



CONTROLE DE INCLINAÇÃO POR MEIO DE PID UTILIZANDO MPU 6050 E ARDUINO

João Lucas Benício da Silva*¹, Carlos Eduardo Ribeiro¹, Carlos Renato Borges Santos², José Augusto Lima Melo¹, Silas Martins Sousa² e Thiago Mendes Borges¹

¹Estudante de Engenharia Elétrica, do Instituto Federal do Triângulo Mineiro - *campus* Paracatu

²Professor do Instituto Federal do Triângulo Mineiro - *campus* Paracatu

Resumo - Este estudo aborda o controle de inclinação, uma etapa fundamental na operação de Veículos Autônomos não Tripulados (VANTs). Apresentamos uma metodologia para determinar o ângulo de inclinação desejado, baseada em fatores como Ângulos de Euler, Filtros Digitais e conceitos físicos. Inicialmente, realizamos uma revisão bibliográfica que orientou a implementação do controle PID. Essa implementação teve como objetivo validar a eficácia da abordagem proposta. Os resultados dos testes demonstraram que o sistema de controle foi capaz de acompanhar o valor desejado do ângulo de inclinação e também de resistir a perturbações de posição, confirmando, assim, a eficácia da estratégia adotada.

Palavras-Chave- Controle Digital. Inclinação. PID. VANT's.

INCLINE CONTROL THROUGH PID USING MPU 6050 AND ARDUINO

Abstract - This study addresses tilt control, a fundamental step in the operation of Unmanned Autonomous Vehicles (UAVs). We present a methodology to determine the desired slope angle, based on factors such as Euler Angles, Digital Filters and physical concepts. Initially, we carried out a bibliographic review that guided the implementation of the PID control. This implementation aimed to validate the effectiveness of the proposed approach. The test results demonstrated that the control system was able to follow the desired value of the inclination angle and also resist position disturbances, thus confirming the effectiveness of the adopted strategy.

Digital Control. PID. Tilt. VANT's.

NOMENCLATURA

K_P Ganho do controlador proporcional.

K_I Ganho do controlador integrativo.

K_D Ganho do controlador derivativo.

*joao.lucas@estudante.iftm.edu.br

I. INTRODUÇÃO

Atualmente, serviços de vigilância militar, análise topográfica de terrenos e fotografias aéreas podem ser realizados por veículos aéreos não tripulados. Devidamente equipados com sensores e câmeras de alta resolução, os drones podem capturar imagens aéreas, transmitir tais imagens em tempo real, coletar dados da topologia de um terreno e gerar modelos 3D úteis em aplicações de cartografia, engenharia e planejamento urbano. Os drones também desempenham um papel importante em missões de busca e salvamento, permitindo a rápida identificação de áreas afetadas por desastres naturais ou situações de emergência. Para isso, é necessário manter uma posição estável durante o voo, ponto crucial para garantir que as informações capturadas sejam precisas e confiáveis em todas essas áreas de aplicação [9].

Com o objetivo de controlar a inclinação ocasionada por variações durante o voo, torna-se necessário realizar o mapeamento inercial do corpo estudado. Isso é semelhante ao que é feito em aeronaves, onde são estabelecidas variáveis de rotação em torno do eixo longitudinal, rotação em torno do eixo transversal e rotação em torno do eixo perpendicular. O IMU 6050 é uma combinação de um acelerômetro e um giroscópio, que registram as grandezas de aceleração linear e orientação com base nas forças gravitacionais e inerciais da matéria. Ao integrar essa funcionalidade com uma biblioteca do Arduino, o processo de mapeamento é realizado coletando dados de variáveis para calcular o posicionamento aproximado do eixo Y. Sendo neste artigo, o único eixo de interesse.

O presente artigo traz um estudo da utilização do MPU e PID no controle da inclinação de um protótipo de drone, a fim de exemplificar a correção de turbulências atmosféricas. O MPU, ou Unidade de Medição Inercial, é empregado para monitorar a orientação da aeronave em tempo real, enquanto o controlador PID (Proporcional, Integral e Derivativo) calcula e aplica ajustes necessários para manter a inclinação desejada, contribuindo para a estabilidade e resposta eficaz do drone em condições de voo desafiadoras [3].

II. MATERIAIS E MÉTODOS

Foi utilizado o *Arduino UNO*[®] para o processamento dos dados oriundos dos sensores e para o controle dos atuadores, tais como motores. Para aquisição dos dados, o *IMU MPU 6050*, que é composto por acelerômetro e giroscópio, foi empregado para auxiliar na obtenção do ângulo da plataforma. Para controle dos motores, foi utilizado o *ESC*, que é um controlador de velocidade eletrônico, usado em projetos de Aero-modelismo.

Para a exposição dos dados, como também para analisar o projeto, foram utilizados, o *display LCD* para visualização do ângulo e do *setpoint*; assim como foi utilizado o *Processing*[®], implementado em *Python* para acompanhamento visual das mudanças de *setpointe* para mudança do ângulo desejado do sistema, foram utilizados *push-buttons*.

A Tabela 1 mostra a lista de materiais utilizados no desenvolvimento deste projeto.

Tabela 1: Materiais utilizados para a elaboração do protótipo.

Item	Quantidade
Microcontrolador Arduino UNO [®]	1
Motor A2212/13T 1000k V	2
ESC 30 A	2
Hélices 1060	2
LCD 16x2	1
Módulo I2C	1
Controle Remoto IR	1
Receptor de Infravermelho AX-1838HS	1
Haste Metalon 20x20 60 cm	1
Jumpers	-
Madeira (Interface)	-

A. Controlador Eletrônico de Velocidade - ESC 30 A

O ESC (*Electronic Speed Controller*) é um controlador eletrônico de velocidade. Além de controlá-la, também permite variar a rotação e de um motor elétrico de corrente contínua, conhecido como motores CC. Os ESC são do tipo *brushed* e *brushless*. Os motores a serem utilizados, são sem escovas, com isso, os ESCs também seguem esse modelo.

No caso dos ESCs modelo *brushless*, são controlados por meio de acionamento das bobinas. Contudo, possuem diversas funcionalidades, onde a grande maioria deles também alimentam o receptor do sistema de rádio e os servos com 5 V através de um sistema chamado de BEC. O ESC utilizado pode ser visto na Figura 1 [4].

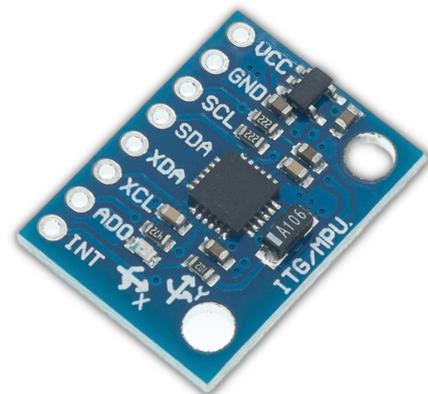
Figura 1: Modelo de Controlador Eletrônico de Velocidade (ESC) utilizado.



B. Unidade de Medida Inercial - IMU

A Unidade de Medida Inercial, trata-se de um dispositivo capaz de medir aceleração e velocidade. Genericamente, consiste de um acelerômetro e um giroscópio. Logo, não é possível medir ângulos diretamente. O MPU 6050 é um modelo de 6 DOF, onde possui um acelerômetro e um giroscópio, ambos de 3 eixos. Os mesmos, operam com 3,3 V e algumas versões possuem um regulador de tensão com possibilidade de operação com 5 V. Para sua utilização foi utilizada a biblioteca disponibilizada por [2]. A placa utilizada foi o MPU 6050. A Figura 2 ilustra o MPU utilizado.

Figura 2: Módulo MPU 6050 utilizado.



C. Motor A2212/13T 1000 kV

Quando se trata de modelagens de princípios de veículos aéreos, é necessária a utilização de motores com alta eficiência, uma vez que, tende a sofrer diversas perturbações em seu trajeto, sejam em mudanças repentinas do ar, como também objetos que desestabilizem o veículo. Os motores sem escovas escolhidos têm como característica o poder de atuação acerca da correção do ângulo desejado.

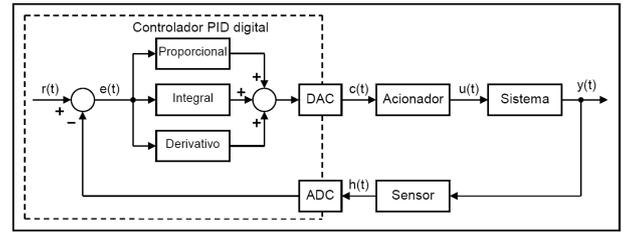
Os motores *brushless* muitas vezes estão presentes nos drones [6]. O trabalho de [9] desenvolveu um drone omnidirecional, possuindo múltiplos graus de liberdade. A vantagem deste sistema é que o veículo pode voar em qualquer direção. Neles são utilizados motores *brushless* e ESCs reversíveis.

Como especificação técnica obtida no *datasheet*, este motor apresenta 14 polos e uma máxima potência igual a 180 W. O mesmo pode ser visto na Figura 3 [7].

Figura 3: Motor A2212/13T 1000 kV utilizado.



Figura 4: Malha de Controle.



D. Controlador Proporcional-Integral-Derivativo - PID

O Controlador Proporcional Integral Derivativo (PID) é uma técnica amplamente utilizada em engenharia de controle e automação. Esse controlador é projetado para ajustar automaticamente sistemas para um valor desejado, mantendo-os estáveis e minimizando erros.

O PID opera com três termos principais: o termo proporcional, que reage ao erro atual, o termo integral, que lida com a acumulação de erros passados, e o termo derivativo, que considera as taxas de variação do erro. Esses três componentes trabalham em conjunto para ajustar o sistema em tempo real, permitindo um controle preciso em uma variedade de aplicações, desde controle de temperatura em processos industriais até automação de veículos.

O controle tradicional PID é uma das estratégias mais desenvolvidas no controle linear de sistemas, sendo este ainda muito usado em controle de sistemas industriais, devido à sua simplicidade, robustez, confiabilidade e devido à facilidade de ajuste dos parâmetros [11]. A Equação 1 mostra como os ganhos são calculados [12]:

$$u_t = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (1)$$

Na fórmula do Controlador Proporcional Integral Derivativo (PID), cada parcela desempenha um papel específico na regulação do sistema. A primeira parcela, responsável pela resposta proporcional (K_p), ajusta o sistema de acordo com o erro atual, $e(t)$. A segunda parte do somatório, referente à integral (K_i), age na acumulação dos erros passados, enquanto a última parcela, relacionada ao ganho derivativo (K_d), leva em consideração as taxas de variação do erro.

O processo é o sistema físico que está sendo controlado, e o sensor de feedback é responsável por medir a saída do processo e fornecer essa informação de volta ao controlador para que ele possa tomar decisões de controle adequadas. A interação entre esses três elementos é essencial para alcançar um controle preciso e estável em sistemas dinâmicos.

Portanto, ao considerar a malha de controle de acordo com a referência de [12], é fundamental entender como o controlador PID se integra a essa estrutura mais ampla para atingir os objetivos desejados de regulação e estabilidade do sistema. A Figura 4 mostra a malha de controle:

A figura da malha de controle apresenta uma configuração essencial para o controle de sistemas dinâmicos. A entrada é fornecida ao controlador PID, que realiza cálculos ponderados de ganhos proporcionais, integrativos e derivativos para gerar um sinal de controle. Esse sinal é convertido em um formato analógico pelo DAC e transmitido ao atuador, que executa a ação física de ajuste no sistema controlado.

Por outro lado, na realimentação, o sensor mede a saída do sistema e converte essa informação em um sinal digital por meio do ADC. Esse *feedback* é crucial, pois permite ao controlador comparar a saída real com a referência desejada e realizar ajustes contínuos para manter o sistema em conformidade com os objetivos de controle, fechando assim o ciclo de controle e garantindo a estabilidade e precisão do sistema controlado. Essa abordagem de malha de controle é fundamental na engenharia de controle e automação para sistemas complexos.

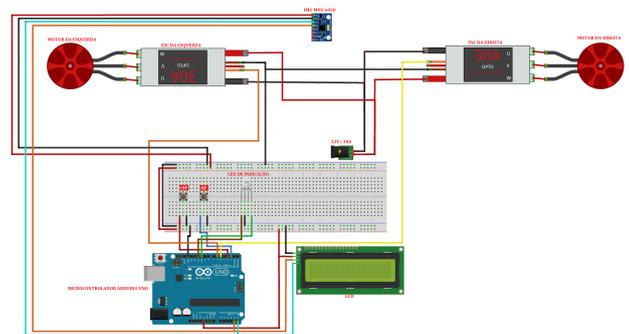
III. RESULTADOS E DISCUSSÕES

O trabalho foi dividido em três etapas para que todos os objetivos fossem alcançados. A primeira etapa integrou a revisão bibliográfica acerca do tema, uma vez que, para a realização deste trabalho, foi essencial obter um embasamento teórico para auxiliar o seu desenvolvimento.

A segunda etapa foi designada a escolha dos componentes necessários para a implementação do projeto.

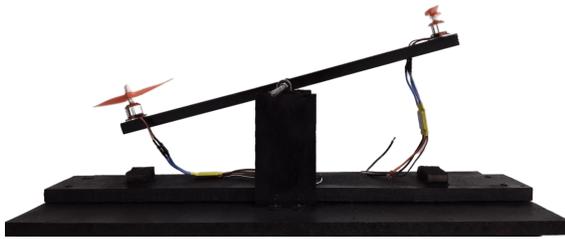
Na terceira etapa, foi desenvolvido o código para programar o Arduino® e assim, o controle PID, garantindo o equilíbrio. A Figura 5 mostra o esquemático utilizado para os testes:

Figura 5: Esquemático do projeto.



Após o esquemático desenvolvido na Figura 5, foi possível a montagem física da plataforma expressa na Figura 6.

Figura 6: Plataforma Física finalizada.



Por meio do microcontrolador, o sistema controla o PID, consequentemente lê os valores de inclinação lidos pelo MPU 6050, assim, apresentando as variáveis de *setpoint* e o ângulo definido, tanto no *display* LCD como também no ambiente integrado *Processing*[®]. Pode se ver nas Figuras 7 e 8 essas implementações.

Figura 7: Protótipo apresentado no simulador, destacando as variáveis de controle mostradas no LCD.

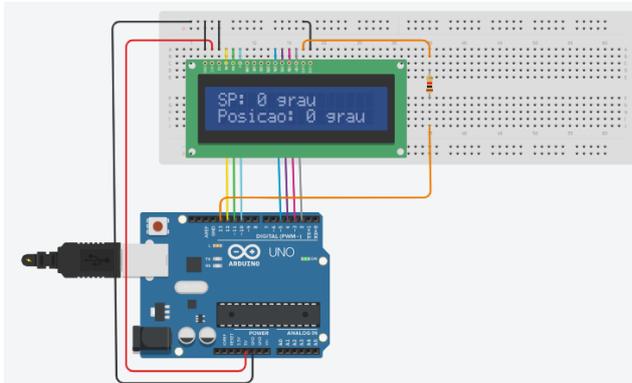


Figura 8: Visualizador de posição desenvolvido no Ambiente *Processing*[®].



Após isso os motores foram colocados na plataforma física e foi feito o teste deles, que apesar de serem idênticos, divergiam nos valores acionamento conforme a Tabela 2. Foi observado também, que a haste que segura os motores eram leves e não necessitava de potência máxima, sendo esse outro impasse já que em potência máxima os motores exigem uma

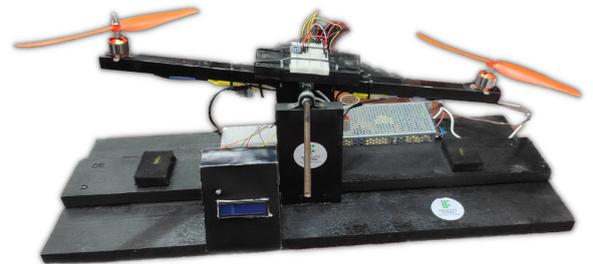
alta corrente e desligava a fonte de tensão. Uma vez resolvidos esses problemas outro contratempo foi com a biblioteca *timer1*, já que é utilizada pela biblioteca "servo.h". Para resolver este problema foi usada a biblioteca *timer2m* encontrada facilmente na internet.

Tabela 2: Relação do valor de acionamento dos motores.

Motores	Faixa de Variação	Valor de acionamento
Motor 1	1000 a 2000	1050
Motor 2	1000 a 2000	1000

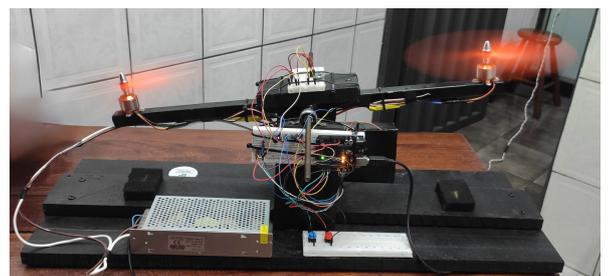
Solucionados esses contratempos, os problemas de acionamento dos motores foram solucionados, a partir disso a parte de controle do sistema foi iniciada, primeiro com valores fixos de acionamento, para evitar choques da haste com a estrutura, conforme a Figura 9.

Figura 9: Projeto com controle fixo.



Uma intervenção foi implementada para otimizar o desempenho do sistema, consistindo na adição do controle proporcional P. Esta inclusão resultou em melhorias substanciais na resposta do sistema, como ilustrado na Figura 10.

Figura 10: Protótipo configurado com controle Proporcional (K_P).



Após a eficaz implementação do controlador P, procedeu-se à incorporação do *Processing*[®], do LCD e de um LED RGB com funções distintas de sinalização: vermelho para indicar o estado "pronto para iniciar" e verde para representar o programa em execução.

Foram conduzidos testes abrangentes com vários valores de "setpoints", ajustáveis por meio de uma *push-button*, uma vez que o uso da biblioteca *Timer2* pelo infravermelho impedia a utilização simultânea de outros componentes-chave no projeto. O primeiro teste foi executado com um setpoint de 10°, conforme demonstrado nas Figuras 11, 12 e 13.

Figura 11: LCD com Setpoint de 10°.



Figura 12: Projeto com Setpoint de 10°.

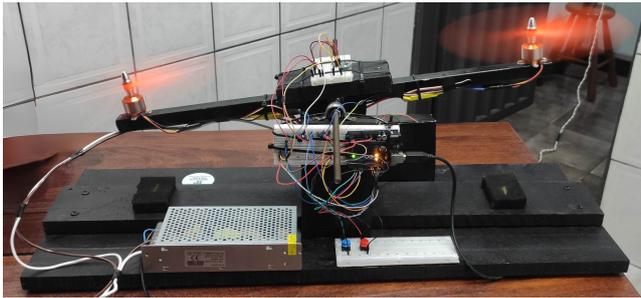
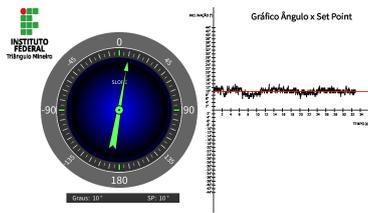


Figura 13: Processing® com Setpoint de 10°.



Logo após foi realizado um teste com um setpoint de 0° conforme mostra as Figuras 14, 15 e 16.

Figura 14: LCD com Setpoint de 0°.



Figura 15: Projeto com Setpoint de 0°.

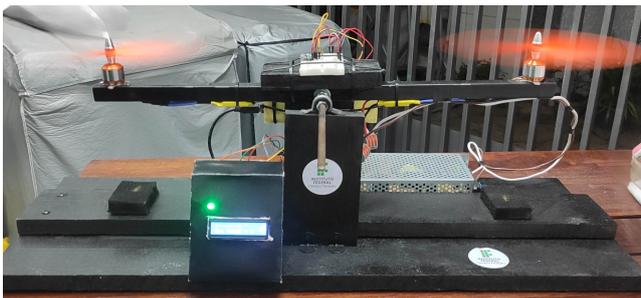
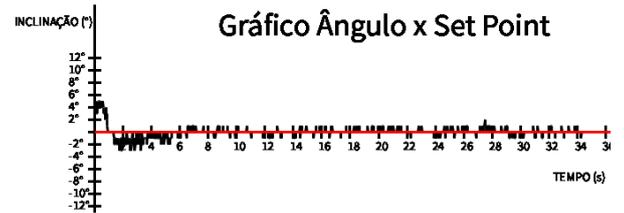


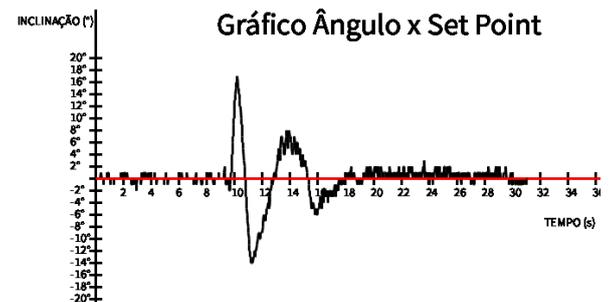
Figura 16: Gráfico apresentando erro máximo de $\pm 1^\circ$ para o setpoint de 0°.



O gráfico da Figura 16 apresenta a setpoint. contínuo e a resposta do sistema apresentou pequeno erro em regime permanente, sendo de $\pm 1^\circ$ em torno do valor de setpoint.

Para verificar a resposta do sistema, foi inserida uma perturbação manual, onde um membro do projeto modificou a posição da haste de forma abrupta, alterando-a em 16°, no instante de tempo igual a 10 segundos. A Figura 17 mostra o comportamento do sistema para a perturbação inserida.

Figura 17: Comportamento do sistema mediante à uma perturbação inserida.



Com a perturbação no instante de 10 segundos, o sistema de controle buscou imediatamente a correção. O sistema apresentou um sobressinal (*overshoot*) de 14° abaixo do setpoint. Para se estabilizar no valor desejado, foi necessário um tempo de aproximadamente 8 segundos.

IV. CONCLUSÃO

Este estudo promoveu uma amplificação da interatividade com a plataforma Arduino®, o que resultou em um aprofundamento substancial no entendimento do sensor MPU 6050. Além disso, proporcionou uma apreensão mais aprofundada das complexidades subjacentes ao controle proporcional-integral-derivativo (PID) e facilitou a incorporação sinérgica dos conceitos ligados ao controle digital de processos, microcontroladores e eletrônica de potência.

Os objetivos estipulados foram alcançados de maneira satisfatória, evidenciando um índice de erro percentual que se mantém dentro dos limites aceitáveis. Para empreendimentos subsequentes, uma direção possível é a transição da confecção do circuito da placa de ensaio (*Protoboard*) para uma placa de circuito impresso (PCB), conferindo uma maior robustez e confiabilidade ao sistema.

Adicionalmente, considera-se viável a substituição do microcontrolador por uma unidade de processamento mais avançada, como o caso exemplar do ESP32, que promove um

desempenho superior. Nesse contexto, a implementação de comunicação sem fio também é uma perspectiva interessante para aprimorar ainda mais o projeto.

REFERÊNCIAS

- [1] BRASIL. Ministério da Economia. Secretaria Especial de Previdência e Trabalho. NR 18 - **Segurança e Saúde no Trabalho na Indústria da Construção**. Brasília, DF, 2021.
- [2] C. R. B. SANTOS. **MPU 6050**. 20/03/2023. Disponível em <https://github.com/santoscrb/MPU-6050>.
- [3] *Electronoobs*. **PID control with arduino**. Acesso em 21 de Setembro de 2022, em: http://electronoobs.com/eng_robotica_tut6.php.
- [4] SPEED CONTROLLER. **Readytosky Simonk 30A Electronic Speed Controller**. readytosky.com/e_productshow/.
- [5] Lage, E.(2020) Ângulos de Euler, **Rev. Ciência Elem.**, V8(03):043, 2.2. doi.org/10.24927/rce2020.043.
- [6] MEYER, Ivo Zatti Lima. **Caracterização de Motores Brushless de Pequeno Porte para Veículos Aéreos Não Tripulados**. Dissertação (Mestrado em Engenharia Mecânica) - Programa de Pós-Graduação em Engenharia Mecânica, Universidade Federal de Minas Gerais, Belo Horizonte, 2019.
- [7] MOTOR BRUSHLESS. **A2212/13T 1000KV Brushless Motor**. <https://himalayansolution.com/product/a2212-13t-1000kv-brushless-motor>.
- [8] OLIVEIRA, Waldri dos Santos; GONÇALVES, Eduardo. Nunes. Implementação em c: filtro de Kalman, fusão de sensores para determinação de ângulos. **ForScience**: revista científica do IFMG, Formiga, v. 5, n. 3, e00287, jul./dez. 2017.
- [9] S. Park, J. Lee, J. Ahn, M. Kim, J. Her, G. Yang and D. Lee. **ODAR: Aerial Manipulation Platform Enabling Omnidirectional Wrench Generation**. IEEE/ASME Transactions on Mechatronics, vol. 23, no. 4, pp. 1907–1918, Ago. 2018, 10.1109/TMECH.2018.2848255.
- [10] SOUZA, T. F. C. de. **Instrumentação e Análise do Consumo Energético Baseado na Pose de um Veículo Elétrico de Competição**. Porto Alegre: Universidade Federal do Rio Grande do Sul, Escola de Engenharia, Departamento de Engenharia Elétrica, 2019.
- [11] C. Xia, **Permanent Magnet Brushless DC Motor Drives and Controls**. Singapore: John Wiley and Sons Singapore Pte. Ltd., 2012.
- [12] K. Ogata; B. Severo, **Engenharia de controle moderno**. 1998. Prentice Hall do Brasil.