado, foi utilizado o diagrama de blocos apresentado na Figura 5.13. Dele foram extraídas as posições angulares φ e θ , a velocidade angular ψ' e os comandos aplicados aos motores: δ_1 a δ_4 . As entradas e simulações aplicadas ao sistema foram as mesmas aplicadas nos controladores PIDs e a referência de ψ' foi mantida em zero.

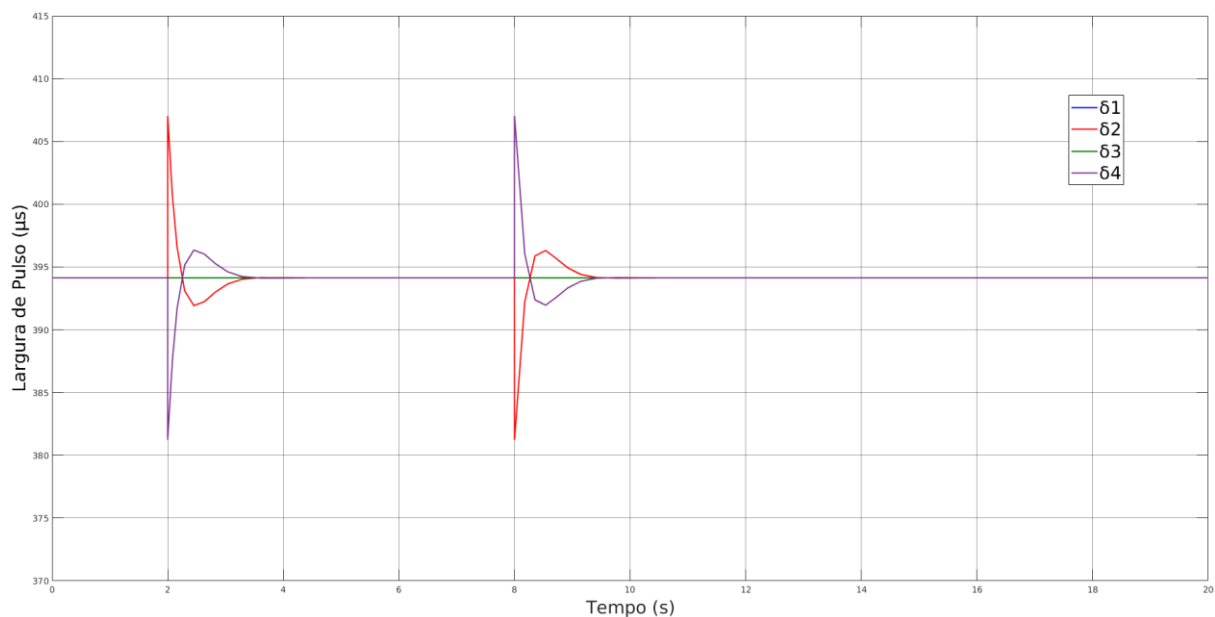
Os resultados encontrados para φ e θ e ψ' nas duas simulações podem ser vistos nas Figuras 7.45 e 7.47. Já os comandos aplicados aos motores nas duas simulações podem ser vistos nas Figuras 7.46 e 7.48. A tabela contendo os valores dos ganhos dos controladores que foram sintonizados na seção 5 pode ser vista na Tabela 7.9.

Figura 7.45 – Simulação 1: Controladores Contínuos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo linear simplificado).



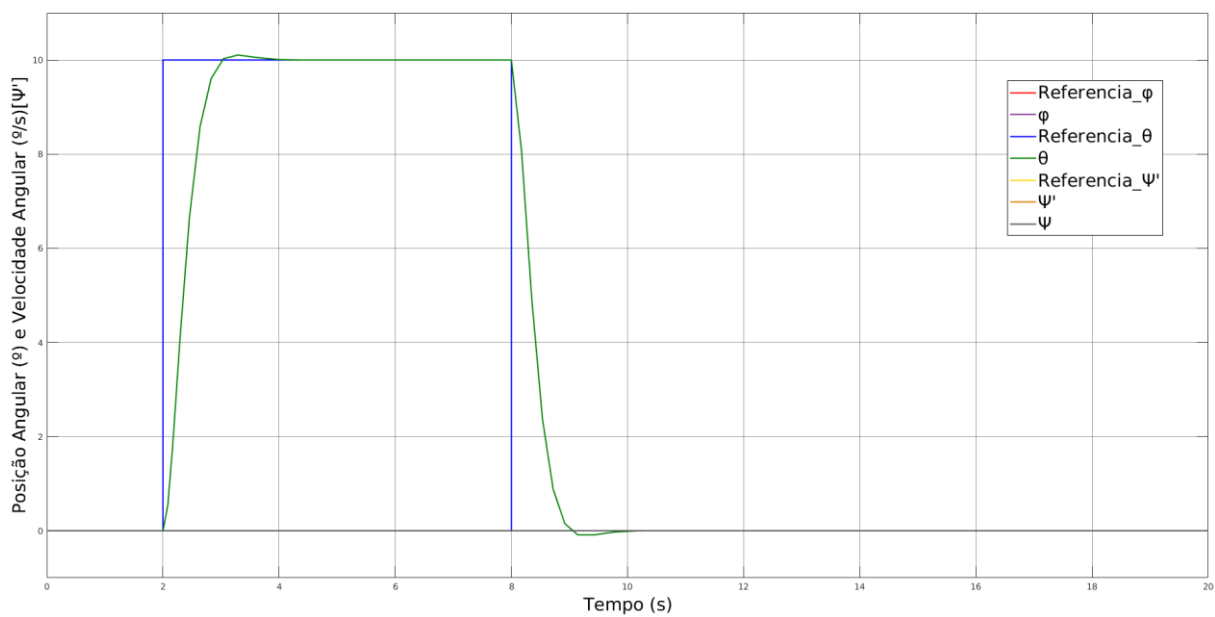
Fonte: Elaboração do próprio autor.

Figura 7.46 – Simulação 1: Saídas δ_1 a δ_4 dos Controladores *Backsteppings* Contínuos aplicados a φ e θ e P aplicado a ψ' (modelo linear simplificado).



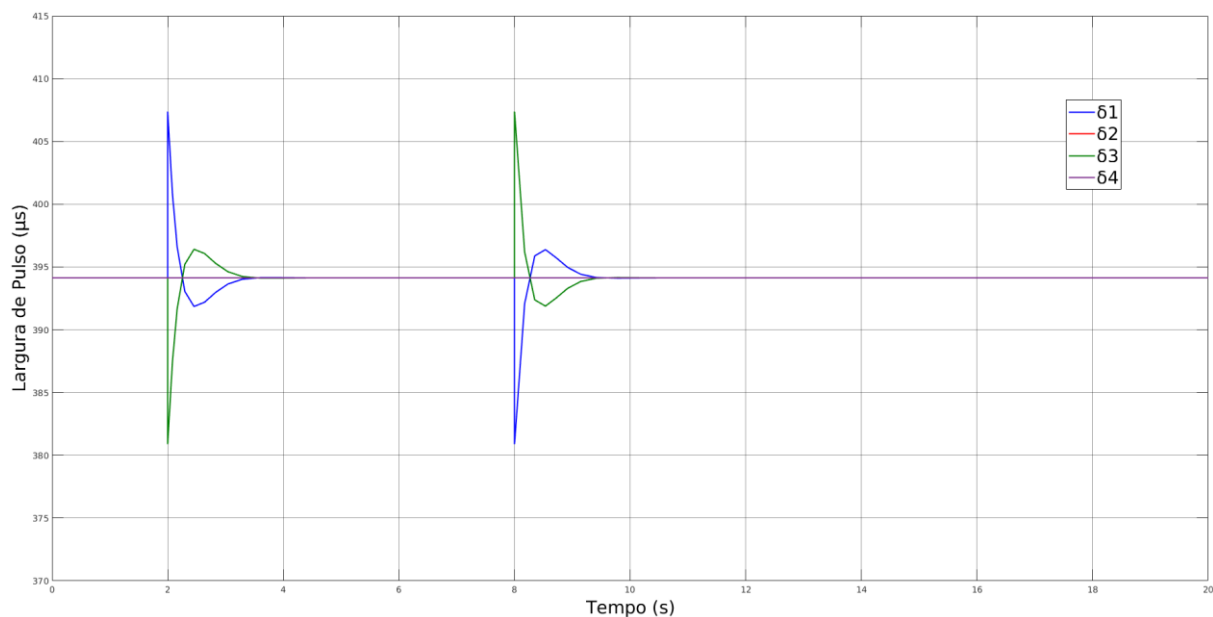
Fonte: Elaboração do próprio autor.

Figura 7.47 – Simulação 2: Controladores Contínuos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

Figura 7.48 – Simulação 2: Saídas δ_1 a δ_4 dos Controladores *Backsteppings* Contínuos aplicados a φ e θ e P aplicado a ψ' (modelo linear simplificado).



Fonte: Elaboração do próprio autor.

Tabela 7.9 – Ganhos dos controladores *Backstepping* contínuos.

Ganhos Controlador <i>Backstepping</i> de φ		Ganhos Controlador <i>Backstepping</i> de θ	
b_{1_0}	159,3798	b_{3_0}	166,5341
b_{1_1}	-6,7824	b_{3_1}	-7,1948
b_{2_0}	59,7973	b_{4_0}	62,4400
b_{2_1}	-2,5465	b_{4_1}	-2,6975
c_1	-0,765	c_2	0,946

Fonte: Elaboração do próprio autor.

A Tabela 7.10 apresenta os erros quadráticos médios obtidos durante as simulações pelos controladores *Backsteppings* e P aplicados ao modelo linear simplificado.

Tabela 7.10 – Erros quadráticos médios obtidos pelos controladores contínuos *Backsteppings* aplicados a φ e θ e P aplicado a ψ' (modelo linear simplificado).

Variável	Simulação 1	Simulação 2
φ	3,2840 °	0,0 °
θ	0,0 °	3,3104 °
ψ'	0 %/s	0 %/s

Fonte: Elaboração do próprio autor.

Através da análise dos resultados obtidos, é possível notar que os controladores *Backsteppings* desenvolvidos foram capazes de atingir a referência nos dois ângulos controlados. Os erros quadráticos médios obtidos nas duas simulações apresentados na Tabela 7.10 também mostram que os desempenhos dos controladores *Backsteppings* aplicados ao controle dos dois ângulos são bem próximos. Também é possível notar que para ambas as simulações a saída apresentou um sobressinal pequeno e inferior ao encontrado pelos controladores PIDs.

7.1.4.2 Modelo Não Linear

Para a realização da simulação no modelo não linear, foi utilizado o diagrama de blocos mostrado na Figura 7.9. Seu conteúdo para o uso dos controladores *Backsteppings* pode ser visto na Figura 7.49.

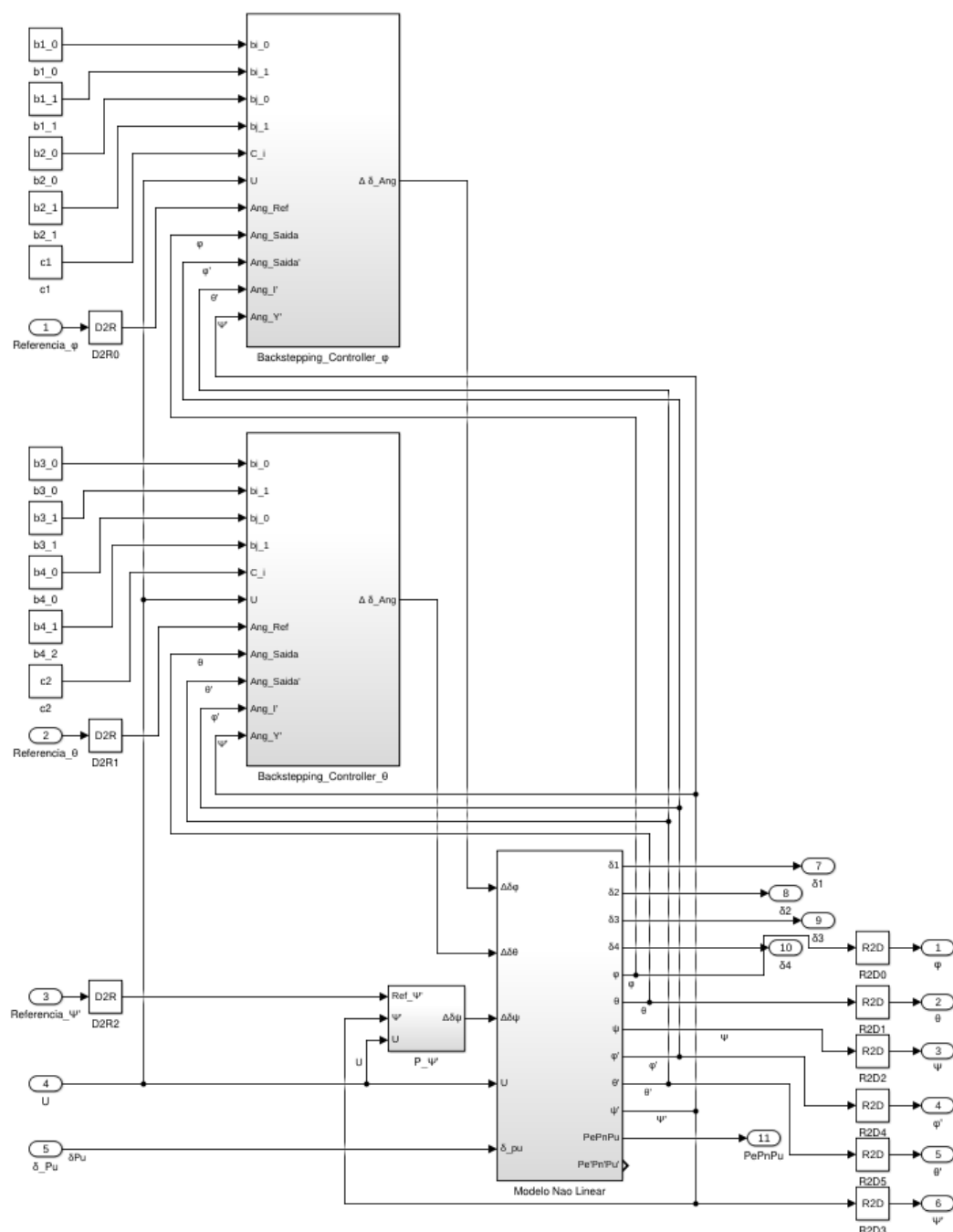
Deste modelo de simulação, foram extraídas as posições angulares φ , θ e ψ , a velocidade angular ψ' e os comandos aplicados aos motores: δ_1 a δ_4 . Os mesmos quatro tipos de simulações aplicados aos controladores PIDs foram aplicados a estes controladores.

Os resultados destas simulações podem ser vistos nas Figuras 7.50 a 7.58.

Por fim, também foi realizada uma simulação de voo utilizando o mesmo planejador de trajetória utilizado com os controladores PIDs. O resultado da simulação de voo pode ser vista na Figura 7.59.

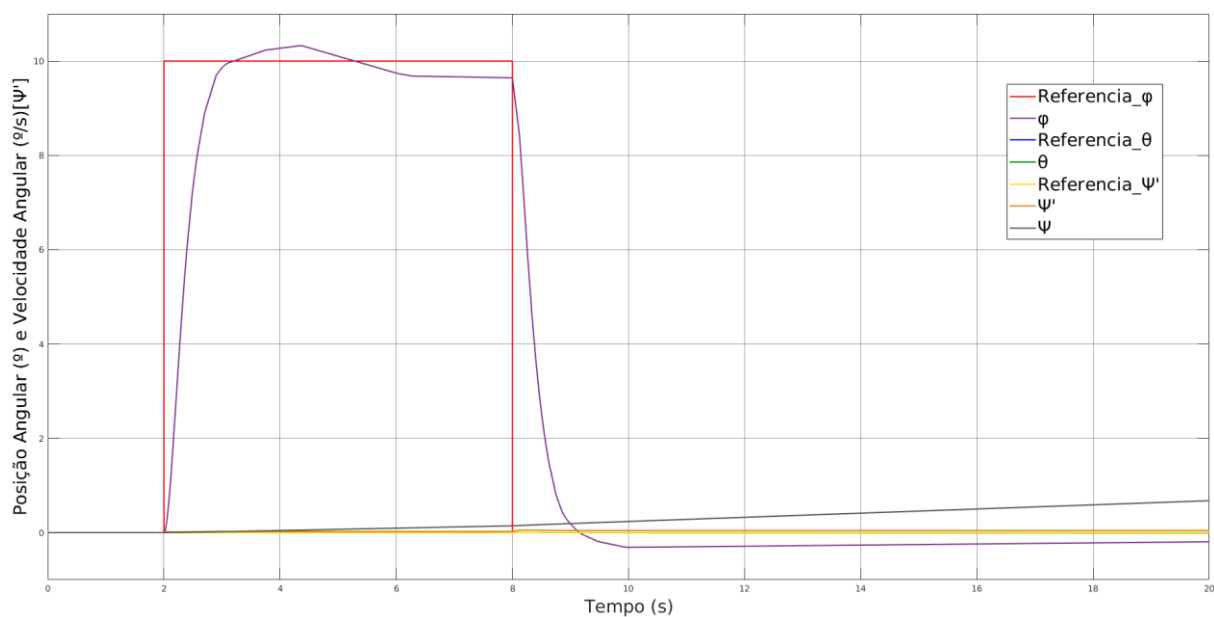
O comparativo entre os ângulos de referência e os ângulos alcançados pode ser visto na Figura 7.60. Na Figura 7.61 é dada uma ampliação na parte inferior da curva para facilitar a visualização dos ângulos φ e θ .

Figura 7.49 – Diagrama de Blocos utilizados para simulação dos controladores *Backstepping* aplicados ao modelo não linear.



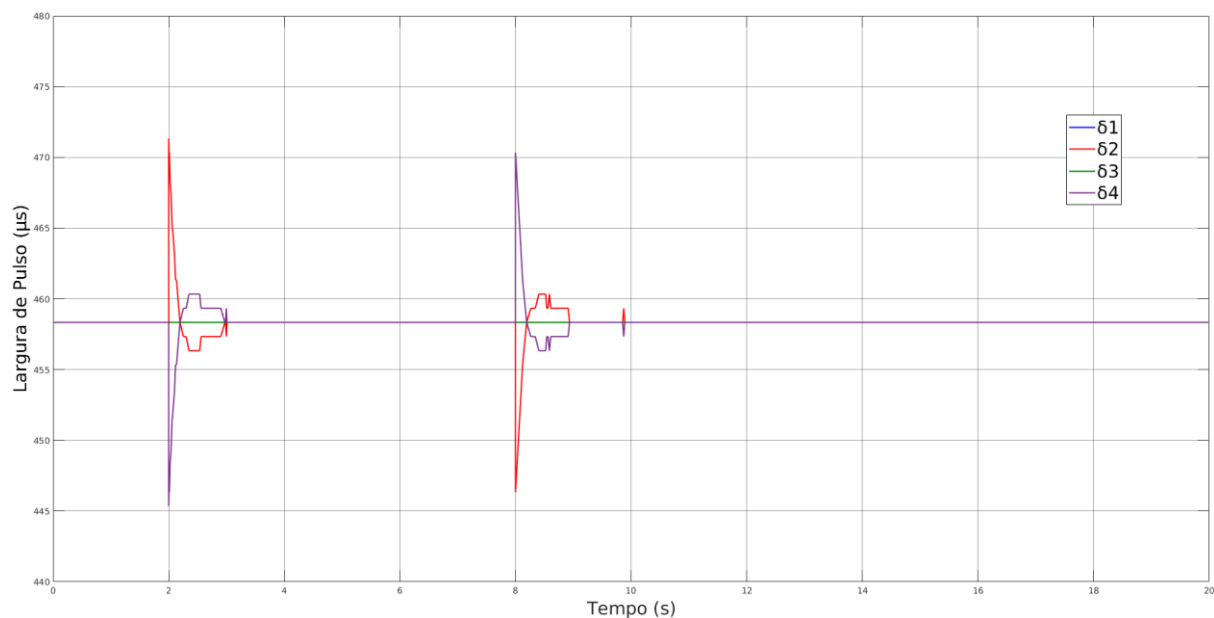
Fonte: Elaboração do próprio autor.

Figura 7.50 – Simulação 1: Controladores Contínuos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



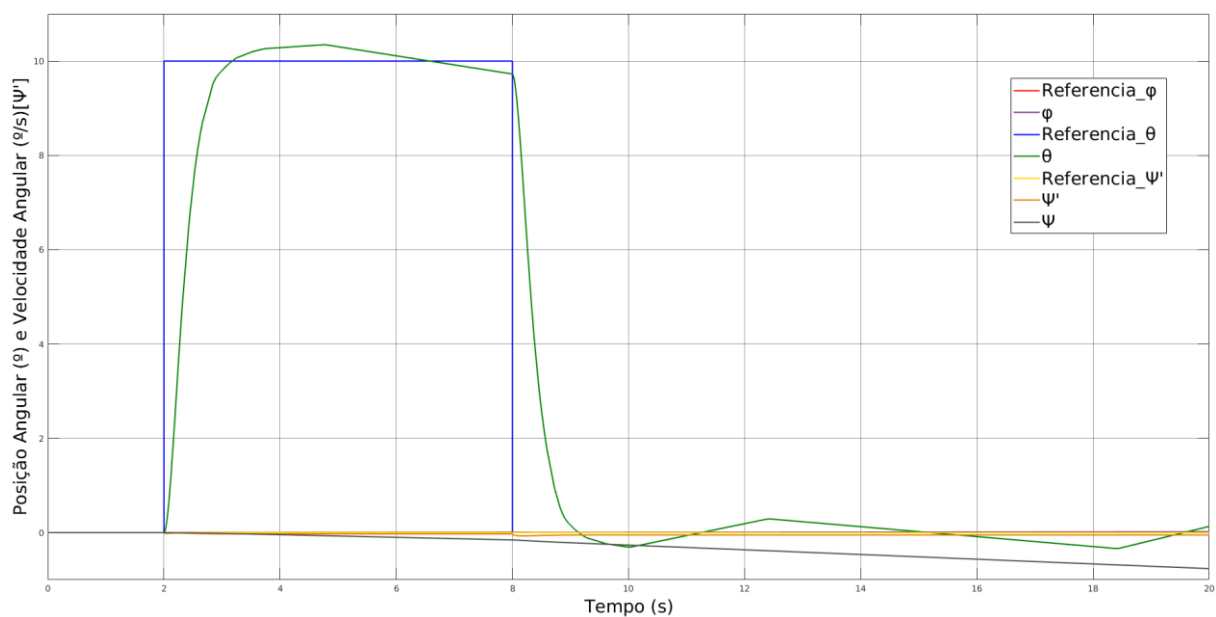
Fonte: Elaboração do próprio autor.

Figura 7.51 – Simulação 1: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos *Backsteppings* aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



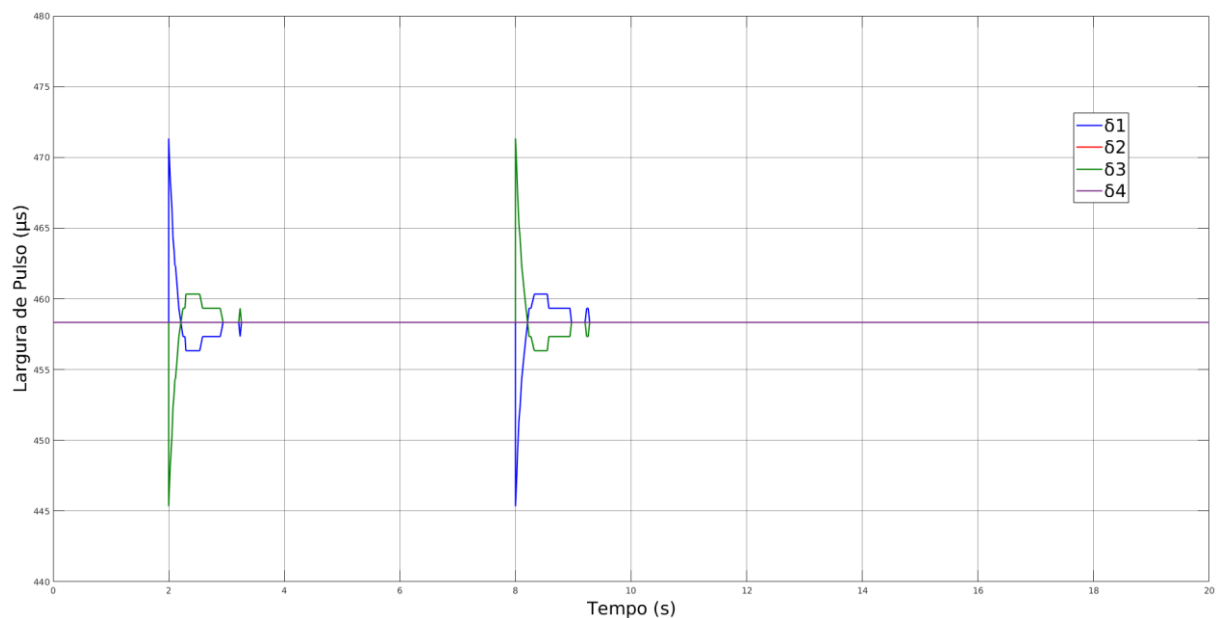
Fonte: Elaboração do próprio autor.

Figura 7.52 – Simulação 2: Controladores Contínuos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



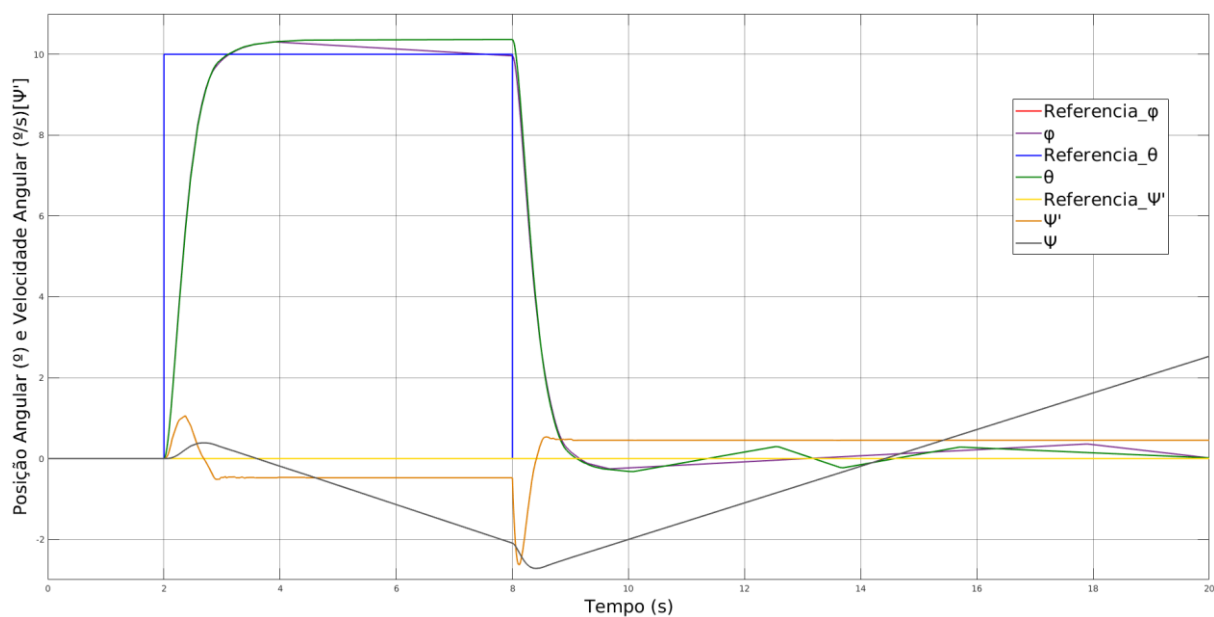
Fonte: Elaboração do próprio autor.

Figura 7.53 – Simulação 2: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos *Backsteppings* aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



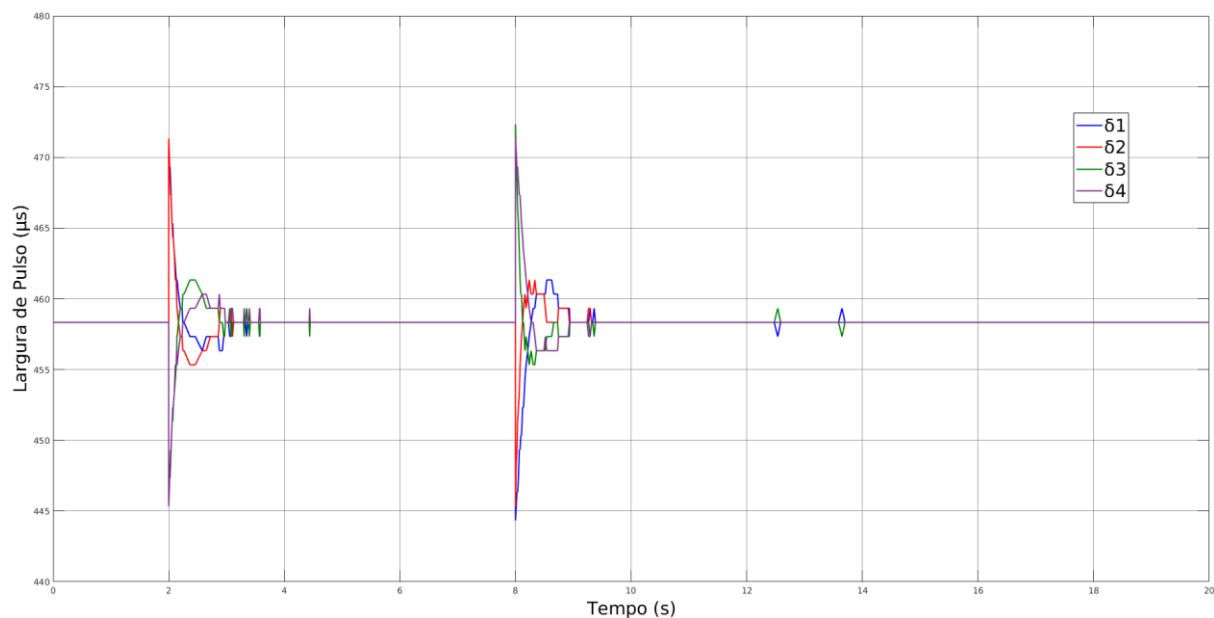
Fonte: Elaboração do próprio autor.

Figura 7.54 – Simulação 3: Controladores Contínuos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



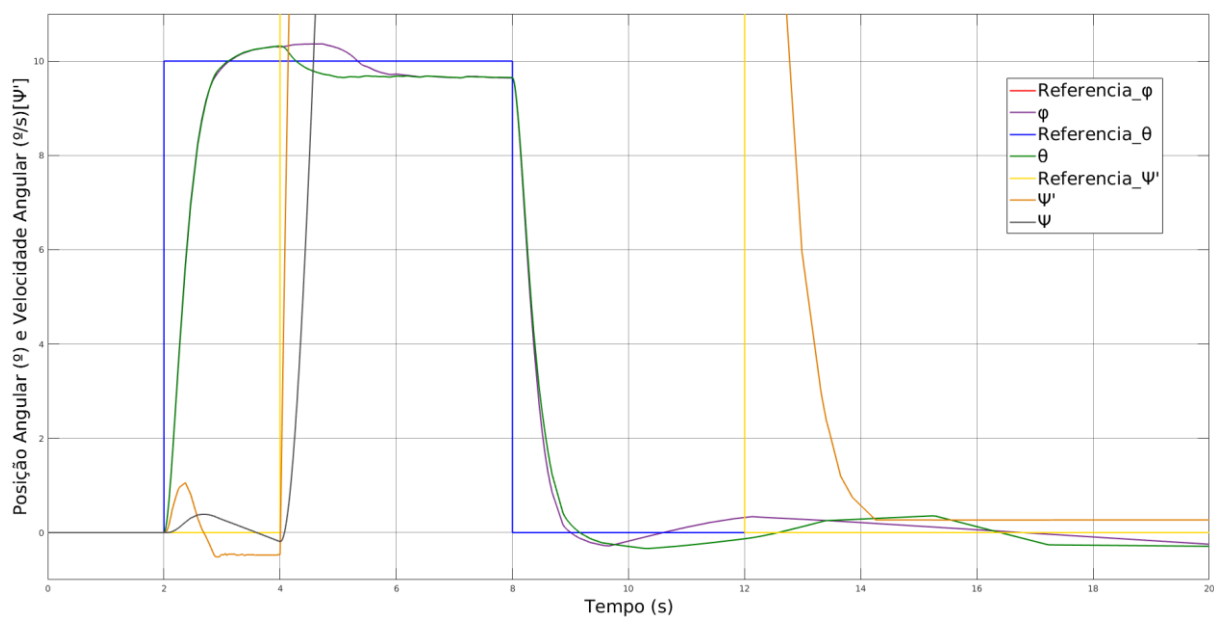
Fonte: Elaboração do próprio autor.

Figura 7.55 – Simulação 3: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos *Backsteppings* aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



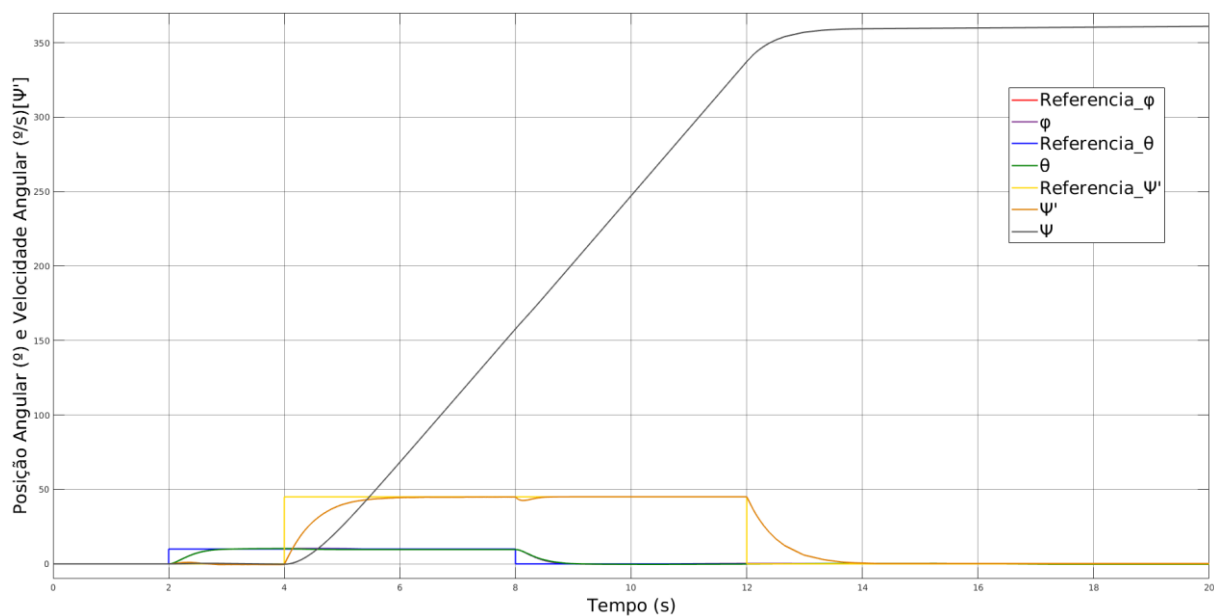
Fonte: Elaboração do próprio autor.

Figura 7.56 – Simulação 4: Controladores Contínuos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear) (zoom).



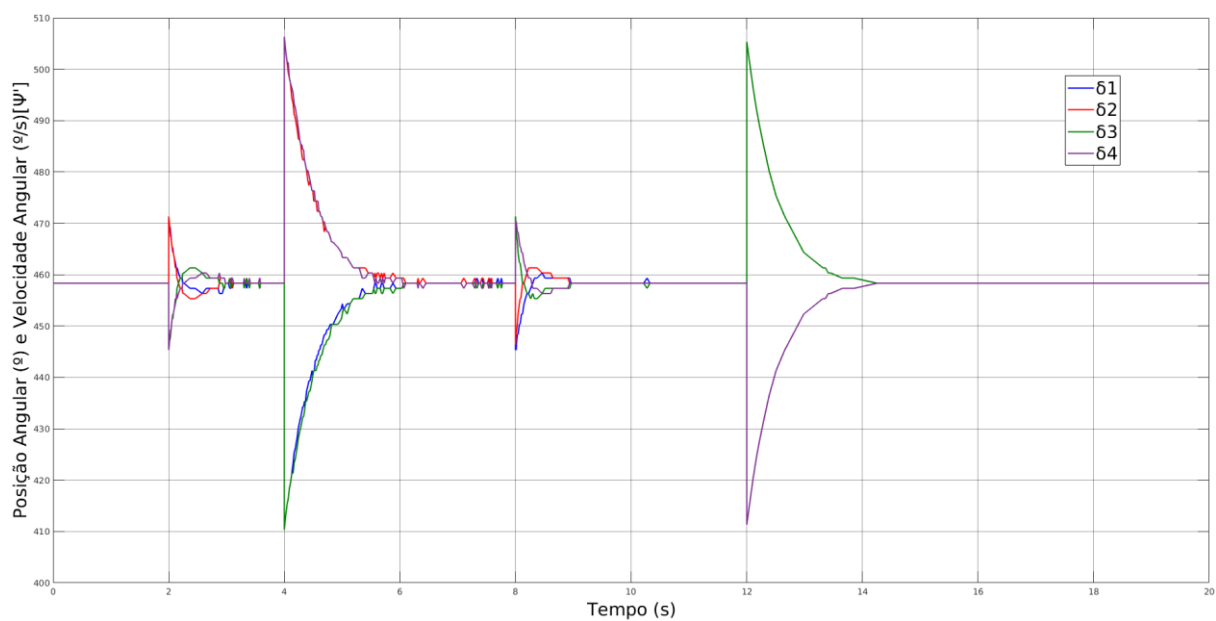
Fonte: Elaboração do próprio autor.

Figura 7.57 – Simulação 4: Controladores Contínuos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' (modelo não linear).



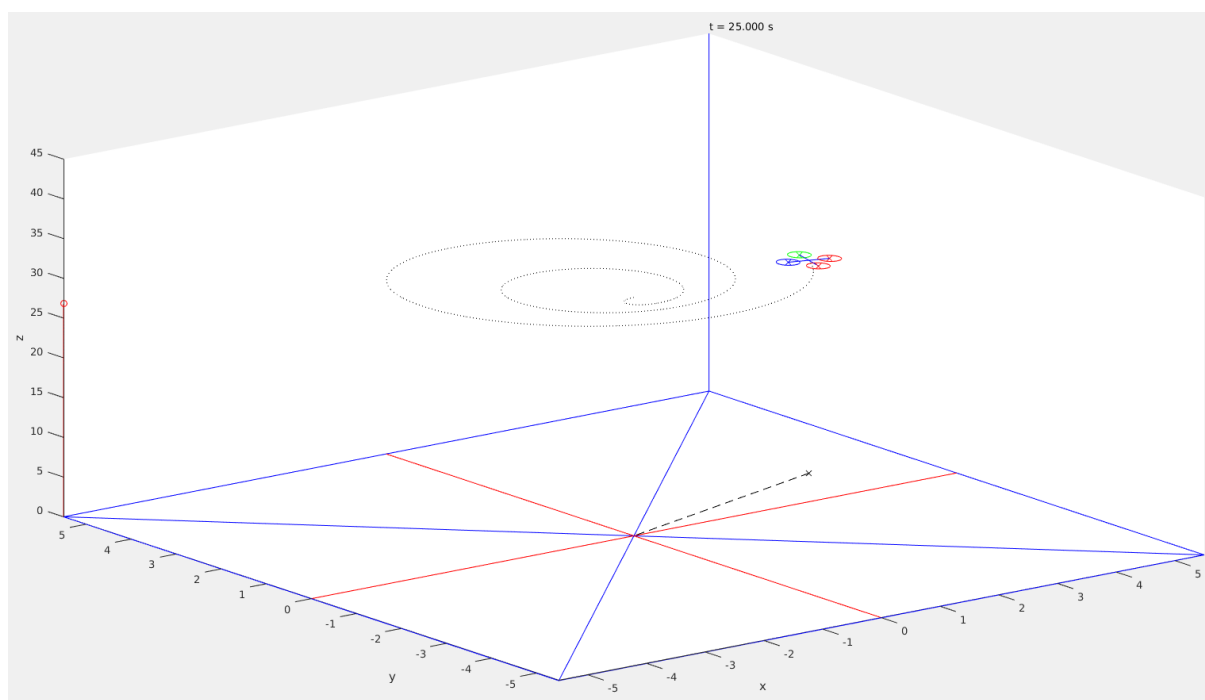
Fonte: Elaboração do próprio autor.

Figura 7.58 – Simulação 4: Comandos δ_1 a δ_4 gerados pelos Controladores Contínuos *Backsteppings* aplicados a φ e θ e P aplicado a ψ' (modelo não linear).



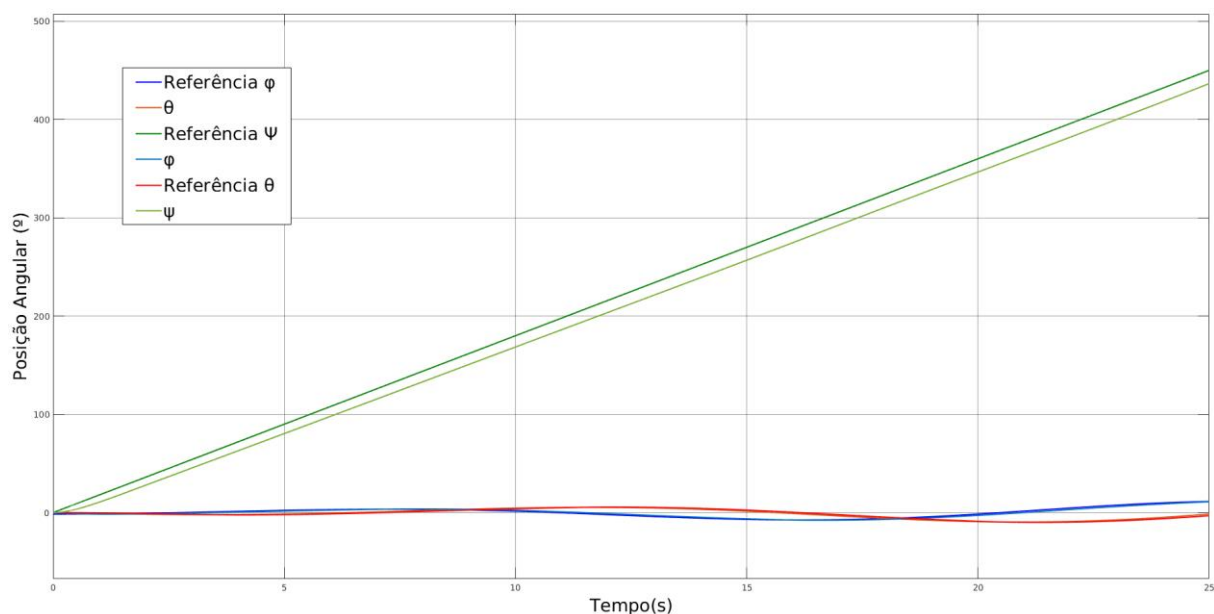
Fonte: Elaboração do próprio autor.

Figura 7.59 – Resultado da simulação de voo utilizando os controladores *Backsteppings* e P.



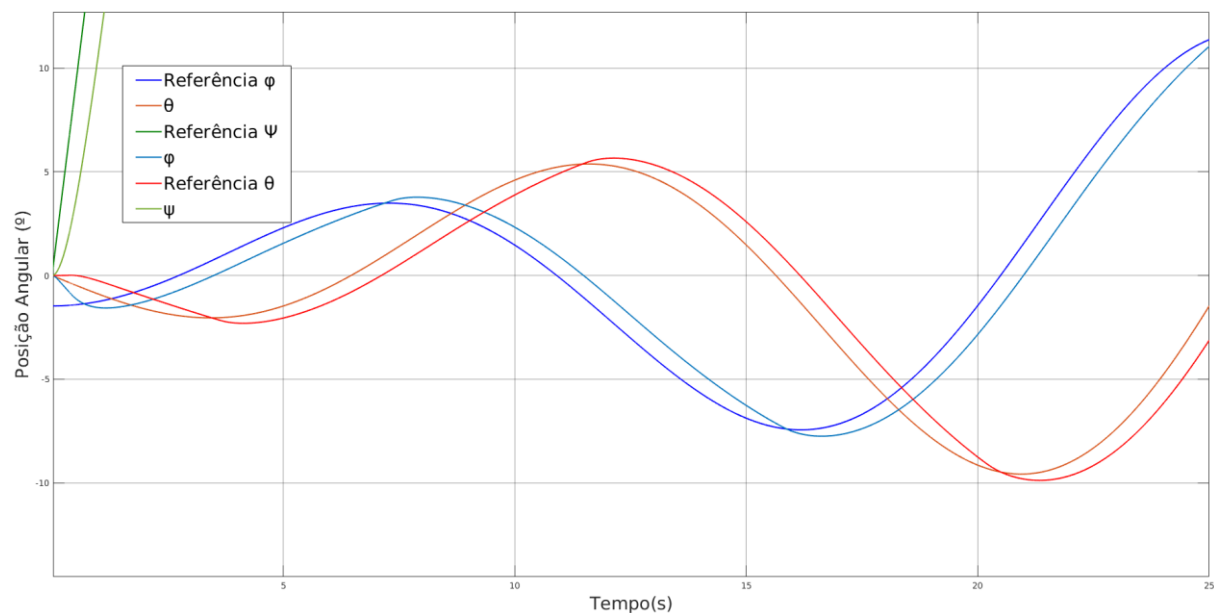
Fonte: Elaboração do próprio autor.

Figura 7.60 – Ângulos de Referência e entrada na simulação de voo utilizando os Controladores Contínuos *Backstepping* e P.



Fonte: Elaboração do próprio autor.

Figura 7.61 – Ângulos de Referência e entrada na simulação de voo utilizando os Controladores Contínuos *Backstepping* e P (zoom).



Fonte: Elaboração do próprio autor.

A Tabela 7.11 apresenta os erros quadráticos médios encontrados nas simulações gerados pelos controladores *Backsteppings* aplicados ao controle de φ e θ e pelo controlador P aplicado ao controle de ψ' (modelo não linear).

Tabela 7.11 – Erros quadráticos médios obtidos pelos controladores *Backsteppings* contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear).

Variável	Simulação 1	Simulação 2	Simulação 3	Simulação 4	Simulação voo
φ	3,4625 °	0,0080 °	3,7942 °	3,0097 °	0,9292 °
θ	0,0026 °	3,9683 °	3,8675 °	3,0271 °	0,8274 °
ψ'	0,0356 °/s	0,0422 °/s	0,7035 °/s	10,4093 °/s	-
ψ	-	-	-	-	11,6400 °

Fonte: Elaboração do próprio autor.

Analisando as saídas das simulações dos controladores *Backsteppings* aplicados ao modelo não linear, é possível detectar que nas simulações 1 e 2, os ângulos de referência são atingidos em um tempo muito similar ao encontrado na simulação linear, superiores aos controladores PIDs e inferiores aos LQRs, e que a diferença entre a referência e saída é menor se comparada com o controladores PIDs. Para o caso do ângulo φ também houve um erro de regime permanente. Já na simulação 3, é possível notar que o ângulo θ sofreu com problema de regime permanente enquanto estava em nível alto, mas não quando retornou para a referência zero. É possível também notar que os três ângulos sofrem influência um do outro, inclusive o eixo de guinada. Por fim, na simulação 4 é possível notar que a atuação no ângulo de guinada, ψ , também influencia o controle dos demais ângulos, porém menos do que no controlador PID. O resultado da simulação de voo também se mostrou eficiente visto que o quadricóptero traçou a rota programada e os controladores atingiram os ângulos desejados.

7.2 Controladores Discretos

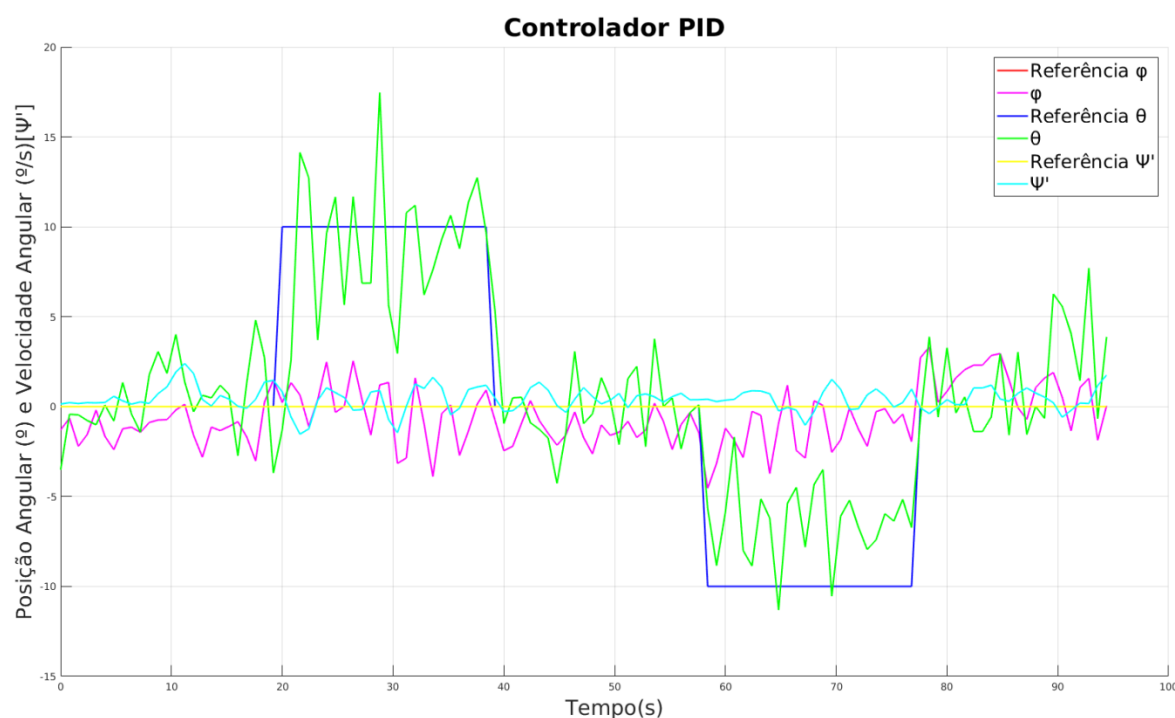
Para os controladores discretos, outro tipo de entrada foi utilizado. Conforme visto na seção 6, apenas o ângulo de arfagem é livre e não acopla a inércia da estrutura giroscópica. Desta forma, apenas a referência deste ângulo foi variada ao longo do tempo e os demais foram mantidos na referência zero. A entrada aplicada ao ângulo θ possui 90 segundos de duração, se mantendo 20 segundos em nível zero, seguido por 20 segundos em amplitude de 10°, por mais 20 segundos em nível zero e depois por mais 20 segundos em amplitude de -10°. Por fim o sinal é retornado para o nível zero por mais 10 segundos. As saídas geradas para cada um

dos controladores desenvolvidos e implementados em software podem ser vistas nas subseções seguintes.

7.2.1 PID

Os resultados obtidos para o controle dos ângulos φ , θ utilizando os controladores PIDs e pelo controlador proporcional aplicado a ψ' podem ser vistos nas Figuras 7.62 e 7.63. O ângulo ψ apresentado na Figura 7.63 é uma estimativa a partir de ψ' medido.

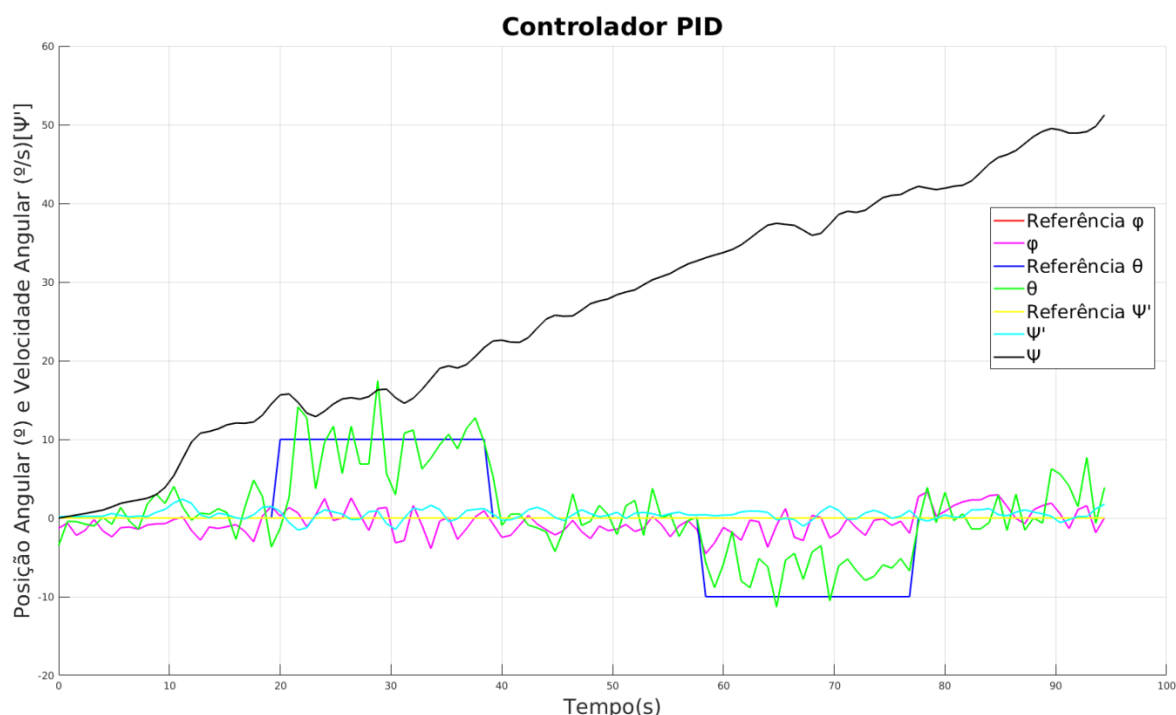
Figura 7.62 – Controladores Discretos PIDs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' .



Fonte: Elaboração do próprio autor.

A Tabela 7.12 apresenta os erros quadráticos médios encontrados para os controladores discretos PIDs aplicados ao controle de φ e θ e para o controlador P aplicado ao controle de ψ' .

Figura 7.63 – Controladores Discretos PID's aplicados ao controle de φ e θ e P aplicado ao controle de ψ' mais ψ estimado.



Fonte: Elaboração do próprio autor.

Tabela 7.12 – Erros quadráticos médios para os controladores discretos PID's aplicados a φ e θ e P aplicado a ψ' .

Ângulo	Erro Quadrático Médio
φ	1,7012 °
θ	3,2908 °
ψ'	0,7852 °/s

Fonte: Elaboração do próprio autor.

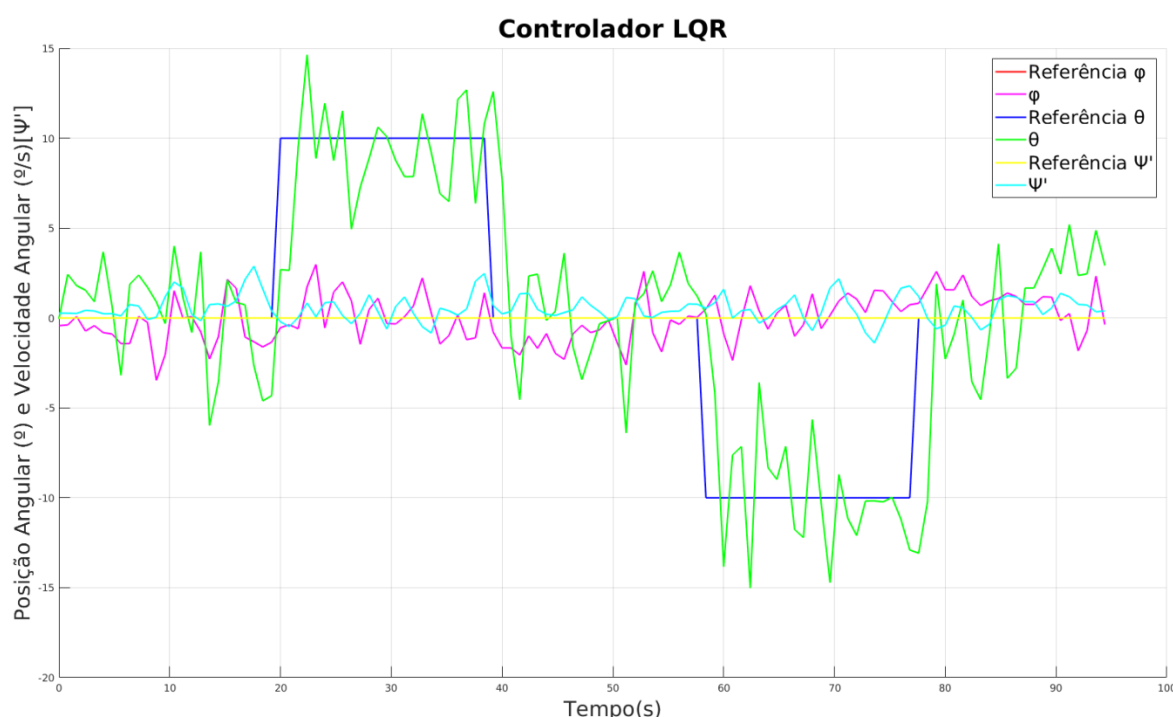
A partir dos resultados obtidos pelos controladores discretos PID's em conjunto com o controlador P aplicado ao modelo real fixado à estrutura giroscópica, é possível notar um comportamento oscilatório em torno da referência em todos os ângulos, principalmente no ângulo θ . Os motivos destas oscilações e dificuldades para atingirem as referências são discutidos na seção de conclusão.

Também é possível notar que a estimativa do ângulo ψ é crescente, embora durante os ensaios isso não tenha ocorrido. Isso se deve ao fato de erros no sensor giroscópico utilizado para realimentar a malha de controle neste eixo.

7.2.2 LQR

Os resultados obtidos para o controle dos ângulos φ , θ utilizando os controladores LQRs e o controlador proporcional aplicado a ψ' podem ser vistos nas Figuras 7.64 e 7.65. O ângulo ψ apresentado na Figura 7.65 é uma estimativa a partir de ψ' medido.

Figura 7.64 – Controladores discretos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' .



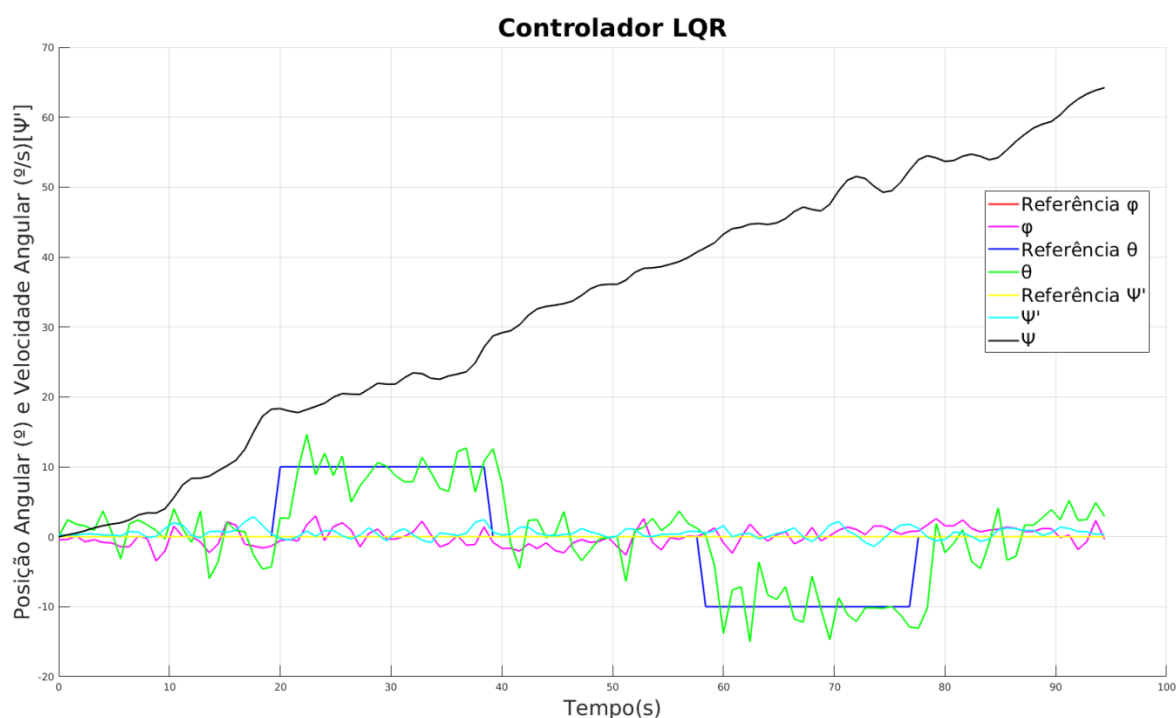
Fonte: Elaboração do próprio autor.

A Tabela 7.13 apresenta os erros quadráticos médios encontrados para os controladores discretos LQRs aplicados ao controle de φ e θ e para o controlador P aplicado ao controle de ψ' .

A partir dos resultados obtidos pelos controladores discretos LQRs em conjunto com o controlador P aplicado ao modelo real fixado à estrutura giroscópica, também é possível notar um comportamento oscilatório em torno da referência em todos os ângulos, principalmente no ângulo θ .

Como os sensores utilizados em todos os ensaios foram os mesmos, o mesmo erro descrito no final da subseção anterior na leitura de ψ' ocorreu neste ensaio.

Figura 7.65 – Controladores discretos LQRs aplicados ao controle de φ e θ e P aplicado ao controle de ψ' mais ψ estimado.



Fonte: Elaboração do próprio autor.

Tabela 7.13 – Erros quadráticos médios para os controladores discretos LQRs aplicados a φ e θ e P aplicado a ψ' .

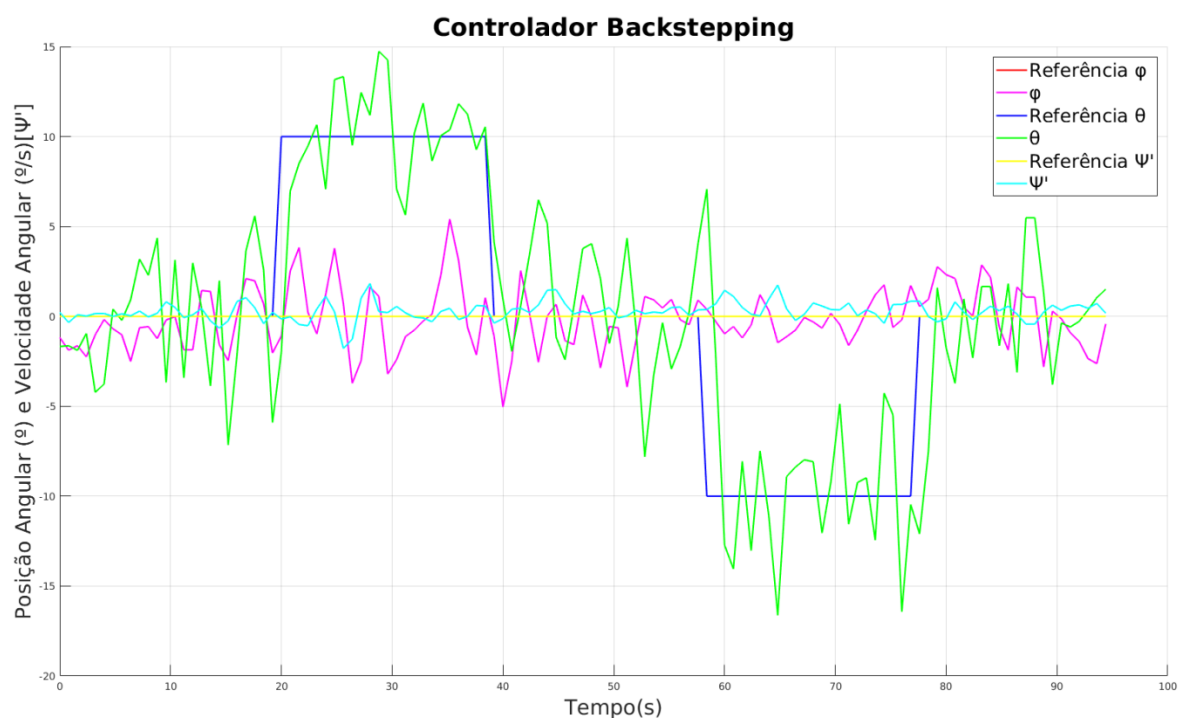
Ângulo	Erro Quadrático Médio
φ	1,2886 °
θ	3,6031 °
ψ'	0,898 °/s

Fonte: Elaboração do próprio autor.

7.2.3 Backstepping

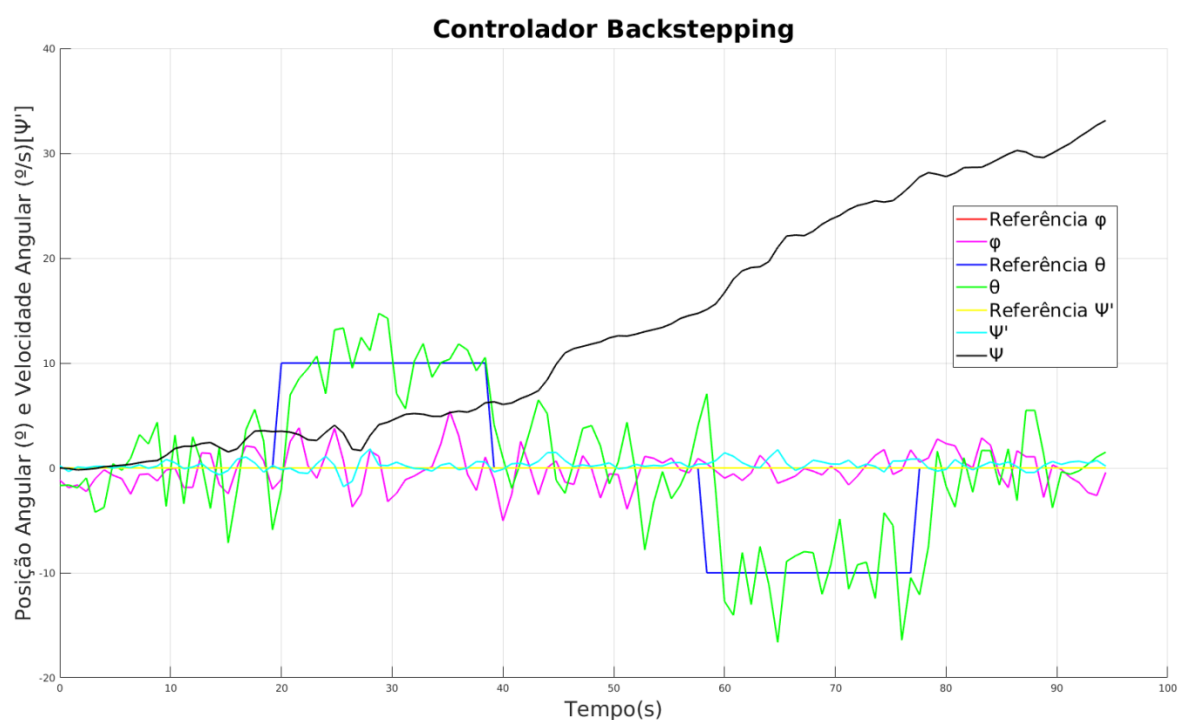
Os resultados obtidos para o controle dos ângulos φ , θ utilizando os controladores *Backsteppings* e controlador proporcional aplicado a ψ' podem ser vistos nas Figuras 7.66 e 7.67. O ângulo ψ apresentado na Figura 7.67 é uma estimativa a partir de ψ' medido.

Figura 7.66 – Controladores discretos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' .



Fonte: Elaboração do próprio autor.

Figura 7.67 – Controladores discretos *Backsteppings* aplicados ao controle de φ e θ e P aplicado ao controle de ψ' mais ψ estimado.



Fonte: Elaboração do próprio autor.

A Tabela 7.14 apresenta os erros quadráticos médios encontrados para os controladores discretos *Backsteppings* aplicados ao controle de φ e θ e para o controlador P aplicado ao controle de ψ' .

Tabela 7.14 – Erros quadráticos médios para os controladores discretos *Backsteppings* aplicados a φ e θ e P aplicado a ψ' .

Ângulo	Erro Quadrático Médio
φ	1,7271 °
θ	3,8239 °
ψ'	0,5856 %/s

Fonte: Elaboração do próprio autor.

A partir dos resultados obtidos pelos controladores discretos *Backsteppings* em conjunto com o controlador P aplicado ao modelo real fixado à estrutura giroscópica, também é possível notar um comportamento oscilatório em torno da referência em todos os ângulos, principalmente no ângulo θ .

Como os sensores utilizados em todos os ensaios foram os mesmos, o mesmo erro descrito na leitura de ψ' ocorreu neste ensaio.

Na seção seguinte, a conclusão, os resultados obtidos para cada técnica de controle são comparados para cada situação. São apresentadas tanto comparações analíticas quanto numéricas de desempenho.

8 Conclusões

Analisando os resultados encontrados nas simulações aplicadas ao modelo linear simplificado, é possível notar que todos os controladores desenvolvidos foram capazes de atingir a referência de entrada em todos os ângulos. Os controladores PIDs foram os que atingiram a referência no menor tempo, porém foram os que apresentaram os maiores sobressinais. Já os controladores LQRs, embora sejam os que apresentaram os maiores tempos para atingirem as referências, não apresentaram nenhum tipo de sobressinal. Por fim, os controladores *Backsteppings*, mesmo apresentando tempos de subidas levemente superiores aos dos controladores PIDs, apresentaram sobressinais inferiores aos mesmos.

A Tabela 8.1 apresenta a média dos erros quadráticos médios para as três técnicas empregadas para o controle dos ângulos φ e θ e do controlador P aplicado ao controle de ψ' no modelo linear simplificado. Nela, é possível notar que para os ângulos φ e θ , os controladores PIDs foram os que apresentaram, na média, os menores erros quadráticos quando aplicados ao modelo linear simplificado, seguidos pelos controladores *Backsteppings* e por fim pelos LQRs.

Tabela 8.1 – Médias dos erros quadráticos médios obtidos pelos controladores contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo linear).

Técnica de Controle	Erro Quadrático Médio
PID (φ e θ)	1,56448 °
LQR (φ e θ)	1,84835 °
<i>Backstepping</i> (φ e θ)	1,6486 °
P (ψ')	12,766 °/s

Fonte: Elaboração do próprio autor.

Para as simulações aplicadas ao modelo não linear, é possível notar que existe um acoplamento entre os ângulos, de forma que as saídas dos controladores de um determinado ângulo são influenciadas pelos demais ângulos. Também é possível notar que os ângulos de referência nem sempre são alcançados, podendo haver um *offset* devido às não linearidades do sistema, como o saturador por exemplo.

Dentre os controladores contínuos aplicados ao modelo não linear, os controladores PIDs foram os que apresentaram os menores tempos para atingirem as referências, porém foram os que apresentam maiores sobressinais e menos estabilidade. Já os controladores LQRs foram os que apresentam maiores tempos para atingirem as referências, porém oscilaram menos do que os controladores PIDs. Por fim, os controladores *Backsteppings* apresentaram tempos para atingirem as referências entre os tempos dos controladores PIDs e LQRs, apresentaram mais estabilidade que os PIDs e são os que apresentaram menos instabilidades quando comandos aplicados simultaneamente ao controle dos três ângulos foram requisitados. Isso se deve à sua componente que relaciona as taxas de saída dos demais ângulos.

Os resultados obtidos nas simulações de voo utilizando as três técnicas também se mostraram satisfatórios, visto que em todas elas o quadricóptero traçou a rota programada e os controladores atingiram os ângulos desejados. Em relação às médias dos erros quadráticos médios para a simulação de voo, os controladores *Backsteppings* foram os que apresentaram os melhores resultados no controle de φ e θ , seguidos pelos PIDs e por fim pelos LQRs. A mesma ordem se repetiu no erro quadrático médio em ψ obtido pelo controlador P utilizado em conjunto com os demais controladores.

A Tabela 8.2 apresenta a média dos erros quadráticos médios para as três técnicas empregadas para o controle dos ângulos φ e θ e do controlador P aplicado ao controle de ψ' no modelo não linear. Nela, é possível notar que para os ângulos φ e θ , os controladores LQRs foram os que apresentaram, na média, os menores erros quadráticos quando aplicados ao modelo não linear, seguidos pelos controladores PIDs e por fim pelos *Backsteppings*. Já no controle de ψ' , o controlador P apresentou um erro quadrático médio menor quando acompanhado dos controladores PIDs, seguidos pelos LQRs e por fim pelos *Backsteppings*.

Para os ensaios realizados com o modelo real utilizando os controladores discretos, é possível notar que os que apresentaram os melhores resultados foram os controladores LQRs. Estes controladores foram os que apresentaram menos oscilações em torno das referências e os que apresentaram menores sobressinais. Logo em seguida vêm os controladores *Backsteppings* e por fim os PIDs.

Tabela 8.2 – Médias dos erros quadráticos médios obtidos pelos controladores contínuos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear) nas simulações 1 a 4.

Técnica de Controle	Erro Quadrático Médio (φ e θ)	Erro Quadrático Médio (ψ')
PID	2,53731 °	2,4519 °/s
LQR	2,51695 °	2,5502 °/s
<i>Backstepping</i>	2,64249 °	2,79765 °/s

Fonte: Elaboração do próprio autor.

A Tabela 8.3 apresenta a média dos erros quadráticos médios para as três técnicas empregadas para o controle dos ângulos φ e θ e do controlador P aplicado ao controle de ψ' no modelo real discreto. Nela, é possível notar que para os ângulos φ e θ , os controladores LQRs foram os que apresentaram, na média, os menores erros quadráticos médios quando aplicados ao modelo real, seguidos pelos controladores PIDs e por fim pelos *Backsteppings*. Já no controle de ψ' , o controlador P apresentou um erro quadrático médio menor quando acompanhado dos controladores *Backsteppings*, seguidos pelos LQRs e por fim pelos PIDs.

Tabela 8.3 – Médias dos erros quadráticos médios obtidos pelos controladores discretos aplicados a φ e θ e pelo controlador P aplicado a ψ' (modelo não linear) nas simulações 1 a 4.

Técnica de Controle	Erro Quadrático Médio (φ e θ)	Erro Quadrático (ψ')
PID	2,49600 °	0,7852 °/s
LQR	2,44585 °	0,8980 °/s
<i>Backstepping</i>	2,77550 °	0,5856 °/s

Fonte: Elaboração do próprio autor.

Após a realização dos ensaios, foi possível identificar que a estrutura de teste traz um problema para a realização dos ensaios. O eixo de rotação não passa pela origem do centro de massa do quadrirrotor. O centro de massa se encontra abaixo

do eixo de rotação, conforme pode ser visto nas Figuras A.2 e A.3. Isso faz com que haja uma força de torque que faz com que o quadrirrotor tenda a se manter num ângulo próximo de zero. Isso faz com que, para se manter uma posição angular em φ ou θ diferente de zero, seja necessário que os rotores rotacionem em velocidades distintas produzindo forças diferentes para poderem anular a força de torque inserida pela estrutura.

Outro fator que prejudicou os resultados dos ensaios no quadrirrotor foi o fato de haver descompensações de massa no quadrirrotor, de modo que o centro de massa seja deslocado em relação ao eixo de rotação imposto pela estrutura giroscópica. Isso também faz com que, para se manter uma posição angular em φ ou θ igual de zero, seja necessário que os rotores rotacionem em velocidades distintas produzindo forças diferentes para poderem anular a força de torque inserida pelo deslocamento no centro de massa.

O terceiro fator que prejudicou os resultados dos ensaios foram os erros dos sensores. Os sensores utilizados não oferecem exatidão, retornando valores diferentes mesmo com as mesmas entradas, apresentando pequenas variações. Os sensores foram calibrados, mas mesmo assim apresentaram este problema. Uma possibilidade para a correção deste problema seria a substituição dos sensores utilizados por sensores mais precisos ou a utilização de filtros digitais, como o filtro de Kalman (Grewal e Andrews (2011)), por exemplo. Analisando a saída do eixo de guinada, por exemplo, é possível notar que o sensor giroscópio mesmo em repouso gera um *offset* em sua saída. Isso pode ser constatado comparando as curvas geradas e o funcionamento ao vivo do quadrirrotor.

Por fim, o último fator que prejudicou os resultados dos ensaios no quadrirrotor foi o fato de os motores serem diferentes entre si, produzindo forças diferentes mesmo quando o mesmo comando de entrada é aplicado.

Todas estas distinções do modelo real utilizado no ensaio em relação ao modelo matemático encontrado, fez com que os resultados práticos encontrados não tivessem uma resposta próxima das simulações.

Por fim, este trabalho abrangeu as principais etapas necessárias para a construção de VANTs quadrirrotores e o desenvolvimento de seus respectivos controladores de estabilização de voo, de forma que seja possível a partir do mesmo, a realização de projetos similares. Foi possível notar que o acoplamento entre os ângulos de Euler podem ser grandes e que não linearidade não mapeadas

na modelagem matemática podem implicar negativamente nos resultados obtidos no sistema real.

Também é importante notar que o tempo de resposta dos componentes escolhidos é muito importante e deve ser levado em consideração em suas escolhas, visto que o tempo de amostragem do sistema acaba tendo que seguir o maior dos menores tempos permitidos. Isso pode fazer com o que o tempo de amostragem do sistema, como um todo, seja o maior do que o permitido para se manter a estabilidade do mesmo.

Como trabalhos futuros para este trabalho estão o desenvolvimento do controlador de altitude, inclusive discreto, e a realização de ensaios em voo livre, sem acoplamento de interferências causadas pela estrutura giroscópica de teste. Desta forma será possível analisar, com mais clareza, a eficiência real dos controladores desenvolvidos.

Referências

ALVES, Ana Sophia Cavalcanti. **Estudo e Aplicação de Técnicas de Controle Embarcadas para Estabilização de Voo de Quadricopteros**, 2012, 121 f. Tese (Doutorado em Engenharia Elétrica) – Universidade Federal de Juiz de Fora.

ÅSTRÖM, K. J.; MURRAY, R. M. **Feedback Systems**. 2th ed. Princeton University Press, 2009.

BEARD, R. W. **Quadrotor Dynamics and Control**, 2008.

CABECINHAS, D.; BRÁS, S.; CUNHA, R., SILVESTRE, C.; OLIVEIRA, P. **Integrated Visual Servoing Solution to Quadrotor Stabilization and Attitude Estimation Using a Pan and Tilt Camera**, IEEE Transactions on Control Systems Technology, 2019.

CHAPMAN, A. **Types of Drones: Multi-Rotor vs Fixed-Wing vs Single Rotor vs Hybrid VTOL**, Australian Drone magazine, 2016. Disponível em <<https://www.auav.com.au/articles/drone-types/>>. Acesso em: 10 out. 2019.

CYPRIAN, A. K. **Military Use of Unmanned Aerial Vehicles – A Historical Study**. Safety & Defense, 2018.

ELBASUENY, B.; EL-SEBAH, M. I. A.; SHAABAN, S. **Dynamic Modelling and Control of quadrotor's attitude around its hover position**, 6th International Conference on Advanced Control Circuits and Systems (ACCS) & 5th International Conference on New Paradigms in Electronics & information Technology (PEIT), Egypt, 2019.

ESTEVES, D. J.; MOUTINHO, A.; AZINHEIRA, J. R. **Stabilization and Altitude Control of an Indoor Low-Cost Quadrotor: Design and Experimental Results**, IEEE International Conference on Autonomous Robot Systems and Competitions, Vila Real, 2015.

GAO, Q.; DU, M.; JI, Y. **The controller design of quadrotor UAV based on internal model control**, 36th Chinese Control Conference, 2017.

GREWAL, M. S.; ANDREWS, A. P. **Kalman Filtering: Theory and Practice Using MATLAB**. 3th ed. 2011.

KANE, T. R.; LEVINSON, D. A. **Dynamics, theory and applications**. McGraw Hill Book Company, 1985.

KARAHAN, M; KASNAKOGLU, C. **Modeling and Simulation of Quadrotor UAV Using PID Controller**, 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), Romania, 2019.

KEANE, J. F; CARR, S. S. **A Brief History of Early Unmanned Aircraft**, Johns Hopkins APL Technical Digest, Volume 32, Number 3, 2013.

KHATOON, S; GUPTA, D.; DAS, L. K. **PID & LQR Control for a Quadrotor: Modeling and Simulation**, IEEE Robotics & Automation Magazine, 2014.

KIYASHKO, M. A.; SHOLMOV, A. K.; EFREMOV, A. A. **Quadrotor mathematical and information model analysis**, IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Russia, 2017.

LEISHMAN, J. G. **A History of Helicopter Flight**, University of Maryland, 2000.

LIMA, Gabriela Viera. **Modelagem Dinâmica e Controle para Navegação de um Veículo Aéreo Não Tripulado do tipo Quadricoptero**. 2015, 120 f. Dissertação (Mestrado em Engenharia Elétrica) – Universidade Federal de Uberlândia.

MAHONY, R.; KUMAR, V.; CORKE, P. **Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor**, IEEE Robotics & Automation Magazine, 2012.

OGATA, K. **Engenharia de Controle Moderno**. 5th ed. 2011.

OKYERE, E.; BOUSBAINE, A.; POYI, G. T.; JOSEPH, A. K.; ANDRADE, J. M. **LQR controller design for quad-rotor helicopters**, The Journal of Engineering, vol. 2019.

ORAL, E.; ÇETIN, L.; UYAR, E. **A novel method on selection of q and r matrices in the theory of optimal control**, International Journal of Systems Control, 2010.

QIAN, L.; LIU, H. H. T. **Dynamics and control of a quadrotor with a cable suspended payload**, 30th Canadian Conference on Electrical and Computer Engineering (CCECE), Windsor, 2017.

SANTANA, P. H. R. Q.; BORGES, G. A. **Modelagem e controle de quadrirrotores**, SBAI, 2009.

SIELAWA, J. T. **Mecânica do Voo - Apostila para os cursos AER-280 e AER-281**. São José dos Campos, 1976.

STARR, G. P. **Introduction to Applied Digital Control**, 2006.

STEVENS, B. L.; LEWIS, F. L. **Aircraft Control and Simulation**. 2nd. ed. Hoboken, New Jersey: Wiley-Interscience, 2003.

ZHONGLI, M.; HUOXIN, L.; QIANQIAN, L.; YANMING, G.; ZUOYONG, L. **Design of Flight Attitude Control Model for the Miniquadrotor**, International Conference on Mechatronics and Automation, Japão, 2017.

ZHOU, J.; WEN, C. **Adaptative Backstepping Control of Uncertain Systems**. 2008.

ZU, Y.; ZHANG, W.; SHAN J. **A trajectory design method for quadrotor based on DMOC method and Dubins path algorithm**, 36th Chinese Control Conference, 2017.

Apêndice A – Propriedades de massa extraídas do CAD 3D

Figura A.1 – Captura de tela das propriedades da massa do quadrrrotor.

```

Massa = 1298.10256 gramas

Volume = 605711.67484 milímetros cúbicos

Área de superfície = 389607.04526 milímetros quadrados

Centro de massa: ( milímetros )
  X = 0.00000
  Y = 0.00000
  Z = 0.00000

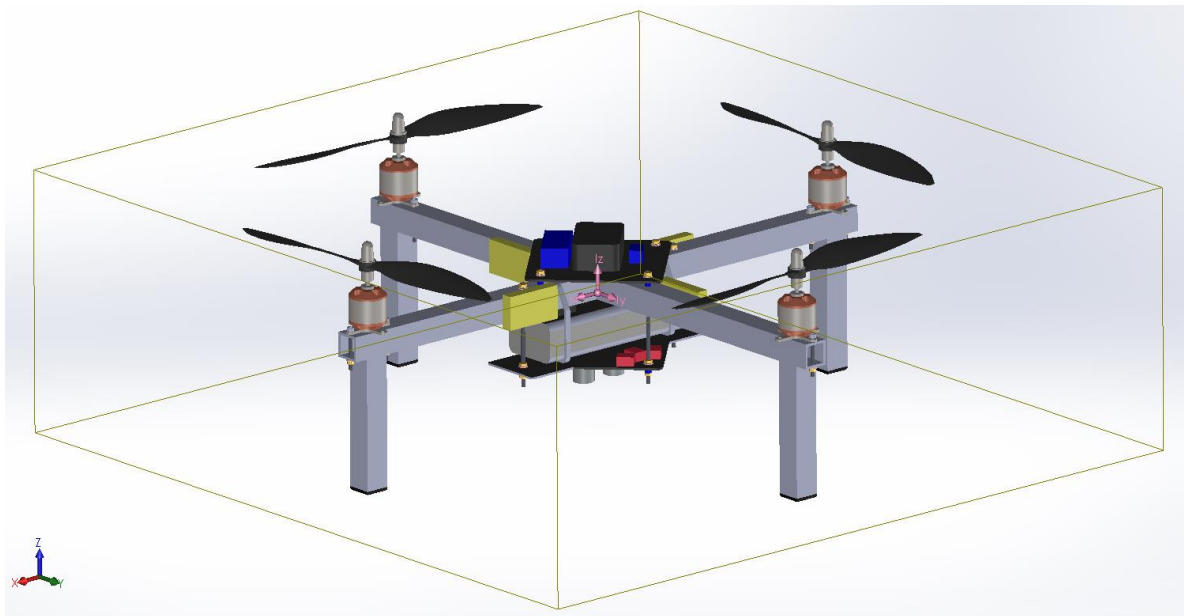
Eixos principais de inércia e momentos de inércia principais: ( gramas * milímetros quad
Tomado no centro da massa.
  lx = ( 0.99996, -0.00837, 0.00101)          Px = 13577038.56236
  ly = ( 0.00837, 0.99996, -0.00210)        Py = 13993288.14213
  lz = (-0.00099, 0.00211, 1.00000)         Pz = 24379848.09930

Momentos de inércia: ( gramas * milímetros quadrados )
Obtido no centro de massa e alinhado com o sistema de coordenadas de saída.
  Lxx = 13577078.31868      Lxy = -3461.51541      Lxz = 10704.43100
  Lyx = -3461.51541        Lyy = 13993305.03055     Lyz = -21870.59812
  Lzx = 10704.43100        Lzy = -21870.59812      Lzz = 24379791.45456

Momentos de inércia: ( gramas * milímetros quadrados )
Tomados no sistema de coordenadas de saída.
  lxx = 13577078.31868      lxy = -3461.51541      lxz = 10704.43100
  lyx = -3461.51541        lyy = 13993305.03055     lyz = -21870.59812
  lzx = 10704.43100        lzy = -21870.59812      lzz = 24379791.45456
  
```

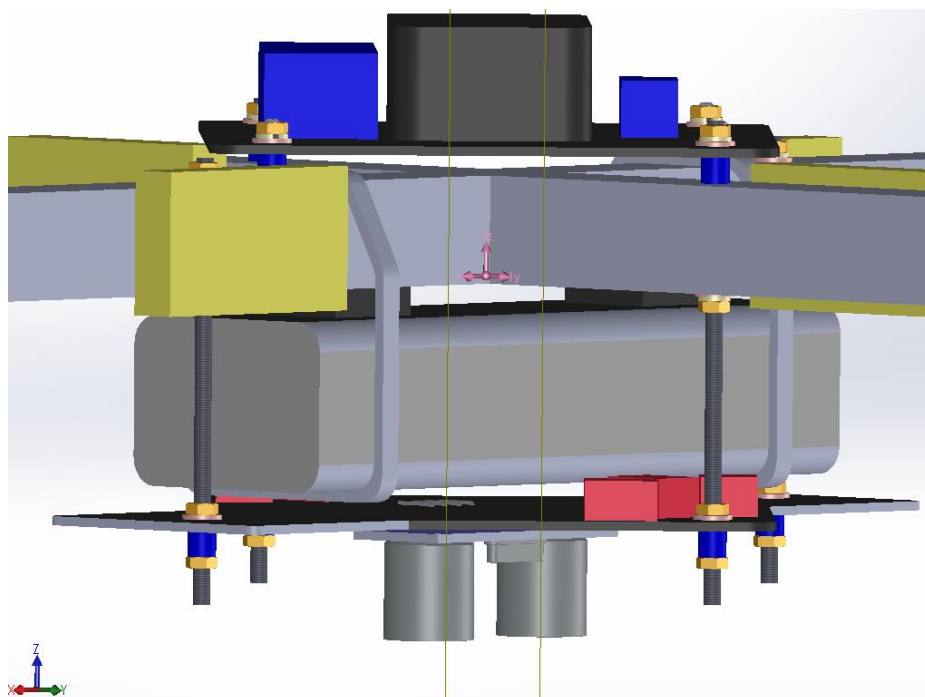
Fonte: Elaboração do próprio autor.

Figura A.2 – Captura de tela do centro de massa do quadrrrotor.



Fonte: Elaboração do próprio autor.

Figura A.3 – Captura de tela do centro de massa do quadrrirrotor (zoom).



Fonte: Elaboração do próprio autor.