



Universidade Federal de Minas Gerais
Escola de Engenharia
Curso de Graduação em Engenharia Elétrica

Projeto e Construção de um Hovercraft para Ensino de Técnicas de Controle

Monografia submetida à banca examinadora designada pelo Colegiado do curso de Graduação de Engenharia Elétrica da Universidade Federal de Minas Gerais, como parte dos requisitos necessários à obtenção do grau de Bacharel em Engenharia Elétrica.

Pedro Ivan Ribeiro Fernandes
Orientador: Prof. Vítor Costa da Silva Campos

Belo Horizonte, Outubro de 2020

Resumo

Este projeto consiste no desenvolvimento de uma planta didática móvel para ensino prático de técnicas de controle em laboratório. Um *Hovercraft* foi escolhido para este fim por se tratar de um veículo versátil com ampla gama de utilização, apesar de pouco conhecido. O escopo do projeto pode ser dividido em quatro partes: projeto e construção do protótipo, elaboração de um ambiente virtual didático com uma interface intuitiva de utilização e demonstração do seu potencial didático para o ensino de controle. Na etapa de projeto, é apresentada uma revisão teórica do funcionamento de um *Hovercraft*, levantamento de sensores, atuadores e demais componentes eletrônicos e mecânicos que serão utilizados, desenho do projeto 3D e projeto eletrônico embarcado. Na etapa de construção, é descrito o processo de montagem e validação funcional do protótipo. Em seguida, é apresentada a implementação do *firmware* do microcontrolador e o desenvolvimento da interface gráfica de usuário que será utilizada para interagir em tempo real com o sistema embarcado de maneira remota, por meio de comunicação sem fio. Utilizando a interface desenvolvida, é realizada a calibração dos sensores, a modelagem empírica do comportamento dinâmico do sistema e a sua validação. Por fim, é proposto e validado um controlador para realizar o controle em malha fechada de velocidade linear e angular. Assim, é demonstrada a aplicabilidade do protótipo construído como ferramenta didática para o ensino de engenharia de controle. São ainda explorados, ao longo do desenvolvimento do protótipo, diferentes áreas abarcadas e correlatas da engenharia de controle, como instrumentação, projeto de sistemas eletrônicos embarcados, comunicação digital sem fio, sistemas operacionais de tempo real, modelagem de sistemas dinâmicos e projeto de controle de sistemas multivariáveis.

Palavras-chave: Hovercraft, Eletrônica Embarcada, Modelagem de Sistemas Dinâmicos, Técnicas de Controle, Ferramenta didática.

Abstract

This paper describes the development of a vehicle prototype for use as a Controls engineering didactic tool in a laboratory environment. A hovercraft was chosen for this purpose because it is a versatile vehicle with a wide range of applications, although considerably little known among graduate students. The scope of the project can be divided into four parts: design and construction of the prototype, development of a didactic virtual environment with an intuitive user interface and demonstration of its potential as a learning tool in the field of Controls Engineering. In the design stage, a brief technical review of the working principles of a hovercraft is presented, followed by a research of relevant sensors, actuators and other electronic and mechanical components for implementation, as well as the design of a 3D model of the prototype and the embedded electronics circuitry design. In the construction stage, the assembly process and functional validation of the prototype are described. Then, the implementation of the microcontroller firmware and the development of the graphical user interface that will be used for interacting in real time with the remote embedded hardware are presented. Using this interface, sensor calibration is then performed, followed by the empirical modeling of the dynamic behavior of the system and its validation. Finally, controllers are implemented, tuned and validated to perform closed loop control of linear and angular speed. Thus, the applicability of the built prototype as a Control System's learning tool is demonstrated. Furthermore, Within the prototype development process, several correlated fields are also explored, such as instrumentation, electronic embedded systems design, digital wireless communication via TCP-IP protocol, use of a real-time operating system, dynamic system modelling and controller design for multivariable systems.

Keywords: Hovercraft, Embedded Electronics, Dynamic System Modelling, Controls Engineering, Prototype, Didactic Tool.

Agradecimentos

Em primeiro lugar, agradeço a Deus por nortear cada etapa da minha jornada acadêmica e profissional e pela benção de me tornar um profissional competente para que eu possa realizar a missão de tornar a vida de outras pessoas melhor a cada dia com o meu trabalho. Agradeço também a meus familiares, amigos e mentores pelo carinho e dedicação incondicionais, me incentivando nos momentos de dificuldade e me aconselhando com sabedoria nos momentos de dúvida. Por fim, agradeço ao meu orientador Prof. Victor Costa pela paciência e disponibilidade durante todo o desenvolvimento deste projeto. Sem suas sugestões, correções e contribuições ele não seria possível.

"Esqueça erros, esqueça fracassos, esqueça tudo, exceto o que vai fazer agora. E faça. Hoje é seu dia de sorte."

William James Durant

Sumário

Resumo	i
Abstract	ii
Agradecimentos	iii
Lista de Figuras	vii
1 Introdução	1
1.1 Motivação e Justificativa	1
1.2 Objetivos do Trabalho	2
2 Pesquisa Bibliográfica	5
2.1 Projeto de um ACV	6
3 Metodologia	9
3.1 Levantamento de requisitos	9
3.2 Levantamento de componentes	10
3.2.1 Motorização e acionamento	10
3.2.2 Microcontrolador	10
3.2.3 Sensores	11
3.3 Esquema elétrico de montagem	12
3.4 Projeto 3D da estrutura	13
3.5 Firmware do Microcontrolador	14
3.6 Calibração do <i>Inertial Measurement Unit</i> (IMU)	17
3.6.1 Efeito de ângulo de elevação e de margem	18
3.6.2 Distorções do Campo magnético	18
3.7 Interface gráfica de usuário (GUI) em MATLAB	19
3.8 Paradigmas para a modelagem do comportamento dinâmico do hovercraft .	21
3.9 Ensaios e modelagem da velocidade linear	22
3.10 Filtragem de velocidade linear	25
3.11 Ensaios e modelagem da velocidade angular	26
3.12 Estrutura de Controle	29
3.12.1 Controlador de Velocidade linear	30
3.12.2 Controlador de velocidade angular	31
3.12.3 Validação da estrutura de controle	33

4 Conclusões	37
4.1 Observações e Propostas de Continuidade	37
4.2 Considerações Finais	38
Referências Bibliográficas	39

Lista de Figuras

2.1	Diferenciação técnica entre diversos veículos aquáticos que se baseiam no mesmo princípio de hovercraft [15].	6
2.2	Ilustração do princípio de sustentação de um ACV [15].	6
3.1	Motot QF2822L	11
3.2	Driver de velocidade Hobbywing Skywalker V2.	11
3.3	Mapa de pinos do NodeMCU-32S [4].	11
3.4	Módulo sensor IMU MPU9250 [8].	11
3.5	Diagrama esquemático de conexões entre os componentes eletrônicos.	13
3.6	Diagrama das conexões da placa de circuito do Hovercraft.	14
3.7	Placa de circuito impresso para o Hovercraft.	15
3.8	Modelo 3D do QX-motor.	16
3.9	Modelo 3D do servo SG-5010.	16
3.10	Diversas vistas do projeto 3D do Hovercraft.	16
3.11	Modelo 3D do protótipo com todas os componentes utilizados.	17
3.12	Foto do protótipo construído.	17
3.13	Janela para salvar os dados obtidos no ensaio.	19
3.14	Aba para calibração do magnetômetro.	20
3.15	Aba para realização de ensaios em malha aberta com o protótipo.	21
3.16	Aba para realização de ensaios em malha fechada com o protótipo.	22
3.17	Três ensaios de resposta ao degrau de 10% de Duty-Cycle.	23
3.18	Velocidade em função do <i>throttle</i> sob distintas condições de <i>lift</i>	24
3.19	Validação da modelagem de velocidade linear.	25
3.20	Comparação entre os sinais lidos pela interface antes e depois da adição do filtro média-móvel.	26
3.21	Três ensaios de resposta ao degrau de 15° de amplitude.	27
3.22	Relação entre velocidade angular e ângulo de leme sob diferentes valores de <i>throttle</i>	28
3.23	Validação do modelo obtido de velocidade angular.	29
3.24	Diagrama de blocos do modelo identificado do sistema.	30
3.25	Comparação do lugar geométrico das raízes do sistema modelado e do sistema desejado, respectivamente.	32
3.26	Lugar geométrico das raízes e resposta ao degrau do sistema em malha fechada com o controlador sintonizado.	33
3.27	Modelo em simulink do sistema de controle projetado.	34
3.28	Ensaio para validação do sistema controlado.	36

Capítulo 1

Introdução

Atualmente, no cenário mundial, observa-se um intenso processo de popularização dos drones e dos veículos aéreos movidos por propulsão a hélices ou turbinas. Não seria estranho ver num futuro próximo estes tipos de veículos realizando as mais diversas funções, de entregar encomendas a auxiliar na busca e salvamento de pessoas em localidades remotas, tal é a versatilidade dessa tecnologia. No entanto, esses avanços só são possíveis graças a uma compreensão avançada de dinâmica e aerodinâmica, da teoria de controle (em especial de sistemas multivariáveis), e ainda de camadas diversas de tecnologias afins, como, por exemplo, navegação remota por GPS, inteligência artificial, visão computacional, automação e sistemas embarcados.

Nesse contexto, é interessante recordar de um tempo no qual a compreensão dessas áreas de conhecimento tão essenciais à tecnologia atual ainda era tímida e experimental. No ano de 1953, o inventor britânico Sir Christopher Cockerell apresentou ao mundo o primeiro *Hovercraft*. A ideia básica de sua invenção se assemelha muito ao princípio de funcionamento dos drones atuais. Por meio da propulsão gerada por turbinas (em geral uma na vertical, que garante sustentação e uma na horizontal, que provém a força propulsora), o primeiro veículo híbrido, que era semelhante a um barco, deslizava quase sem atrito sobre qualquer superfície, devido ao chamado efeito-terra [15].

Fixando uma saia ou uma boia de tecido flexível como borracha às laterais do veículo, a turbina vertical criava um bolsão de ar capaz de reduzir drasticamente o atrito com a superfície, fosse água ou terra, resultando numa maneira elegante e inovadora de transportar passageiros através do canal da Mancha, entre Dover (Reino Unido) e Calais (França).

1.1 Motivação e Justificativa

A proposta deste trabalho de conclusão de curso é realizar um resgate dessa invenção singular, explorando tecnologias atuais para demonstrar a riqueza de possibilidades a serem exploradas sobre ela. O projeto deste veículo envolve conhecimentos de natureza teórica nas áreas de mecânica fundamental, aerodinâmica, motorização elétrica e microeletrônica aplicada. Sua construção demanda ainda uma segunda camada de conhecimentos técnicos práticos, englobando o desenho e impressão 3D, prototipação de um sistema eletrônico embarcado, instrumentação e acionamento de diferentes motores elétricos. O controle de um *Hovercraft*, por fim, enriquece ainda mais o escopo deste trabalho, agregando aporte teórico de modelagem e controle de sistemas lineares e programação de microcontroladores.

Trata-se, portanto, de um projeto de cunho multidisciplinar complexo.

No entanto, um trabalho de conclusão de curso deve não apenas apresentar um desafio interessante, mas também abordar um problema do mundo real ou agregar valor para a sociedade de alguma forma. Sendo assim, objetiva-se realizar a construção de um pequeno *Hovercraft* de duas turbinas para ser utilizado como uma planta didática para o ensino de diversas técnicas modernas de controle. O protótipo construído deve ser compreendido como um mínimo produto viável (MVP, em inglês). Para tanto, é importante desenvolver um ambiente amigável e de fácil interação com o *hovercraft*, a fim de possibilitar o aprendizado na prática de maneira interessante e relativamente autônoma pelo estudante de controle.

Analisando a gama vasta de plantas didáticas para ensino prático de engenharia de controle, observa-se que são produtos de nicho e, portanto, de altíssimo valor agregado. Além disso, outra tendência é a aproximação com plantas industriais, utilizando adaptações para fomentar a intimidade do aluno com hardwares industriais. Por outro lado, essa aproximação com a experiência na indústria não contempla outra gama de aplicações da teoria de controle: o desenvolvimento de produtos e de bens de consumo, em especial os veículos e meios de transporte. Por fim, percebe-se que as plantas didáticas desse tipo priorizam a robustez em relação à versatilidade, impossibilitando o acompanhamento do desenvolvimento de novas tecnologias.

Nesse contexto, a motivação deste trabalho é contrapor todos esses três aspectos, desenvolvendo um projeto de baixo custo e com foco em um veículo móvel. Mais do que isso, deseja-se obter uma aplicação que seja aberta e acessível, permitindo a implementação de funcionalidades adicionais com o passar do tempo. Dessa forma, espera-se atingir a primeira versão do protótipo dessa planta para ensino de controle, demonstrando o potencial didático e a versatilidade da aplicação escolhida.

1.2 Objetivos do Trabalho

Conforme apresentado, o escopo deste trabalho pode ser dividido em quatro partes: projeto e construção do protótipo, criação de um ambiente didático para sua utilização e demonstração do seu potencial didático para o ensino de controle. Sendo assim, ficam estabelecidos os objetivos esperados em cada uma das etapas.

- Projeto estrutural, eletrônico e funcional:
 - revisão teórica e definição de topologia e estrutura física;
 - dimensionamento de baterias e motorização;
 - levantamento e seleção de componentes eletrônicos;
 - elaboração do diagrama elétrico de conexão entre os componentes;
 - projeto 3D da estrutura física do hovercraft;
 - projeto e prototipação da placa de circuito.
- Construção e Montagem:
 - construção da estrutura física do hovercraft;
 - soldagem e fixação de componentes eletrônicos na estrutura;

- testes de requisitos funcionais;
- implementação de melhorias construtivas.
- Desenvolvimento de aplicação de software e código do microcontrolador:
 - escolha das plataformas didáticas para implementação;
 - desenvolvimento de firmware para microcontrolador, englobando leitura dos sensores, acionamento dos motores, comunicação sem fio, programação *over-the-air* e estrutura de controle;
 - levantamento de funcionalidades para a interface didática;
 - implementação da interface gráfica do usuário para interagir em tempo real com o hardware embarcado.
- Modelagem e controle do protótipo:
 - modelagem empírica do comportamento dinâmico do protótipo construído;
 - validação experimental dos modelos obtidos, realizando ensaios em malha aberta;
 - análise de técnicas de controle apropriadas à solução do problema;
 - síntese de controlador e sintonia fina a partir de ensaios em malha fechada;
 - validação do controlador projetado.

Capítulo 2

Pesquisa Bibliográfica

Desde sua invenção em 1953, a ideia principal na qual se baseia o hovercraft, a redução do atrito com a superfície de deslocamento por meio de uma almofada de ar pressurizado, sofreu diversas modificações para produzir aplicações com objetivos diversos. Os veículos conhecidos como hovercrafts são divididos em duas categorias principais: os ACVs (*air-cushion vehicles*) e SESs (*surface-effect ships*). A diferenciação técnica entre estas categorias está na forma com que flutuam ou boiam. ACVs são suportados totalmente por uma almofada de ar, com uma cortina de ar (produzida por um fluxo de alta pressão) ou um sistema de saia flexível em torno de sua periferia para vedar o ar, em geral impulsionado por hélices ou por turbinas. Já os SESs são veículos estritamente aquáticos, que substituem a saia ou almofada flexível por uma estrutura rígida em sua parte inferior, permitindo redução da potência propulsora para se manterem flutuando e em troca, abrindo mão da capacidade de se locomoverem tanto na água quanto na terra [15].

A partir da ideia inicial de usar o fluxo de ar perpendicular ao movimento para reduzir o atrito, foram posteriormente desenvolvidas aplicações primas que se baseiam no chamado efeito-solo. Esse efeito aerodinâmico pode ser definido como uma força de reação que surge sobre aeronaves pela propulsão do ar contra a superfície inferior e facilita a decolagem de helicópteros e de drones, por exemplo. Os chamados WIGs (*Wing-in-ground effect craft*) e PARWIGs (*power augmented ram wing craft*) se baseiam nesse princípio, apresentando um movimento similar a um voo rasante, sendo suportadas por elevação dinâmica, em vez de uma almofada estática. O WIG inicialmente flutua na água e sua decolagem é semelhante à de um hidroavião. Asas similares às de um avião posicionadas perto do solo geram elevação nas suas superfícies pressurizadas, o que aumenta significativamente devido ao efeito-solo. As características aero-hidrodinâmicas de uma WIG são, portanto, uma otimização significativa do design de um hidroavião para melhorar sua carga útil [15].

Neste recorte da teoria por trás desse tipo de veículo, decidiu-se por um enfoque sobre os ACVs devido a dois aspectos principais: sua característica anfíbia, interessante em especial para produzir um protótipo que possa ser testado tanto na terra quanto na água; e a sua facilidade de construção, pois a saia que conterá o ar pressurizado pode ser construída de maneira relativamente simples utilizando um material impermeável e elástico como a borracha, lona ou plástico.

Nesse ponto, é importante ressaltar ainda que, embora não se pretenda usufruir da característica anfíbia desse tipo de hovercraft neste projeto, trata-se de uma potencialidade importante para uma futura iteração do protótipo.

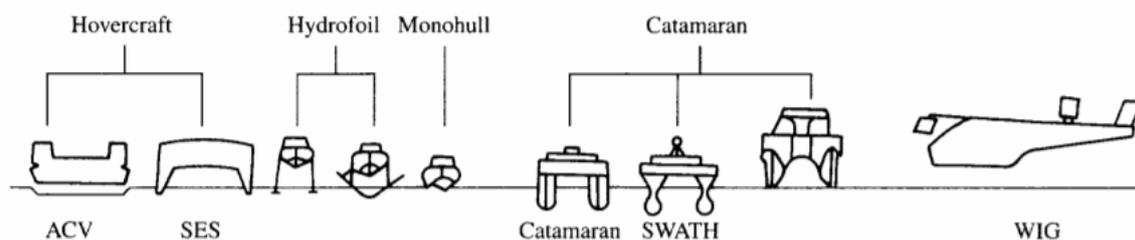


Figura 2.1: Diferenciação técnica entre diversos veículos aquáticos que se baseiam no mesmo princípio de hovercraft [15].

2.1 Projeto de um ACV

No projeto de um *hovercraft* com saia, ou ACV, parte do fluxo de ar gerado pelas hélices ou turbina é fornecido a uma cavidade de ar sob o veículo. O volume dessa câmara é ampliado por uma saia flexível ao seu redor, o que possibilita ao *hovercraft* pairar sobre o solo e superar obstáculos. O aumento da pressão é o resultado da diferença entre o volume de ar que entra na cavidade pela ação da turbina e que sai por pequenas fendas em sua estrutura. Quando atinge o equilíbrio, a pressão é grande o suficiente para suportar o peso do veículo, e o ar pressurizado escapa por fendas na parte inferior interna da estrutura, reduzindo o atrito com a superfície (figura 2.2).

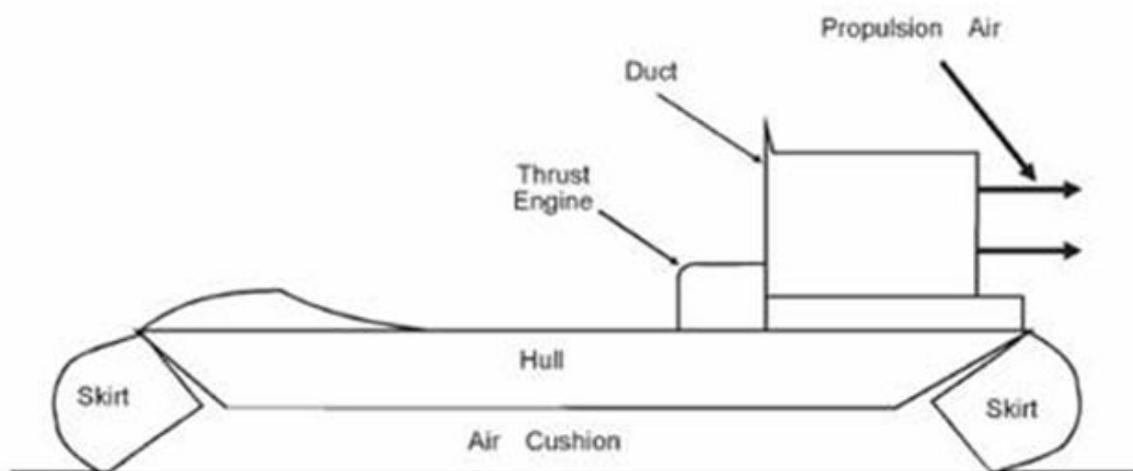


Figura 2.2: Ilustração do princípio de sustentação de um ACV [15].

Como ponto de partida para o projeto do *hovercraft* deste trabalho, propõe-se fazer uma breve análise do projeto desenvolvido por Consten (2003) [5]. Muito embora o projeto não envolva a construção física do protótipo, sua análise traz *insights* interessantes e será motivação de diversas decisões para esse projeto posteriormente.

No trabalho, são utilizados dois motores c.c., cada qual alimentado por uma bateria de lítio independente e acoplado a um ventilador. O motor de propulsão é controlado por um driver de velocidade comercial, enquanto o de sustentação é alimentado diretamente pela bateria, fornecendo uma potência constante. Conforme apresentado pelo autor, essa escolha por simplicidade de projeto introduz uma desvantagem significativa: sem esse controle de

velocidade, abre-se mão de controlar o atrito entre o veículo e a superfície sobre a qual se desloca, o que penaliza a sua estabilização em uma posição estática. A redução da potência do ventilador de sustentação poderia ter sido utilizada para aumentar propositalmente o atrito com a superfície, atuando como um freio que poderia ser usado para estabilizar o *hovercraft*.

Além disso, um leme controlado por um servomotor é empregado para direcionar o fluxo de ar do motor de propulsão para diferentes direções, produzindo movimento de guinada ou giro e favorecendo a manobrabilidade do *hovercraft*. Como sensor, optou-se por uma câmera *X10 XCam2*, equipada com um transmissor *wireless* que envia o vídeo em tempo real ao computador, onde deve ser então processado a fim de determinar a posição e velocidade do *hovercraft*. Os atuadores são conectados diretamente a um driver que recebe sinais de rádio *AM* de um computador remoto, o qual implementa o controle enquanto processa as imagens provenientes da câmera via comunicação *wireless*. A fim de implementar seu controle de maneira autônoma para seguimento de trajetória por meio de um computador, em vez de um controle manual utilizando um joystick, o sistema *RTLlinux* foi utilizado, de forma a atender às restrições de controle em tempo real.

Como foi percebido por Consten (2003), o sensoriamento por câmeras demanda elevada complexidade computacional que foge do escopo técnico necessário ao projeto, sendo necessário o emprego de algoritmos de visão computacional. Além disso, torna-se necessário o envio de grandes pacotes de dados (vídeo em tempo real) e sua performance torna-se limitada a um ambiente com boas condições de iluminação e referenciais estáticos. Como alternativa, uma variedade de sensores pode ser utilizada, dos quais se destacam:

1. Giroscópio, a fim de sensorar a velocidade angular [1].
2. Magnetômetros em três eixos, a fim de determinar a orientação do *hovercraft* em relação a um eixo tridimensional de referência [1].
3. Sensor GPS, capaz de determinar com exatidão a posição do *hovercraft* no espaço em latitude e longitude, bem como sua velocidade escalar [2].
4. Sensor óptico, capaz de mensurar o deslocamento do *hovercraft* em dois eixos, para se obter uma estimativa das velocidades via diferenciação numérica [9].
5. Acelerômetros em três eixos: embora esse sensor seja incapaz de medir velocidade sem acúmulo de erro ao longo do tempo (o que ocorre ao realizar simplesmente a integração numérica da aceleração), trata-se de um sensor acessório interessante, empregado para refinar técnicas de medição de velocidade vetorial [13].

Embora não seja possível implementar um sistema de controle capaz de navegar o *hovercraft* pelo espaço livre com o uso desses sensores, é possível projetar um sistema de controle de malha fechada para sua velocidade linear e angular, o que seria suficiente para se realizar um estudo sobre técnicas de modelagem, controle de sistemas lineares e projeto de controladores.

Uma observação interessante feita por Costen (2003) é a necessidade de se centralizar o máximo possível o centro de massa do *hovercraft* para favorecer sua controlabilidade. A posição do centro de massa deve coincidir com o posicionamento da turbina de sustentação, ou então a estrutura de *hovercraft* deve forçar uma distribuição equilibrada de fluxo de ar para garantir sua estabilidade [5].

Por fim, Consten apresenta uma solução de malha aberta para o controle remoto do hovercraft, mas devido à dificuldade de implementar um sensoriamento remoto via câmeras, não foi implementado um controle de malha fechada. No entanto, seu modelo dinâmico foi levantado usando o método de Euler-Newton. Alternativamente, seria interessante realizar uma modelagem empírica, fornecendo um recorte temporal do comportamento dinâmico do protótipo, podendo ser refeito no caso de mudanças construtivas. Além disso, recomendações são apresentadas em relação à distribuição de peso no *hovercraft*, chamando atenção para seu forte efeito na estabilidade e também para o posicionamento das baterias, que são os itens mais pesados do protótipo e devem ser fixadas de tal forma que possam ser reposicionadas, após seu recarregamento, exatamente no mesmo lugar.

Capítulo 3

Metodologia

Antes de introduzir a metodologia do projeto e a construção do hovercraft, é importante pontuar que diversos projetos faça-você-mesmo (DIY) vêm sendo realizados nos últimos anos por hobistas e entusiastas da eletrônica embarcada. Embora seus resultados não estejam devidamente documentados para serem utilizados como referências bibliográficas, é necessário ressaltar que não só serviram de inspiração para este projeto, mas também trouxeram diversos *insights* interessantes quanto às diversas opções de projeto.

3.1 Levantamento de requisitos

O primeiro ponto de decisão importante quanto ao projeto de um *hovercraft* é o número de propulsores. Conforme explorado na pesquisa bibliográfica, a utilização de propulsores independentes produz uma diferença significativa de desempenho, pois torna-se possível desacoplar a sustentação e movimentação do *hovercraft*, permitindo que ele se mantenha estável sem se mover, o que pode ser obtido reduzindo a propulsão do propulsor de sustentação para aumentar o atrito e desligando o propulsor de movimentação.

Igualmente importante é a definição da técnica empregada para permitir a manobrabilidade do *hovercraft*. A maneira mais simples de se atingir esse objetivo é o emprego de um leme na direção do fluxo de ar gerado pelo propulsor de movimentação. Esse leme pode ser movimentado pelo acionamento de um servomotor acoplado a ele.

Para fornecer energia ao sistema, as baterias de lítio se apresentam como a opção mais viável, devido a sua alta densidade de energia, seus relativamente baixos peso e custo, praticidade de recarga e grande potência de saída.

Conforme apresentado na pesquisa bibliográfica, sensores são importantes para a orientação espacial do hovercraft, sendo componentes essenciais à implementação do controle em malha fechada. As variáveis mais essenciais identificadas foram: velocidades linear e angular e posição angular em relação a um eixo de referência. No entanto, mais do que apenas realizar medições confiáveis, é interessante produzir redundância a fim de contornar as limitações de precisão e exatidão de cada um dos sensores escolhidos, bem como gerar uma abundância de dados para aumentar a versatilidade de aplicações didáticas futuras do protótipo.

A fim de realizar as operações e o controle do *hovercraft*, é interessante escolher um microcontrolador apropriado, levando em consideração fatores básicos como a tensão de alimentação e dos pinos de entrada e saída, a capacidade de fornecimento de corrente para os

componentes a ele conectados, a memória de dados e de programa, a capacidade de processamento, o preço e a disponibilidade. Além disso, dois requisitos essenciais são a possibilidade de programação sem fio (*Flashing Over-the-air - FOTA*) e a comunicação sem fio, a fim de permitir o registro de dados dos ensaios remotamente. Por se tratar de um protótipo único, a escalabilidade do projeto pode ser preterida em relação a praticidade de programação e de uso.

Em razão da necessidade de prototipagem rápida e de reprodutibilidade do projeto para permitir eventuais melhorias futuras, a impressão 3D em PLA transparente se apresenta como uma escolha interessante para a construção, produzindo uma estrutura leve, porém resistente.

Com relação aos ambientes em que o software e firmware são desenvolvidos, um ponto relevante é a escolha por ambientes familiares aos docentes e discentes, de forma a permitir a melhoria contínua da interface gráfica e a fácil modificação dos códigos. Para tanto, foram escolhidos o MATLAB para implementar a interface com o usuário e o Arduino para implementação do firmware do microcontrolador.

3.2 Levantamento de componentes

3.2.1 Motorização e acionamento

Para pequenas aplicações, como drones e aeromodelos, os motores elétricos do tipo brushless são especialmente apropriados. No entanto, faz-se necessário acoplar a eles uma turbina ou hélices e ainda uma estrutura para protegê-la. Assim, como uma solução prática, escolheu-se o modelo QX-motor QF2822L(2222), por se tratar de um motor com turbina já dimensionada e acoplada.

Para acionamento desse motor, decidiu-se por um driver de velocidade também comercial e conhecido entre hobistas, o Hobbywing Skywalker V2 [7]. Com até 40A na saída e controle PWM, esse modelo se apresenta como uma escolha prática e de fácil utilização. Além disso, esse modelo apresenta uma saída auxiliar regulada de 5V, que é ideal para alimentar a maioria dos microcontroladores mais comuns.

Por fim, a fim de acionar o leme que direciona o fluxo de ar do motor de propulsão para produzir movimento de guinada do hovercraft, é utilizado o servomotor SG-5010 [10]. Com 47g, 180° de movimento e torque estático de 5.5kg/cm, trata-se de um componente robusto para permitir a movimentação precisa do leme.

3.2.2 Microcontrolador

Com relação à escolha de microcontrolador, os primeiros fatores considerados foram a conectividade Wi-Fi, a capacidade de processamento rápido e paralelo e a praticidade de implementação do firmware. Nesse sentido, o microcontrolador ESP32-WROOM-32 [14] condensam funcionalidades como Wi-Fi, Bluetooth v4.2 e Bluetooth Low Energy num módulo integrado a um poderoso microprocessador de 32 bits com dois núcleos. Além disso, a compatibilidade com o framework Arduino IDE é uma praticidade adicional, facilitando a elaboração, o teste e a depuração do código-fonte e permitindo a utilização de uma vasta gama de bibliotecas e tutoriais disponíveis.



Figura 3.1: Motot QF2822L



Figura 3.2: Driver de velocidade Hobbywing Skywalker V2.

Por se tratar de um protótipo rápido, a necessidade de construção de placas de circuito impresso para condicionamento dos pinos e demais funcionalidades de microcontroladores de uso mais profissional foi evitada pela escolha da placa NodeMCU32S [4]. Esse modelo apresenta placas de pinos para a montagem rápida de circuitos de teste usando protoboards, um conector micro-USB e um bootloader embutido para permitir a programação via USB facilmente, e ainda um regulador de tensão de 5V para 3,3V, o que permite que o módulo seja alimentado facilmente quando na montagem final por meio dos drivers de velocidade dos motores, que fornecem uma tensão regulada de 5V.

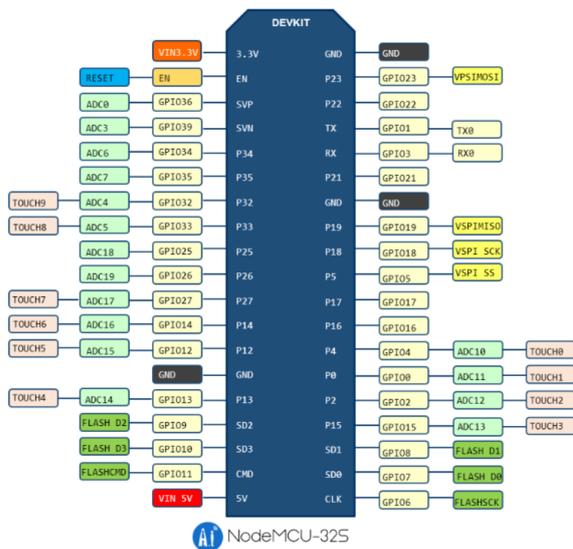


Figura 3.3: Mapa de pinos do NodeMCU-32S [4].

Figura 3.4: Módulo sensor IMU MPU9250 [8].

3.2.3 Sensores

A fim de permitir um sensoriamento completo de todas as variáveis de interesse apresentadas na seção 3.1, atendendo aos requisitos de redundância e checagem de plausibilidade dos dados obtidos, decidiu-se por implementar neste projeto todos os sensores descritos na seção 2.1 como alternativa à câmera de vídeo.

Conforme descrito, são necessários acelerômetro, giroscópio e magnetômetro. Para esse fim, escolheu-se o módulo IMU (*Inertial Measurement Unit*) MPU-9250 [8], que reúne esses três sensores em três eixos (x, y e z). Além de condicionamento de sinais integrado, esse módulo apresenta comunicação SPI e I2C, e é vendido em uma PCB com regulador de tensão de 5V incluso e fácil acesso às entradas por meio de uma barra de pinos (Figura 3.4). Basta configurar a taxa de aquisição e range de cada sinal para utilizar este módulo.

Adicionalmente, o módulo GPS BN-220 [2] é utilizado para localização espacial do *hovercraft*, bem como para medição de sua velocidade linear. Com comunicação serial, boa precisão e alimentação de 5V, trata-se também de uma solução de prática utilização, uma vez que não necessita de calibração. Sua utilização abre uma vasta gama de aplicações como o controle de trajetória.

Por fim, a utilização de um sensor óptico de deslocamento é uma forma elegante de medir a informação vetorial da direção e sentido do movimento do *hovercraft* no espaço, bem como a informação de velocidade vetorial em duas dimensões, obtida dividindo o deslocamento pelo período de amostragem utilizado. Sensores desse tipo podem ser encontrados em um mouse óptico, como o modelo MO-870 PS2 da marca *Braview*. Removendo todos os componentes não essenciais, isto é, deixando apenas o CI com o sensor óptico e um led que produz a iluminação da superfície necessária ao funcionamento, foi possível realizar testes com esse sensor e verificar a sua exatidão. Por se tratar de um mouse com conector do tipo PS2, ele possui comunicação serial de baixa latência e pode produzir leituras a uma frequência de $100Hz$.

No entanto, diversas dificuldades acabaram impossibilitando a utilização desse sensor no protótipo final. Em primeiro lugar, para realizar medições, o sensor óptico precisa deslizar sobre uma superfície regular e sem imperfeições, o que muitas vezes não acontece. Em segundo lugar, a elevação do *hovercraft* sobre sua saia interfere com a capacidade de medição do sensor óptico, pois a distância até a superfície torna-se muito grande. E, por último, o sensor utilizado não foi projetado para sofrer impactos e ser utilizado em superfícies empoeiradas, de forma que a falta de robustez dessa solução acabou contribuindo também para a decisão de abandonar a utilização desse sensor. Ainda assim, o código para sua utilização foi implementado no *firmware* do microcontrolador e o projeto das conexões necessárias para sua compatibilização de um ponto de vista de hardware também foi realizado e testado.

3.3 Esquema elétrico de montagem

O diagrama esquemático da montagem dos componentes eletrônicos do projeto é apresentado na Figura 3.5. A alimentação dos componentes provém de uma bateria de lítio (B1) de 2200mAh com tensão nominal de 14,8V, cujos polos positivo e negativo são indicados por VCC e VEE, respectivamente. Fazendo uma derivação em paralelo dos polos da bateria usando conectores XT60, obtém-se a alimentação de ambos os drivers de velocidade Hobbywing SkyWalker V2 40A. Os drivers, por sua vez, fornecem uma tensão regulada de 5V, que é utilizada para alimentar o microcontrolador e os sensores, bem como o servomotor. Os drivers alimentam as três fases de cada um dos motores QX-motor QF2822L(2222) por meio de conectores Banana Ease.

Alimentado pela tensão de 5V, o microcontrolador (U6) apresenta 3 saídas programadas como PWM (P5, P17 e P16), que são os sinais de controle para os drivers de velocidade e o servomotor. A comunicação com o IMU (U1) é estabelecida em protocolo SPI por meio dos

pinos P18 e P19, programados como SCL e SDA, respectivamente. Já o módulo GPS BN220 (U8) se comunica via protocolo I2C por meio dos pinos P14 e P12 do microcontrolador, programados como RX e TX, respectivamente.

Por fim, a comunicação com o sensor óptico do mouse evidencia a incompatibilidade dos níveis lógicos de tensão entre os pinos do microcontrolador (3,3V) e as entradas do sensor (5V). Para contornar este problema, um conversor de nível lógico bidirecional foi incluído entre os dois, de forma a compatibilizar a comunicação serial entre eles.

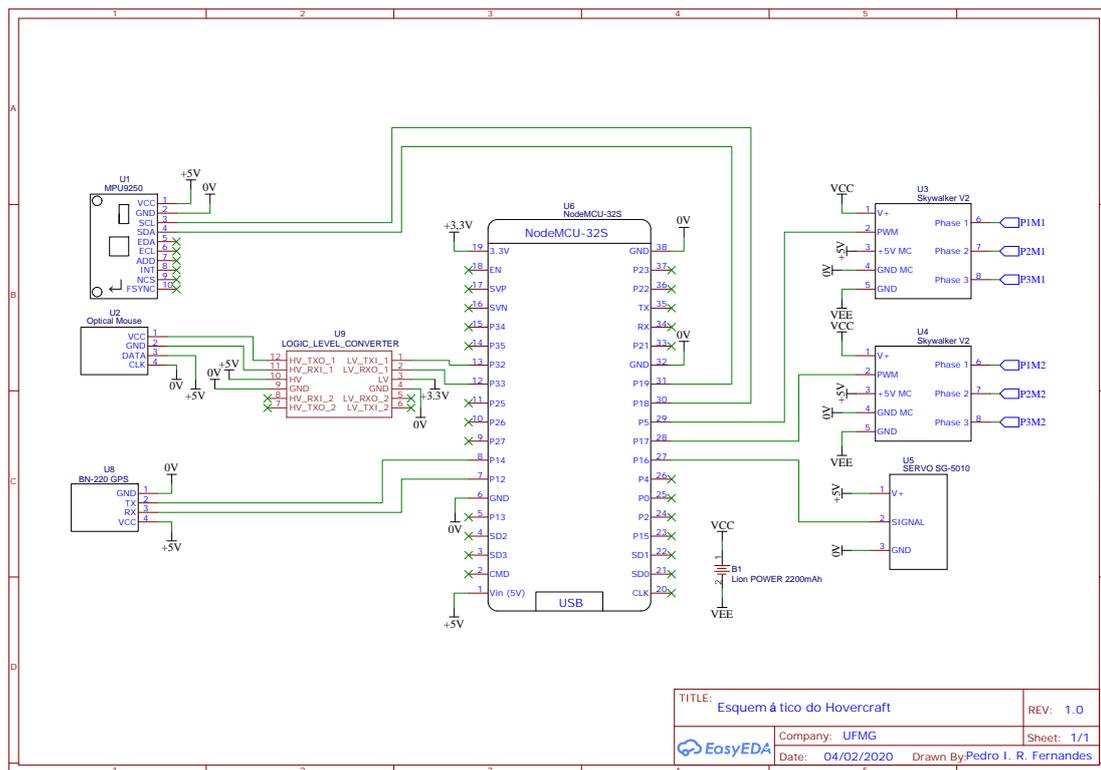


Figura 3.5: Diagrama esquemático de conexões entre os componentes eletrônicos.

Na Figura 3.6, é apresentado o diagrama de conexões utilizado na placa de circuito construída para este projeto. Como se pode perceber, os sensores GPS e óptico, bem como as conexões dos drivers de velocidade e do servomotor são substituídos por barras de pinos, mas as conexões permanecem as mesmas. O sensor IMU é soldado diretamente na placa.

Para este protótipo, a placa de circuito foi construída usando-se uma placa perfurada e soldando-se manualmente as conexões, o que agilizou e barateou o processo de confecção, porém a melhor solução seria confeccionar uma PCB para este fim. Sendo assim, foi adicionado a este projeto o desenho de uma proposta de PCB (Figura 3.7) que poderia substituir a placa feita à mão.

3.4 Projeto 3D da estrutura

Para o projeto da estrutura, foi utilizado o software Fusion 360, uma ferramenta de modelagem e projeto 3D da Autodesk com licença gratuita para estudantes. Dentro dessa ferramenta, o primeiro passo para a elaboração da estrutura é a modelagem prévia dos motores e

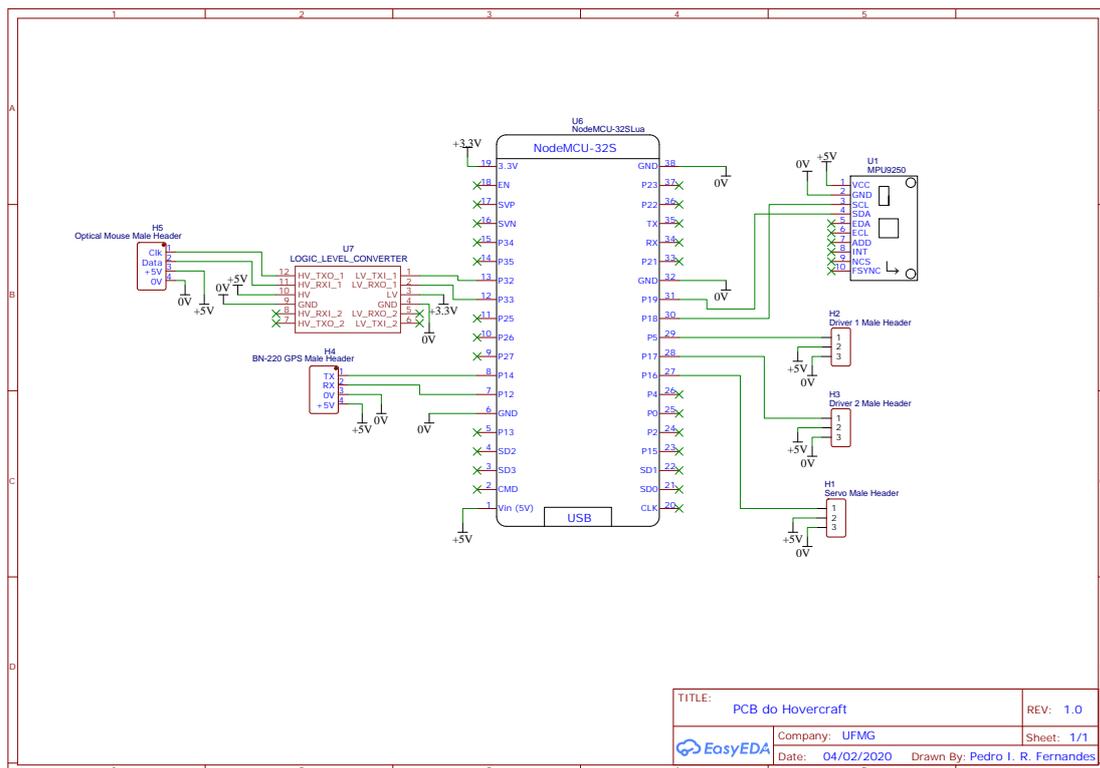


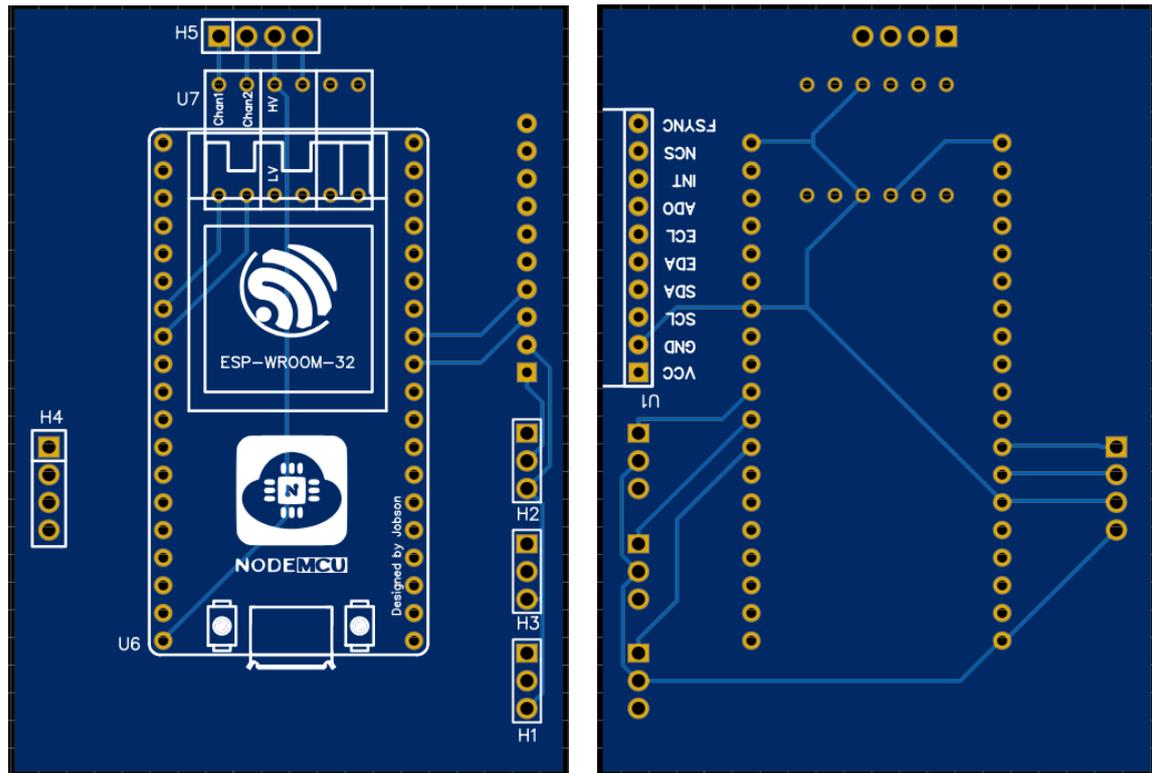
Figura 3.6: Diagrama das conexões da placa de circuito do Hovercraft.

do servomotor para auxiliar no dimensionamento (Figura 3.8). Em seguida, foi planejado o posicionamento dos demais componentes como bateria e sensores, bem como das conexões elétricas. Por fim, a partir de *insights* de um projeto acadêmico [12], foi realizado o *design* da estrutura do *hovercraft* (Figura 3.10).

A estrutura física do protótipo possui 25 centímetros de comprimento por 20 centímetros de largura, e sua altura após a montagem de todos os componentes é de cerca de 15 centímetros. A estrutura vazada na parte inferior (Figura 3.10) é preenchida pelo ar movimentado pela turbina de *lift* (Figura 3.1) através da cavidade no centro da superfície superior da estrutura. O ar em alta pressão no interior da estrutura infla a saia plástica fixada em seu entorno, mostrada na Figura 3.12. Para permitir a saída de ar pela parte inferior da estrutura, foram colocadas fendas na base da estrutura (Figura 3.11), provendo sustentação e estabilidade do *Hovercraft*.

3.5 Firmware do Microcontrolador

Conforme afirmado anteriormente, a escolha pela utilização do *framework Arduino* introduz diversas vantagens, como a disponibilidade de uma vasta gama de bibliotecas com suporte a diversas funcionalidades, monitor serial para depuração do código e suporte para *flash over-the-air* diretamente do programa. Já a escolha pelo microcontrolador ESP32, com seu processador de dois núcleos com suporte para o sistema operacional de tempo real para microcontroladores *freeRTOS*, oferece ainda mais praticidade, permitindo dividir as funcionalidades em tarefas que serão processadas de maneira paralela e inserir restrições de tempo



(a) Vista superior;

(b) Vista inferior;

Figura 3.7: Placa de circuito impresso para o Hovercraft.

real que provêm maior robustez ao *firmware*.

O primeiro passo para a elaboração do *firmware* é a obtenção de bibliotecas para implementar suas funcionalidades de comunicação com os periféricos, assim como com o computador, utilizando o protocolo TCP-IP. As bibliotecas utilizadas e suas respectivas funções são destacadas a seguir:

- TinyGPS++: dado que a comunicação com módulos GPS segue um protocolo padrão, foi escolhida esta para a comunicação com o módulo BN-220.
- MPU9250: implementa a comunicação com o Inertial Measurement Unit (IMU), modelo MPU-9250, por meio dos protocolos SPI e I2C, incluindo rotinas de comando de aquisição e de leitura de dados.
- espsoftwareserial e PS2Mouse: biblioteca de implementação da comunicação serial gerenciada por software específica para ESP e biblioteca para comunicação serial com o mouse, configuração de sensibilidade, aquisição e leitura de dados.
- Servo_ESP32: biblioteca para comando de servomotores por meio de sinais PWM. Além disso, também pôde ser utilizada para realizar o comando dos drivers de velocidade dos motores *brushless*.
- ArduinoOTA: Biblioteca para a atualização remota do *firmware* do microcontrolador por meio de uma rede *wi-fi* (*flash over-the-air*).

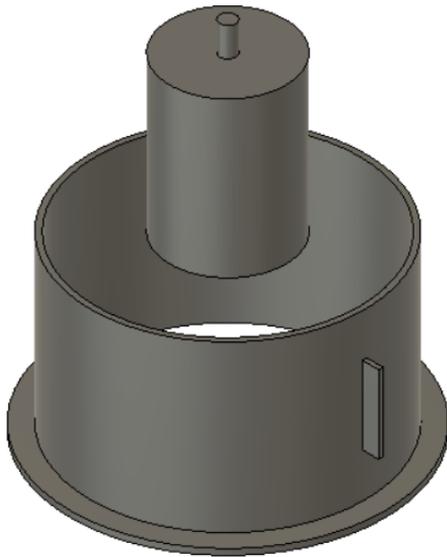


Figura 3.8: Modelo 3D do QX-motor.

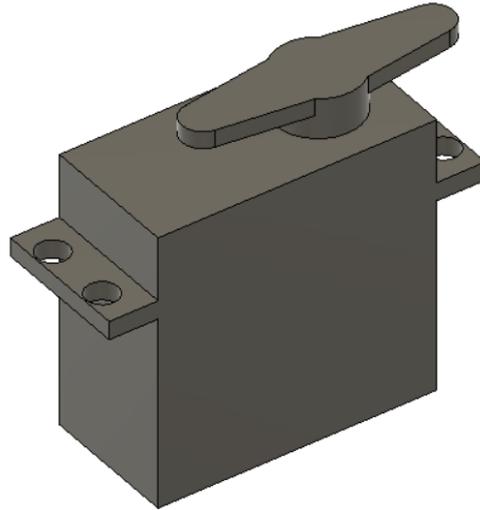


Figura 3.9: Modelo 3D do servo SG-5010.

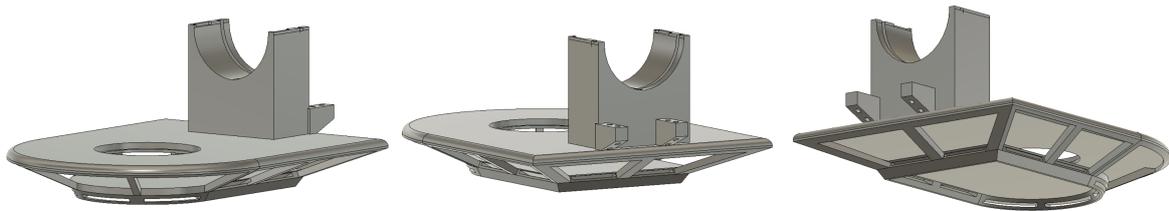


Figura 3.10: Diversas vistas do projeto 3D do Hovercraft.

- *lwip/sockets*: biblioteca que implementa o socket para realizar a função de servidor WEB via protocolo TCP-IP, permitindo assim a comunicação com a interface de usuário implementada no computador (cliente).

Uma vez incluídas as bibliotecas e implementadas as funções básicas do *firmware*, o próximo passo é a estruturação das funcionalidades em tarefas, a fim de dividi-las em fluxos independentes de código que serão processados paralelamente pelos núcleos do processador. Um paradigma interessante para realizar essa divisão é separar tematicamente as funcionalidades, a fim de reduzir o número de variáveis compartilhadas entre as tarefas, as quais terão que ser geridas por mecanismos de exclusão mútua. Esse tipo de mecanismo é responsável por administrar qual tarefa deve ter acesso às variáveis compartilhadas num determinado momento, bloqueando o acesso pela outra tarefa.

Nesse caso, a decisão de projeto foi agrupar as funcionalidades de leitura dos sensores, o processamento dos dados, que inclui a rotina de controle, e o envio de comando aos atuadores por meio de sinais PWM. Essa decisão facilita o controle sobre o tempo de execução desse fluxo em *loop* leitura-processamento-atuação, permitindo atribuir a ele uma restrição de tempo real usando comandos nativos do sistema operacional *freeRTOS*.

Enquanto isso, o outro núcleo realiza as tarefas de comunicação, funcionando como servidor WEB, por meio do protocolo TCP-IP. Essa tarefa é responsável por escanear constantemente a rede esperando pela conexão do cliente. Quando um cliente se conecta, o

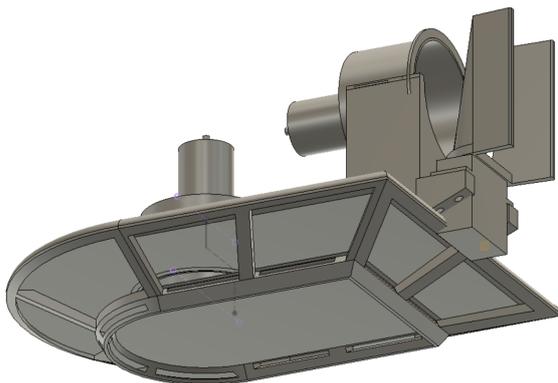


Figura 3.11: Modelo 3D do protótipo com todas os componentes utilizados.

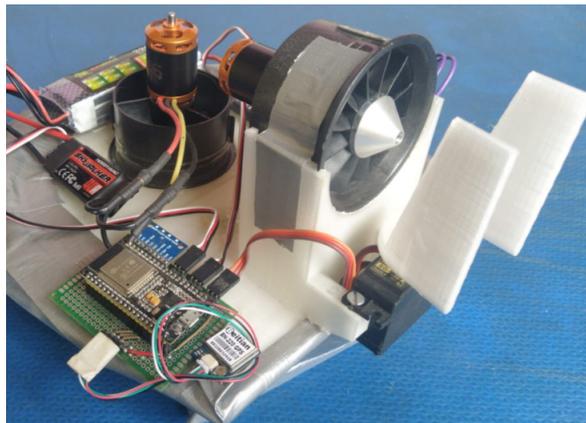


Figura 3.12: Foto do protótipo construído.

microcontrolador recebe dele um valor que representa um de quatro possíveis modos de funcionamento que podem ser definidos pelo usuário na interface:

- Calibração;
- manual;
- automático;
- *flash over-the-air*.

Nos três primeiros modos, o microcontrolador envia para a interface os últimos dados lidos dos sensores. No modo manual, ele recebe do cliente sinais de *duty-cycle* para os motores e ângulo desejado para o servomotor. Já no modo automático, ele recebe *setpoints* de velocidade linear e angular, com os quais será alimentada a malha de controle. Quando colocado em modo *flash over-the-air*, o microcontrolador interrompe a conexão com o cliente, fecha a *socket* de escuta, bem como a tarefa do outro núcleo e aguarda a requisição de atualização do *firmware*, que pode ser feita pelo próprio *framework* do *Arduino*.

3.6 Calibração do *Inertial Measurement Unit* (IMU)

Conforme apresentado durante a escolha dos sensores, o *Inertial Measurement Unit*, ou IMU, é um módulo periférico composto de nove sensores, sendo três acelerômetros, três giroscópios e três magnetômetros, um de cada para cada dimensão espacial. O módulo escolhido, o MPU-9250, inclui ainda um microprocessador capaz de realizar medições com estes sensores, fazer correções, ajustes de sensibilidade e comunicar-se por protocolo I2C ou SPI.

As variáveis medidas por cada um destes sensores são: aceleração linear inercial, com o acelerômetro; velocidade angular, com o giroscópio; e a variação da força do campo magnético, com o magnetômetro. No entanto, uma gama muito mais vasta de informações pode ser inferida a partir do conjunto das medições de um IMU [3]. Por exemplo, combinando o acelerômetro com o magnetômetro, é possível determinar a direção e o sentido de movimento de um corpo (*heading*). Combinando o giroscópio com o acelerômetro, é possível obter os ângulos de rotação de um corpo em relação a um *frame* de referência (*pitch*, *roll*, *yaw*) [1].

No entanto, para obter medições confiáveis e precisas com esse tipo de módulo, é necessário realizar ajustes e calibrações, mitigando imprecisões intrínsecas do instrumento e do ambiente no qual será utilizado.

3.6.1 Efeito de ângulo de elevação e de margem

Os ângulos de elevação e margem são definidos como os ângulos formados entre os eixos x e y do sensor, respectivamente, e a superfície (neste caso, o chão). Posicionando o IMU no *hovercraft*, é possível verificar estes ângulos utilizando o fato de que, quando em repouso, a aceleração da gravidade deveria ser medida no sentido positivo do eixo z , por convenção. Aplicando rotações destes dois ângulos às medições, obtém-se um realinhamento entre o plano do sensor e o plano xy de referência (o solo), de forma a corrigir as medições realizadas [6].

3.6.2 Distorções do Campo magnético

As distorções do campo magnético terrestre são resultado de influências magnéticas externas geralmente chamadas de Hard-Iron e Soft-Iron Effects. Efeitos desse tipo podem ser observados na presença de objetos de ferro com uma massa significativa nas proximidades de um magnetômetro, ou ainda pela presença de fortes correntes elétricas. Se nenhum efeito de distorção estiver presente, ao girar o sensor em 360° e traçar os dados resultantes em duas dimensões (x e y) resultará em um círculo centrado em $(0, 0)$ cujo raio deve ser a intensidade do campo magnético terrestre na região. No entanto, a presença desses efeitos pode produzir uma distorção no traçado da circunferência, produzindo um deslocamento de seu centro no caso do Hard-Iron Effect, ou mesmo a sua deformação no caso do Soft-Iron effect [6].

Os Soft-Iron Effects são, em geral, causados por correntes nas proximidades do sensor que produzem aumento ou diminuição da intensidade do campo magnético resultante. Seu efeito, no entanto, pode ser mitigado evitando-se a utilização do sensor em ambientes fechados e de grande atividade elétrica. Além disso, uma vez que a informação de orientação proveniente do magnetômetro está fortemente associada ao sentido do vetor magnético, e não a sua intensidade, esse efeito tem menor impacto sobre as medições. Dessa forma, acabou não sendo necessário, ou mesmo prático, incluir a correção desta distorção na calibração do instrumento para esta aplicação [6].

O efeito Hard-Iron, por outro lado, tem grande impacto sobre a utilização do magnetômetro, pois está fortemente associado à presença de materiais ferromagnéticos fortes na proximidade do sensor, o que é, muitas vezes, inevitável. Além disso, o deslocamento do centro do círculo gerado pelos dados do magnetômetro pode muitas vezes fazer com que seu vetor resultante aponte sempre na mesma direção, o que inutilizaria suas medições para o fim de determinar orientação. Dessa forma, torna-se essencial implementar um método para realizar esta correção [6].

Em termos práticos, o método consiste em posicionar o magnetômetro sobre a superfície de referência e girá-lo lentamente em pelo menos 360° , a fim de realizar medições. Em seguida, plotando os dados obtidos, verifica-se um formato próximo de um círculo em torno de um centro de coordenadas (x,y) . Para realizar a correção, basta subtrair os valores do centro do círculo (ou elipse) das medições. Para se obter estes valores, no entanto, uma forma prática é calcular a média das componentes x e y de todo o conjunto de medições. No modo

calibração, foi implementado na interface gráfica um gráfico para facilitar a realização desse ensaio, plotando os dados em tempo real num *scatter plot*. Após realizar o procedimento de giro do protótipo, o usuário pode interromper o ensaio, e o programa gerará automaticamente os valores de correção que devem ser aplicados às medições do magnetômetro.

3.7 Interface gráfica de usuário (GUI) em MATLAB

A escolha pela utilização da plataforma do MATLAB para a interface gráfica de usuário se baseou principalmente no fato de que, assim como no caso do Arduino, o estudante de engenharia de controle apresenta, em geral, maior familiaridade com este framework para a realização de modelagem de sistemas dinâmicos, assim como para o projeto de controladores. A fim de incentivar ainda mais a utilização dos recursos do MATLAB e SimuLink, foi implementada a função "salvar dados", que exporta os dados obtidos no ensaio diretamente para o workspace do programa (figura 3.13), oferecendo praticidade para a utilização de seus recursos pelo aluno.

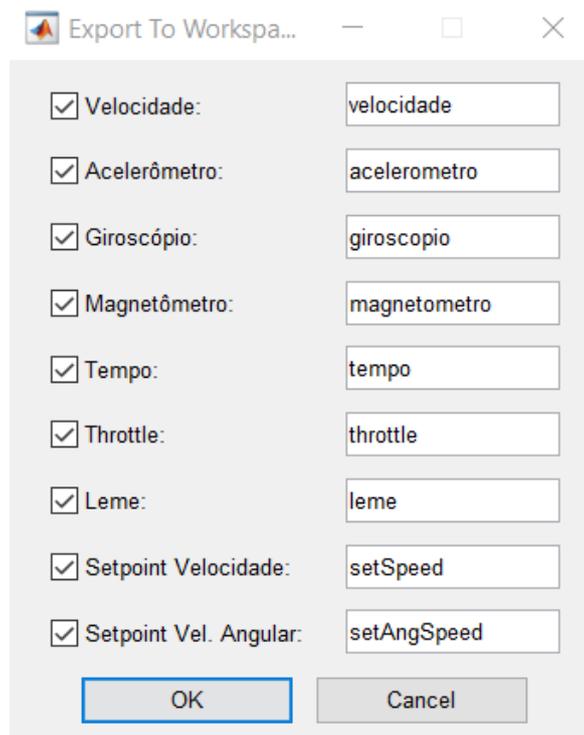


Figura 3.13: Janela para salvar os dados obtidos no ensaio.

Explorando inicialmente a interface, observa-se no lado esquerdo a inserção de elementos de interface ligados à comunicação com o protótipo, bem como o comando de "salvar dados", descrito anteriormente. Do lado direito, a interface é subdividida em abas específicas para cada etapa de desenvolvimento do projeto de controle (Calibração, Manual e Automático). A escolha por essa divisão em abas é proposital e deve ser entendida como uma recurso didático que pretende levar o aluno a uma compreensão implícita da sequência de etapas do projeto de controle:

1. calibração dos elementos primários e finais do processo;

2. ensaios de identificação da planta;
3. modelagem e validação do seu comportamento dinâmico;
4. projeto e síntese do controlador;
5. ensaios e validação do controlador e do processo controlado.

Na primeira aba, denominada "Calibração"(Figura 3.14), o aluno realizará o procedimento descrito na seção 3.6.2. Ao parar a leitura de dados, serão impressos na tela os offsets em x e y para correção das leituras do magnetômetro.

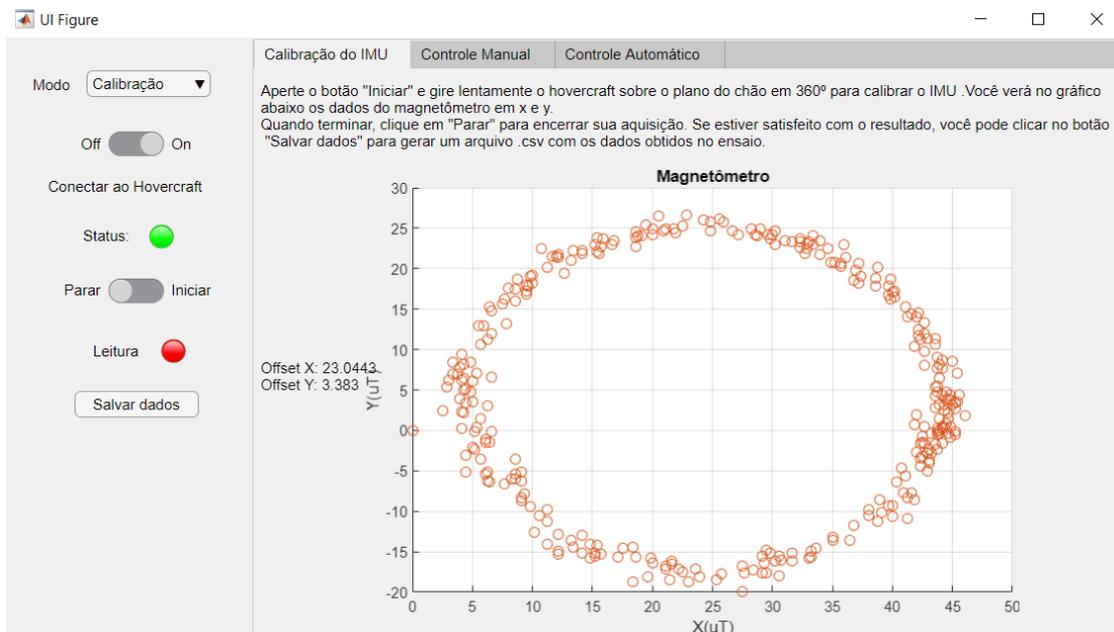


Figura 3.14: Aba para calibração do magnetômetro.

Na segunda aba, denominada "Controle Manual"(Figura 3.15), o usuário tem acesso aos controles dos motores, que só serão enviados ao *hovercraft* se o modo selecionado for o modo manual. Os motores de sustentação ("*Lift*") e de propulsão ("*Throttle*") são controlados enviando um valor de 0 a 100% que representa o *duty-cycle* dos sinais PWM enviados aos *drivers* de velocidade. Já o servomotor que controla o leme é comandado por um valor entre -90° e 90° . O botão denominado "RESET" zera esses três comandos, permitindo a desativação dos atuadores do hovercraft. Ainda nesta aba, podem ser visualizados em tempo real os gráficos com os sinais enviados aos dois atuadores principais (throttle e leme), bem como a evolução das variáveis de saída (velocidade linear e angular) no tempo. Essas representações gráficas também devem ser entendidas como dispositivos didáticos, pois indicam implicitamente as relações entre as variáveis controladas e de processo, que devem ser utilizadas para elaboração da modelagem e dos controladores. Fica também subentendido que o motor de lift não deve ser utilizado como variável de processo, uma vez que se trata de um sistema *sobreatuado*.

Já na terceira aba, denominada "Controle Automático"("Figura 3.16"), o aluno tem acesso a dois *setpoints*: a velocidade linear, dada em m/s, e a velocidade angular, expressa em $^\circ/s$. Observa-se novamente a escolha proposital pela ausência de qualquer comando relacionado

3.8. PARADIGMAS PARA A MODELAGEM DO COMPORTAMENTO DINÂMICO DO HOVERCRAFT

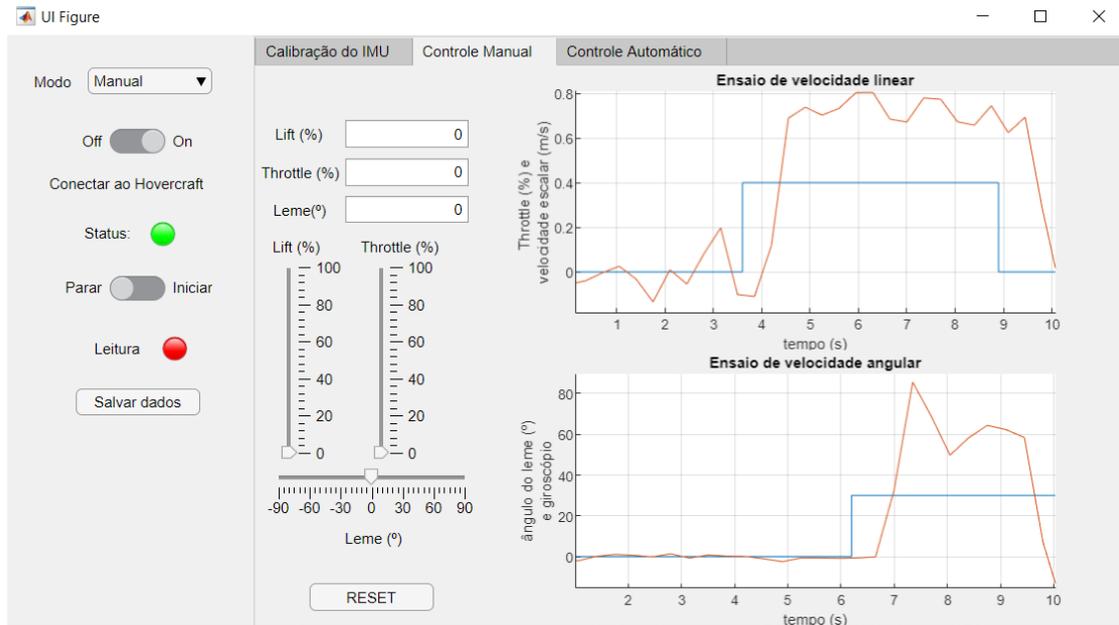


Figura 3.15: Aba para realização de ensaios em malha aberta com o protótipo.

aos atuadores. O aluno é induzido aqui à compreensão de que os controladores projetados devem fornecer, a partir dos I, os valores para os três atuadores. São visíveis ainda os gráficos para exibição da interação entre os *setpoints* e as variáveis controladas.

3.8 Paradigmas para a modelagem do comportamento dinâmico do hovercraft

Para iniciar o processo de modelagem dinâmica do *hovercraft*, é importante revisar os objetivos de controle propostos, bem como os recursos de controle disponíveis. Neste protótipo, se propõe a realizar um controle independente de velocidades linear e angular. Para atingir isso, se dispõe de dois motores *brushless*, um para sustentação do protótipo sobre um bolsão de ar, que será chamado daqui em diante de motor de *lift*, e um para propulsão horizontal do hovercraft, que será chamado de motor de *throttle*. Além disso, o *hovercraft* conta com uma asa, ou leme, posicionada na direção do fluxo de ar do motor de *throttle*, por meio da qual é possível direcionar o fluxo de ar produzido pela turbina de *throttle*.

Dessa forma, ficam estabelecidas as duas variáveis controladas (velocidade linear e angular), bem como as três variáveis manipuladas (potência no motor de *lift*, no motor de *throttle*, e ângulo do servomotor que movimenta o leme). Trata-se, portanto, de um sistema MIMO com três entradas e duas saídas, ou seja, um sistema sobre-atuado. Assim, faz-se necessário em primeiro lugar decidir qual variável será utilizada para o controle de cada saída.

De maneira geral, o procedimento formal utilizado para fazer o pareamento de variáveis demanda a elaboração de uma matriz de correlação entre elas. No entanto, neste caso específico, esse procedimento não se faz necessário tendo em vista as seguintes observações práticas acerca do comportamento do protótipo:

- o motor de *lift* não produz impacto significativo sobre nenhuma das saídas de forma

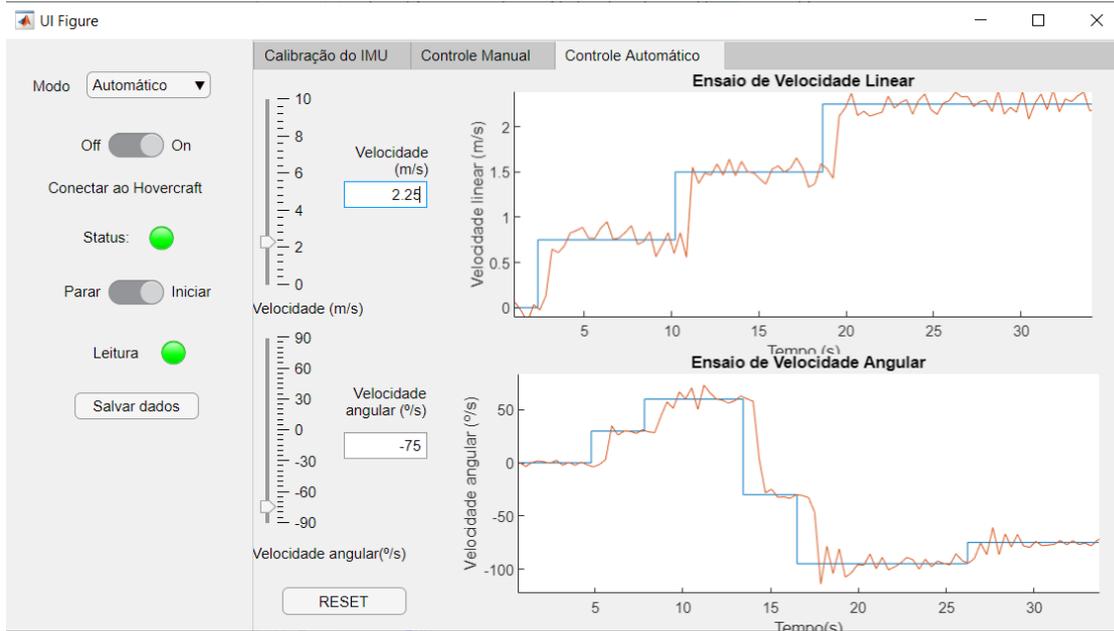


Figura 3.16: Aba para realização de ensaios em malha fechada com o protótipo.

independente, apenas amplifica ou atenua o efeito realizado pelos outros atuadores;

- o motor de *throttle* apresenta impacto predominantemente sobre a velocidade linear, não produzindo efeito significativo sobre a velocidade angular sem a atuação do leme;
- o leme apresenta impacto predominantemente sobre a velocidade angular; no entanto, a intensidade desse efeito é fortemente acoplada ao comando aplicado aos outros motores.

Sendo assim, a escolha de pareamento mais simples para um propósito didático é utilizar o motor de *throttle* para controle da velocidade linear, considerando o motor de *lift* como uma perturbação sobre o seu ganho estático. Analogamente, deve-se utilizar o leme para controle da velocidade angular, considerando os outros motores como perturbações sobre o seu ganho estático. Essas escolhas levam ainda em consideração as limitações do conhecimento da maioria dos estudante de controle acerca de sistemas MIMO, permitindo o tratamento das variáveis de saída como sistemas SISO independentes.

3.9 Ensaios e modelagem da velocidade linear

Seguindo essa abordagem de tratar o sistema MIMO como sistemas SISO independentes, decorre que a metodologia de modelagem deve primeiro abordar o comportamento do sistema em movimento retilíneo, levando em consideração a influência dos motores de *lift* e *throttle*, enquanto o leme não é acionado. Mais especificamente, a influência do motor de *throttle* sobre a velocidade linear deve ser mensurada sob diferentes níveis de atuação do motor de *lift*.

Nesse sentido, a estratégia escolhida para identificação da planta com respeito à velocidade linear consiste na realização de ensaios de resposta ao degrau. Degraus de pequena

amplitude (10%) são aplicados ao motor de *throttle* sob diferentes intensidades de atuação do motor de *lift* (25, 50 e 75%), a fim de modelar o comportamento tanto em regime permanente quanto transitório.

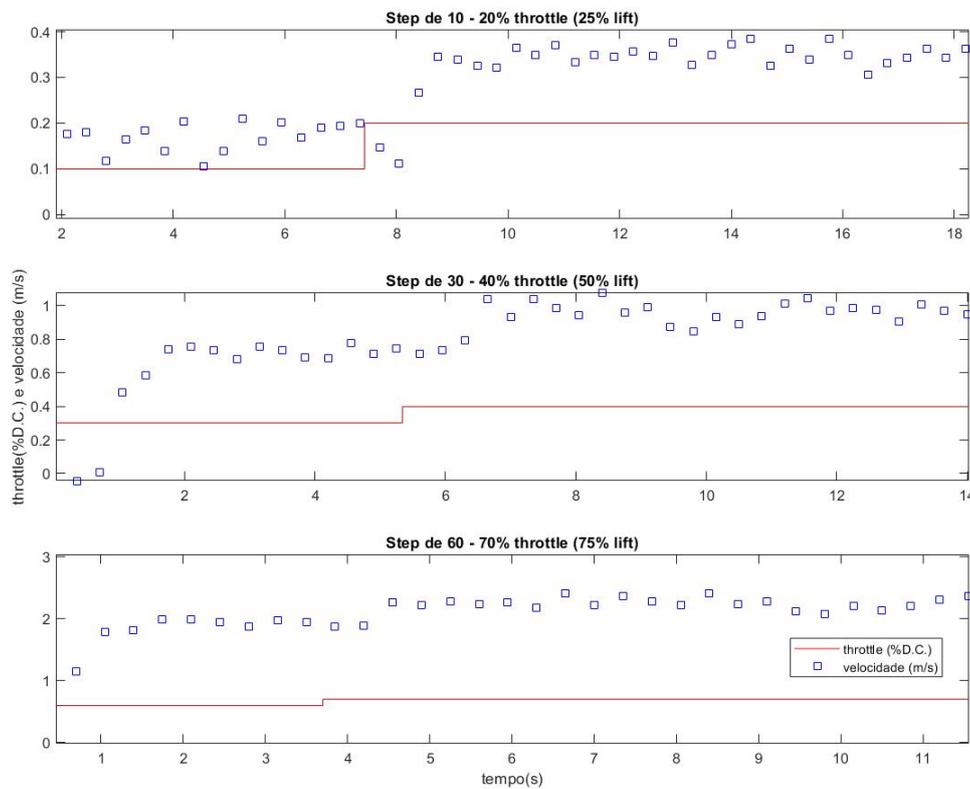


Figura 3.17: Três ensaios de resposta ao degrau de 10% de Duty-Cycle.

Três desses ensaios são apresentados na figura 3.17 para visualização. Pode-se observar que a medição de velocidade linear apresenta ruído significativo, em especial em baixas velocidades. Isso se justifica pelo fato de que, não sendo possível a implementação do sensor óptico para medição dessa grandeza, conforme explorado na seção 3.2.3, a solução alternativa mais simples foi utilizar a medição de velocidade provinda do GPS. A discussão sobre esse ponto, bem como a descrição das medidas tomadas para tratar seus efeitos é apresentada na seção 3.10.

A partir das formas de onda da velocidade na figura 3.17, observa-se que o sistema apresenta dinâmica transitória rápida com um pequeno atraso. Dessa forma, é possível aproximar a sua resposta por um sistema de primeira-ordem mais atraso, de forma dada por

$$P_v(s) = e^{-\theta s} \cdot \frac{K_v}{\tau s + 1}$$

em que θ é o valor do atraso no tempo em segundos, τ é a constante de tempo do sistema em segundos e K_v é o ganho de regime permanente do sistema. Esse atraso observado é consistente com o valor médio da latência de comunicação entre o microcontrolador e o computador, acrescido do tempo de processamento do microcontrolador.

O valor de θ pode ser obtido calculando-se o atraso médio de todos os ensaios de resposta ao degrau realizados ($\theta = 0,368s$). Já a constante de tempo do sistema foi obtida a partir de seu tempo de acomodação médio ($t_s = 4\tau$), obtido graficamente dos ensaios. Calculando-se a média dos tempos de acomodação, obteve-se $\tau = 0,281s$.

Com relação ao ganho K_v , ele é em geral obtido dividindo-se a variação de regime permanente na saída pelo amplitude do degrau aplicado na entrada. No entanto, a fim de verificar a linearidade do ganho, é necessário plotar os diversos valores de regime permanente num gráfico saída-entrada e realizar a regressão linear para obter o valor do ganho (coeficiente de primeira ordem da função de regressão $f(x) = ax + b$) com maior confiabilidade. Esse procedimento foi realizado para as três condições do motor de lift testadas e os resultados obtidos são apresentados na Figura 3.18.

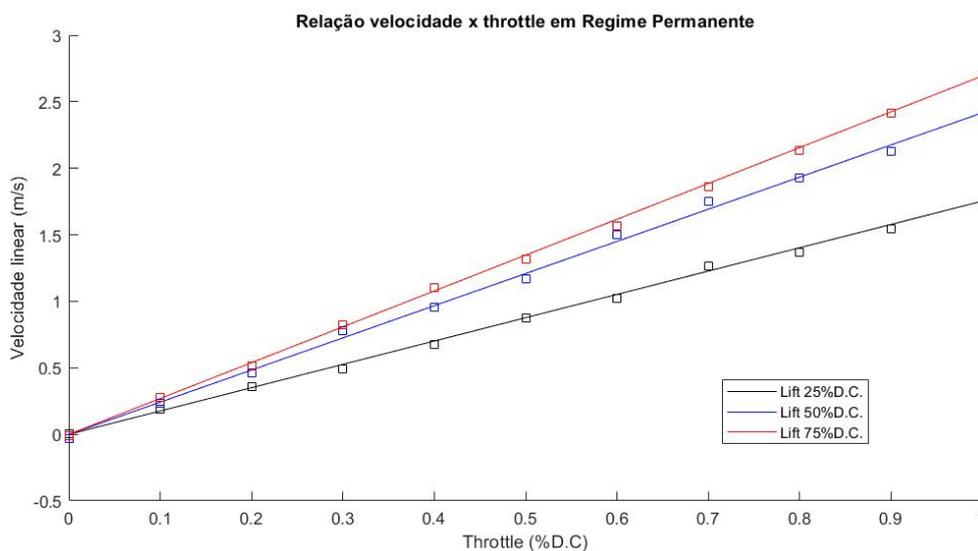


Figura 3.18: Velocidade em função do *throttle* sob distintas condições de *lift*.

Pode-se observar que o ganho aumenta com o *duty-cycle* aplicado ao motor de *lift*. Por regressão foram obtidos os valores $K_v(25\%) = 1,753$, $K_v(50\%) = 2,417$ e $K_v(75\%) = 2,694$. É interessante observar que, o valor de *duty-cycle* de *throttle* é representado por valores entre 0 e 1. Assim, é possível empregar uma estratégia de ganho escalonado no projeto do controlador de velocidade linear do sistema.

Com isso, chega-se ao seguinte modelo de velocidade linear em função do *Duty-cycle* de *throttle* no domínio s :

$$P_v(s) = e^{-0,368s} \cdot \frac{K_v(lift)}{0,281s + 1}$$

De posse desse modelo no domínio s , é possível aplicar novamente os sinais de entrada ao sistema e comparar o seu comportamento dinâmico com aquele do modelo levantado. O resultado dessa comparação para três condições de ensaio é apresentado na figura 3.19. Como se pode observar, apesar do forte ruído na leitura de velocidade, tem-se um modelo bem ajustado em regime transitório e permanente.

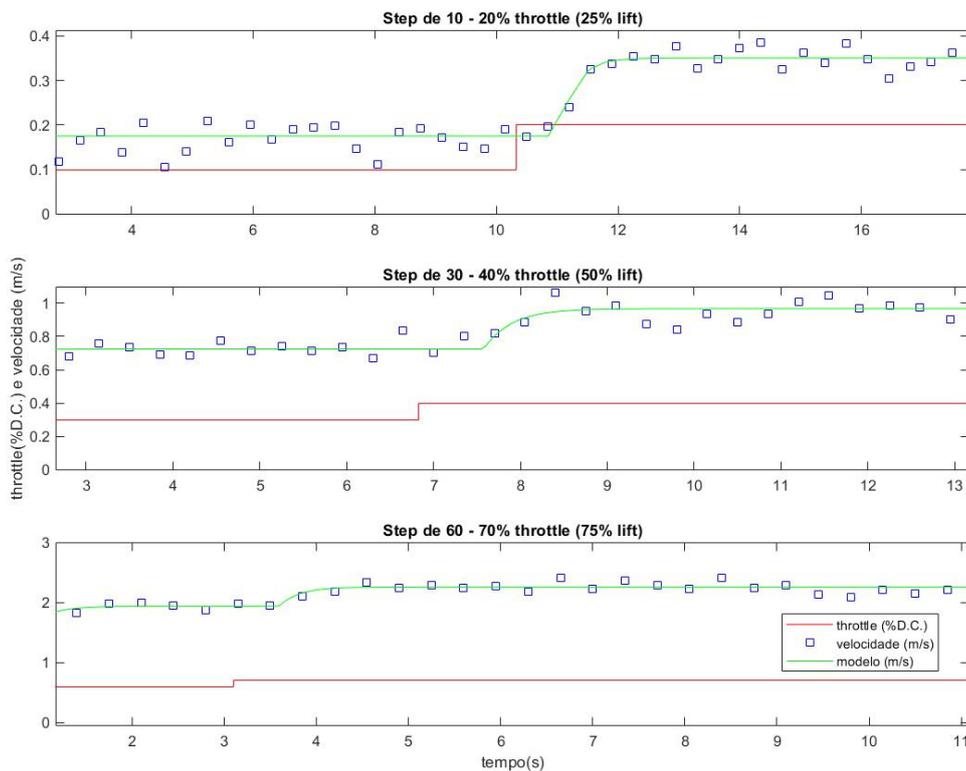


Figura 3.19: Validação da modelagem de velocidade linear.

3.10 Filtragem de velocidade linear

Além de um erro de até $\pm 0,1 m/s$ [2], o módulo GPS produz apenas o valor escalar da velocidade, o que limita a gama de aplicabilidade de suas medições. A escolha pela sua utilização decorre da incapacidade de embarcar um sensor óptico na superfície inferior do protótipo para medição da velocidade em duas dimensões. Outra alternativa explorada foi integrar numericamente as leituras do acelerômetro no tempo, obtendo assim a velocidade em duas dimensões. No entanto, essa opção foi descartada, uma vez que acarreta um acúmulo gradativo de erro de velocidade ao longo do tempo.

Definida a utilização do BN-220 para medição de velocidade, é necessário tomar medidas para mitigar os efeitos de sua baixa exatidão, em especial em baixas velocidades. Uma solução simples para isso é implementar um filtro de média-móvel no *firmware* do microcontrolador para reduzir a intensidade do ruído.

O filtro média-móvel é um filtro digital aplicável em situações em que há ruídos aleatórios de alta amplitude e curta duração. Seu objetivo principal é atenuar a amplitude de ruído, mantendo transições agudas. Em termos práticos, consiste em tomar o valor médio de uma sequência de N amostras de um sinal. À medida que novas amostras são coletadas, o filtro desloca a média para o conjunto mais recente de N amostras para gerar novos valores de saída. A equação que representa a saída y do filtro em função da sequência de entradas x é apresentada a seguir:

$$y[i] = \frac{1}{N} \cdot \sum_{j=0}^{N-1} x[i-j]$$

Uma vez que a rotina de aquisição dos sensores tem tempo de amostragem fixado em 100ms, e sabe-se que a latência da transmissão de dados via Wi-fi é de cerca de 350ms, a escolha do valor $N = 3$ traz uma suavização à leitura de velocidade, evitando que a inserção do filtro acrescente atraso ao sistema, como pode ser observado na figura 3.20 em que se compararam as leituras na condição mais crítica observada. Foi verificada uma redução da amplitude visualizada de erro, de cerca de 0,12m/s para 0,055m/s.

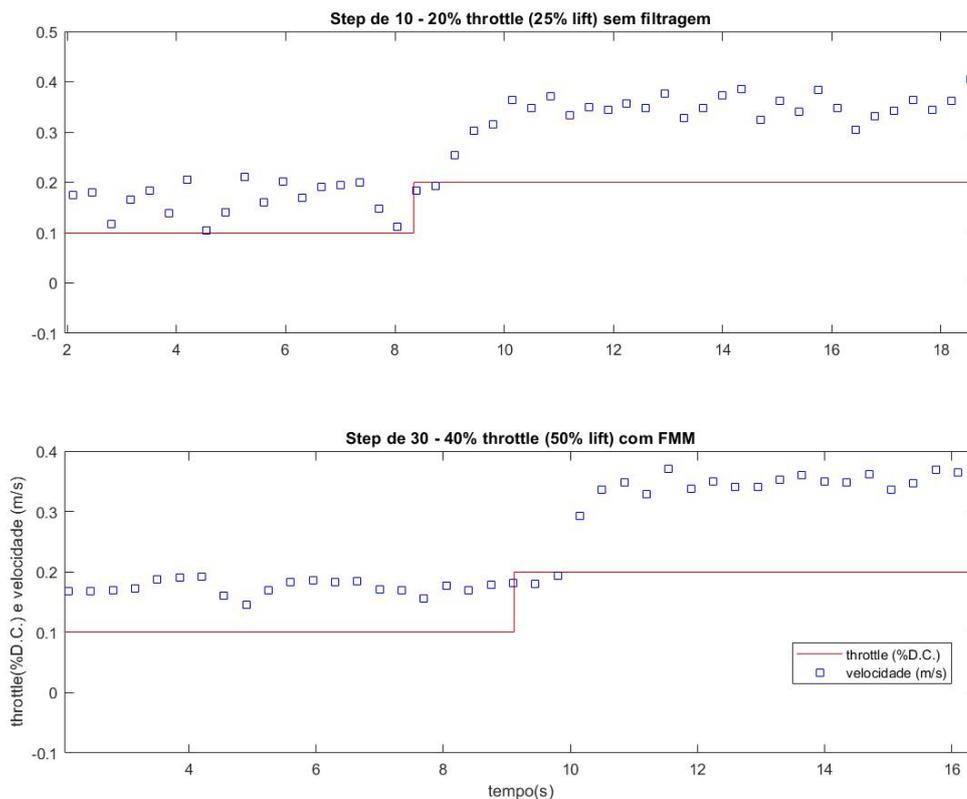


Figura 3.20: Comparação entre os sinais lidos pela interface antes e depois da adição do filtro média-móvel.

3.11 Ensaios e modelagem da velocidade angular

Seguindo a abordagem de tratar o sistema MIMO como sistemas SISO independentes, para o controle de velocidade angular é utilizada a angulação do leme controlado pelo servomotor como variável manipulada e a medição da velocidade angular do protótipo por meio da componente "z" lida do giroscópio, medida em graus por segundo, como variável controlada.

Primeiramente, são realizados testes qualitativos, a fim de observar características do comportamento dinâmico do sistema. Para isso, degraus de angulação do leme são aplicados

sob *throttle* e *lift* constantes em variados pontos de operação. Desses testes, observa-se que a velocidade angular do protótipo apresenta acentuado comportamento oscilatório, em especial em condições de *throttle* elevado. Observa-se ainda que a intensidade do *throttle* aplicado apresenta significativo impacto sobre o ganho na velocidade angular, enquanto a intensidade de *lift* apresenta um efeito bem mais brando. Sendo assim, por simplicidade de modelagem, decide-se considerar apenas o acoplamento entre *throttle* e velocidade angular na modelagem desta variável.

Em seguida, é definido o escopo e a característica dos ensaios a serem realizados para identificação deste sistema. O atuador do leme apresenta uma limitação de resolução de 5° em seu movimento, de forma que são escolhidos degraus de 15° de amplitude. Como o objetivo dos controladores que são projetados é operar com *setpoints* predominantemente constantes, serão realizados ensaios de resposta ao degrau de angulação do leme sob condições de *throttle* constante. Sendo assim, analogamente aos ensaios realizados para velocidade linear, são realizados ensaios de resposta ao degrau de 15° de 0 a 90° , com *throttle* em 25, 50 e 75%, cujos dados devem ser analisados em regime permanente e transitório. Os resultados de três desses ensaios são apresentados na Figura 3.21 para visualização do comportamento em diferentes condições de *throttle*.

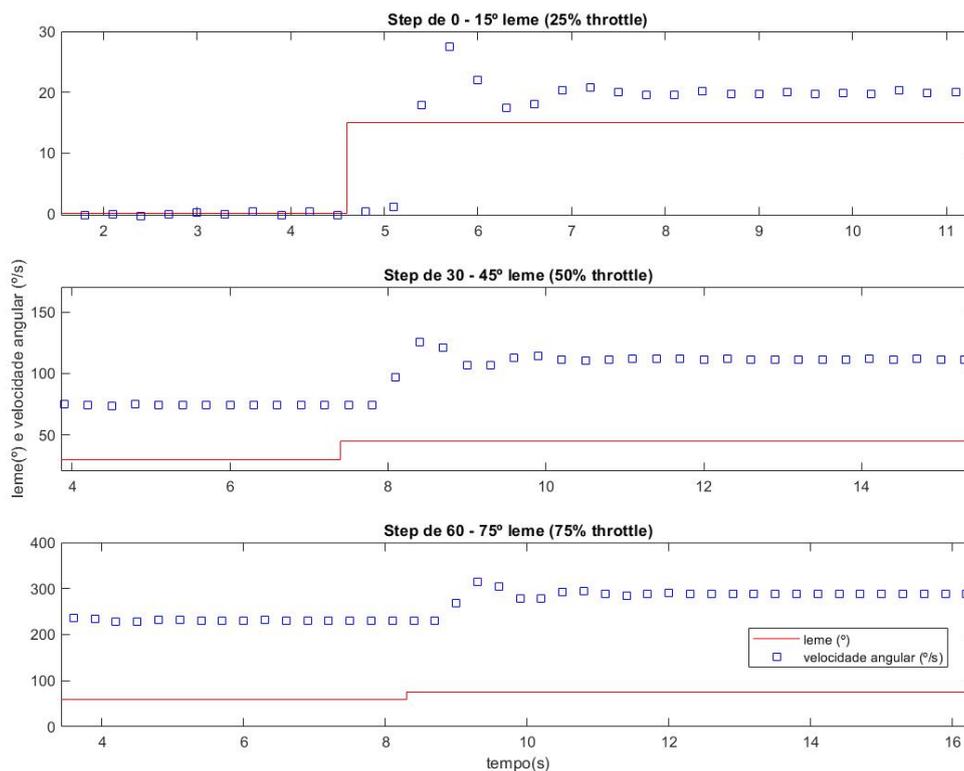


Figura 3.21: Três ensaios de resposta ao degrau de 15° de amplitude.

Observa-se um atraso de transporte similar ao identificado na velocidade linear, cuja origem foi discutida na seção 3.9. Conforme observado qualitativamente, o sistema apresenta comportamento oscilatório com baixa constante de amortecimento. Por essas razões, decide-se por modelá-lo como um sistema de segunda ordem mais atraso. Observando-se, ainda, o

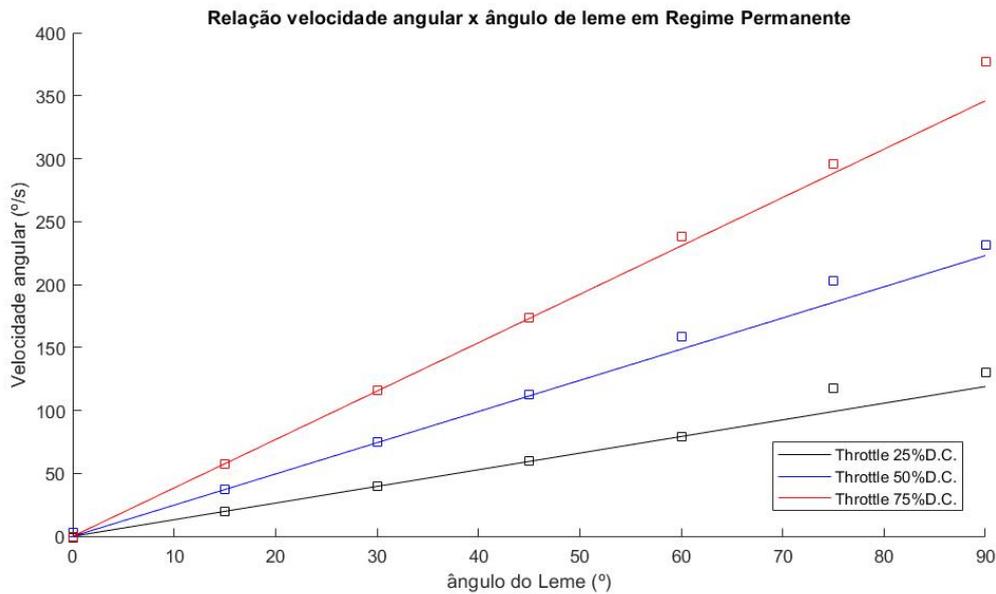


Figura 3.22: Relação entre velocidade angular e ângulo de leme sob diferentes valores de throttle.

seu comportamento em regime permanente, percebe-se ainda que é fortemente influenciado pela intensidade do *throttle* (Figura 3.22), de forma que deve-se novamente utilizar um ganho escalonado. A forma do sistema é então [11] :

$$P_a(s) = K_a(DC_t) \cdot e^{-\theta s} \cdot \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

em que θ é o valor do atraso no tempo em segundos, ζ é a constante de amortecimento, ω_n é a frequência natural em radianos por segundo e K_a é o ganho de regime permanente escalonado pelo *duty-cycle* de *throttle*.

O valor utilizado para o atraso de transporte será o mesmo utilizado na velocidade linear, $\theta = 0,368s$. Já os valores de ζ e ω_n são obtidos a partir dos valores médios de máximo percentual de *overshoot* (M_p) e tempo de acomodação (2%) do sistema ($t_a(2\%)$) obtidos dos ensaios. Para isso, são utilizadas as relações a seguir [11]:

$$M_p = e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}} = 42,92\% \longrightarrow \zeta = 0,26$$

$$t_a(2\%) = \frac{4}{\zeta\omega_n} = 3,197s \longrightarrow \omega_n = 4,9rad/s$$

A função que representa o ganho escalonado $K_a(throttle)$ pode ser obtida a partir da regressão linear dos valores em regime permanente apresentados na Figura 3.22. Calculando-se a inclinação das três retas obtidas, tem-se:

$$K_a(25\%) = 1,322$$

$$K_a(50\%) = 2,478$$

$$K_a(75\%) = 3,844$$

$$K_a(DC_t) = 5,089 \cdot DC_t$$

Obtem-se, por fim, a função de transferência da velocidade angular em função do ângulo do leme:

$$P_a(s) = 5,089 \cdot DC_t \cdot e^{-0,368s} \cdot \frac{24,01}{s^2 + 2,548s + 24,01}$$

A partir da função de transferência modelada, é possível confrontar os resultados experimentais com a resposta obtida com o modelo, a fim de validá-lo. Os ensaios apresentados na Figura 3.21 foram refeitos, e seus resultados são comparados com o modelo na Figura 3.23. Pode-se observar que a característica oscilatória do sistema foi capturada de maneira satisfatória, com leve variação no ganho modelado em relação ao real.

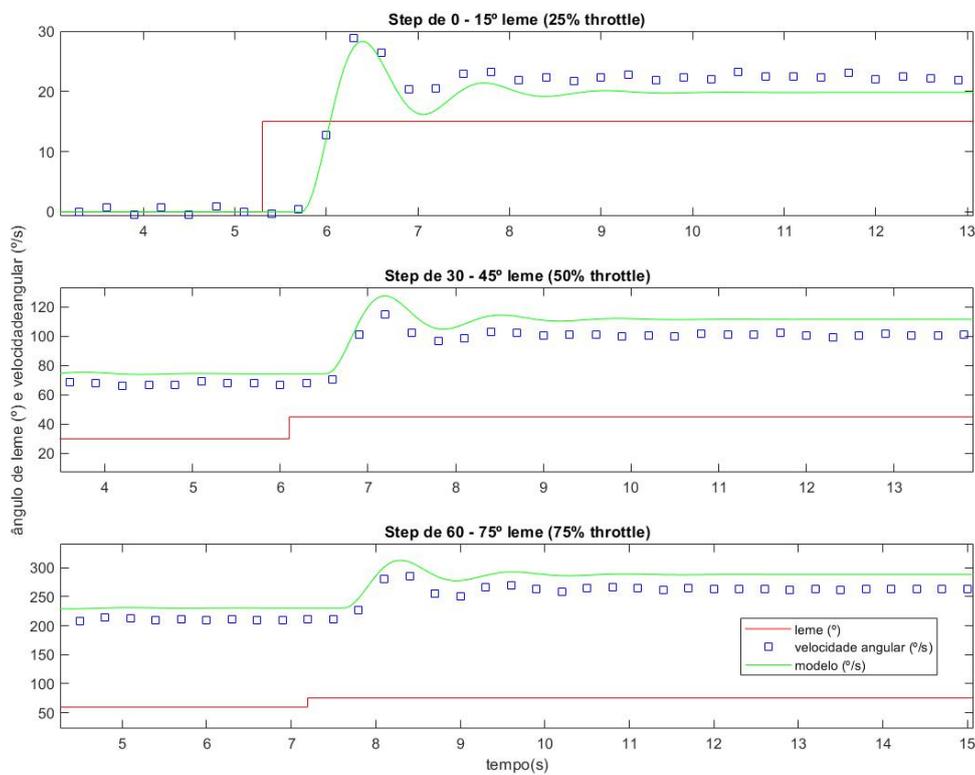


Figura 3.23: Validação do modelo obtido de velocidade angular.

3.12 Estrutura de Controle

De posse dos modelos dinâmicos das variáveis manipuladas (velocidade linear (m/s) e angular (°/s)) em função das variáveis controladas (*Lift* (%D.C), *Throttle* (%D.C) e *Rudder* (°)), é possível realizar a representação do sistema modelado na forma de diagrama de blocos (Figura 3.24). Nesse modelo estão representados a discretização dos sinais de entrada, por meio de seguradores de ordem zero (ZOH); o atraso de transporte, predominantemente proveniente da latência de comunicação; o escalonamento dos ganhos da planta de velocidade linear

pelo valor de Lift e da planta de velocidade angular pelo valor de throttle; o comportamento dinâmico, expresso pelas funções de transferência $P_v(s)$ e $P_a(s)$; e a amostragem discreta dos sensores, representada também por seguradores de ordem zero.

É interessante observar que não são representados atrasos de transporte na leitura das variáveis controladas. Isso se deve ao fato de que suas leituras são enviadas juntamente com a leitura de tempo do módulo GPS, o que compensa esse atraso do ponto de vista do modelo.

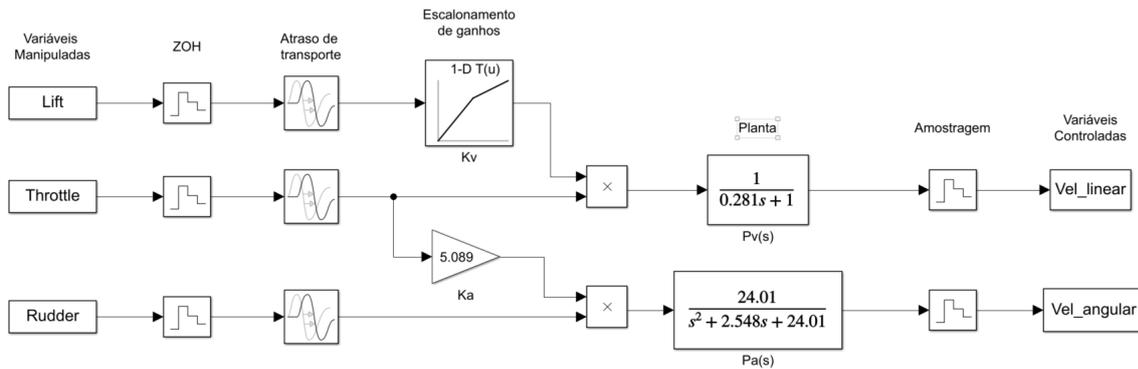


Figura 3.24: Diagrama de blocos do modelo identificado do sistema.

São utilizadas duas referências para o controle das variáveis manipuladas, que serão recebidas da interface gráfica. Dessa forma, o atraso de transporte modelado é atribuído diretamente à leitura desses setpoints pelo controlador. Em razão disso, o valor aplicado de lift será inferido utilizando o conceito de controle por faixas. De acordo com o valor do setpoint de velocidade linear, é atribuído o valor de 25, 50 ou 75% de duty-cycle ao lift, de forma a reduzir o esforço de controle necessário para atingir o valor de referência. Por simplicidade, os parâmetros de histerese para a definição do valor de lift foram definidos arbitrariamente a partir da figura 3.18.

$$Lift(sp_v \leq 1m/s) = 25\%$$

$$Lift(1m/s < sp_v < 2m/s) = 50\%$$

$$Lift(sp_v \geq 2m/s) = 75\%$$

3.12.1 Controlador de Velocidade linear

Com relação ao controle de velocidade linear, foram definidos os seguintes objetivos para o controlador projetado:

- seguimento de referência constante com o menor erro possível, considerando o ruído considerável nas medições.
- tempo de acomodação ($t_a(5\%)$) de 0,6s.
- capacidade de atingir o valor de referência sem sobreoscilação.

A partir dessas definições, pode-se sintonizar um controlador PI via síntese direta. Considerando um tempo de acomodação de 0,6s, pode-se calcular a constante de tempo desejada para o sistema por meio da aproximação $\tau_d \approx t_a(5\%)/3 = 0,20s$. Com isso, a dinâmica desejada da planta em malha fechada é:

$$T_v(s) = \frac{1}{\tau_d s + 1} = \frac{K_v C_v(s) P_v(s)}{1 + K_v C_v(s) P_v(s)}$$

$$\frac{1}{0,2s + 1} = \frac{K_v C_v(s)}{0,281s + 1 + K_v C_v(s)}$$

A partir dessa expressão, é extraída algebricamente a expressão do controlador C_v no domínio s:

$$C_v(s) = \frac{1}{K_v} \cdot \frac{0,281s + 1}{0,2s}$$

Convertendo a função de transferência do controlador para o domínio z com período de amostragem de 0,1s (definido no firmware do controlador), tem-se:

$$C_v(z) = \frac{1}{K_v} \cdot \frac{1,405z - 0,905}{z - 1}$$

A partir da função de transferência em z, é possível então depreender a equação de diferenças que deve ser implementada no controlador:

$$\frac{T(z)}{V(z)} = \frac{1}{K_v} \cdot \frac{1,405z - 0,905}{z - 1}$$

$$T[k] = T[k - 1] + \frac{1}{K_v} (1,405V[k] - 0,905V[k - 1])$$

3.12.2 Controlador de velocidade angular

Com relação ao controle de velocidade angular, foram definidos os seguintes objetivos para o controlador projetado:

- seguimento de referência constante com o menor erro possível;
- redução de 50% de tempo de acomodação;
- atenuação da componente oscilatória do sistema o máximo possível ($\zeta \geq 0,5$);
- redução de máximo overshoot ($M_p \leq 10\%$).

A partir desses objetivos, deriva-se o comportamento desejado do sistema ($T_d(s)$):

$$\zeta = 0,6 > 0,5 \longrightarrow M_p = e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}} = 9,48\% < 10\%$$

$$t_a(2\%) = \frac{4}{\zeta\omega_n} = 1,599s \longrightarrow \omega_n = 4,169rad/s$$

$$T_d(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{17,38}{s^2 + 5,00s + 17,38} = \frac{17,38}{(s + 2,5 + j3,336)(s + 2,5 - j3,336)}$$

Essas definições indicam que os polos dominantes da função de transferência do sistema controlado devem ter parte real menor do que $-2,5$ (a fim de garantir a condição de tempo de acomodação) e parte imaginária de módulo menor que $3,336$ (para garantir as condições de oscilação). Comparando-se o lugar geométrico das raízes do sistema modelado ($G_a(s)$) e desejado ($T_d(s)$), apresentado na figura 3.25, é possível utilizar o método de posicionamento de polos para sintonizar um controlador PID que satisfaça os objetivos propostos.

A estrutura de PID para a sintonização seguirá a forma:

$$C_a(s) = \frac{(s - z_1)(s - z_2)}{s(s - p)}$$

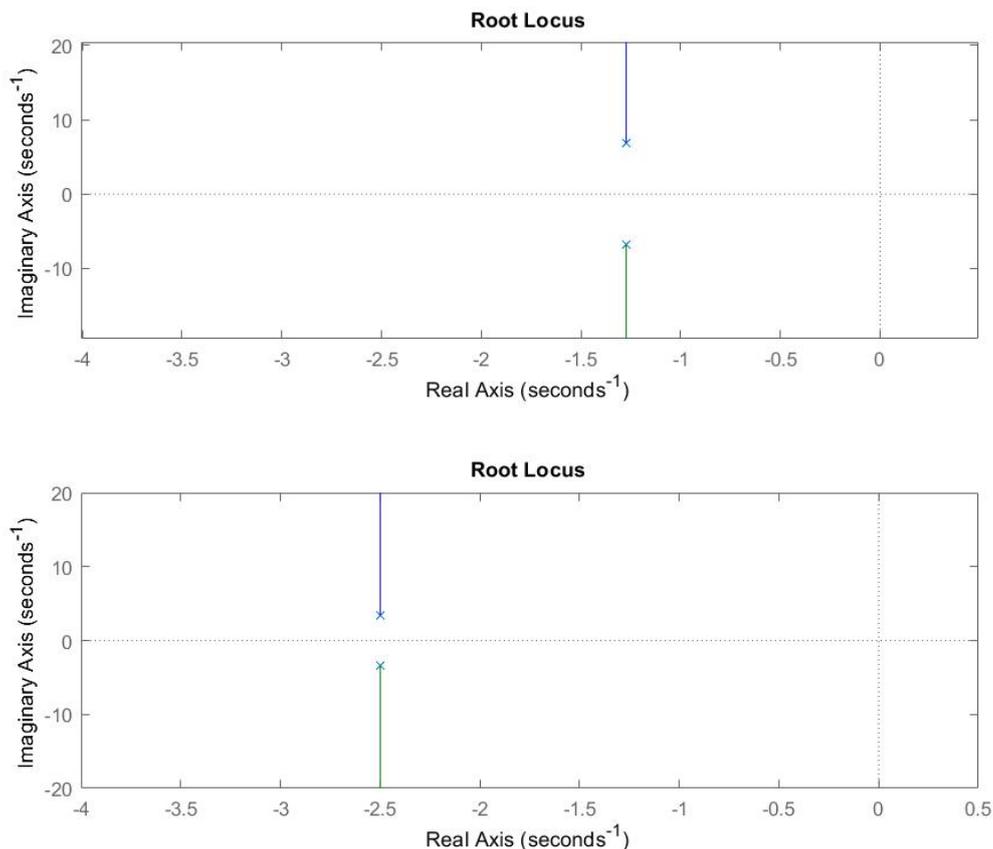


Figura 3.25: Comparação do lugar geométrico das raízes do sistema modelado e do sistema desejado, respectivamente.

Dessa forma, a sintonização do controlador PID se resume ao posicionamento de seus dois zeros (z_1 e z_2) e do polo (p). Observa-se na figura 3.25 que é necessário atrair os polos dominantes do sistema para a esquerda, evitando-se o aumento de sua parte imaginária. Para

isso, pode-se fixar os dois zeros em $s = -3 \pm 3,5j$ e seu polo em $s = -10$, de forma a evitar que este influencie de forma dominante na dinâmica. É interessante observar o novo lugar das raízes do sistema e sua resposta ao degrau unitário (figura 3.26). Os zeros complexos conjugados do controlador aceleram a resposta do sistema sem aumentar significativamente sua componente aleatória. Pode-se observar também que não houve sobreoscilação, o que indica que o controlador ajustado deve atender os objetivos propostos.

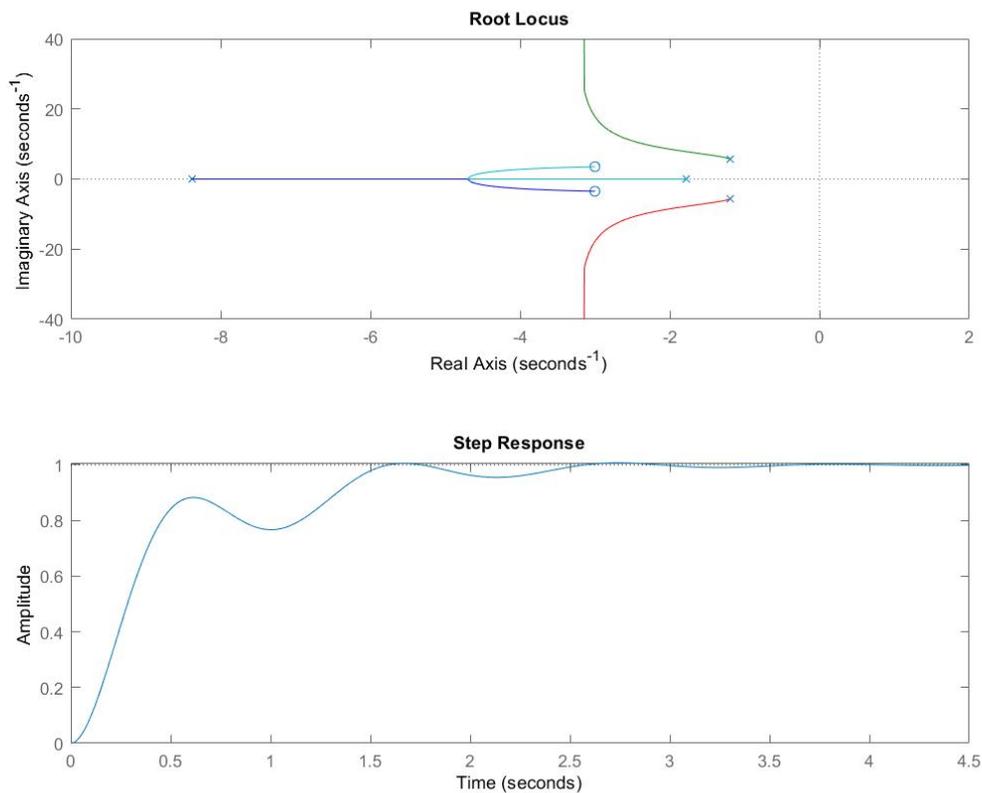


Figura 3.26: Lugar geométrico das raízes e resposta ao degrau do sistema em malha fechada com o controlador sintonizado.

Convertendo-se então o controlador projetado para o domínio z , tem-se:

$$\frac{V_a(z)}{\Theta_a(z)} = \frac{z^2 - 1,543z + 0,6769}{z^2 - 1,368z + 0,3679}$$

Com isso, a equação de diferenças implementada no firmware do microcontrolador é:

$$\Theta_a[k] = 1,368\Theta_a[k-1] - 0,3679\Theta_a[k-2] + V_a[k] - 1,543V_a[k-1] + 0,6769V_a[k-2]$$

3.12.3 Validação da estrutura de controle

Para realizar a validação dos controladores projetados, é necessário representar o modelo completo do sistema em malha fechada na forma de diagrama de blocos do *SimuLink*, a

fim de tornar possível a comparação dos dados experimentais com os obtidos por meio do modelo. Na figura 3.27, é possível ver os blocos funcionais que compõem o sistema e que serão utilizados para validá-lo.

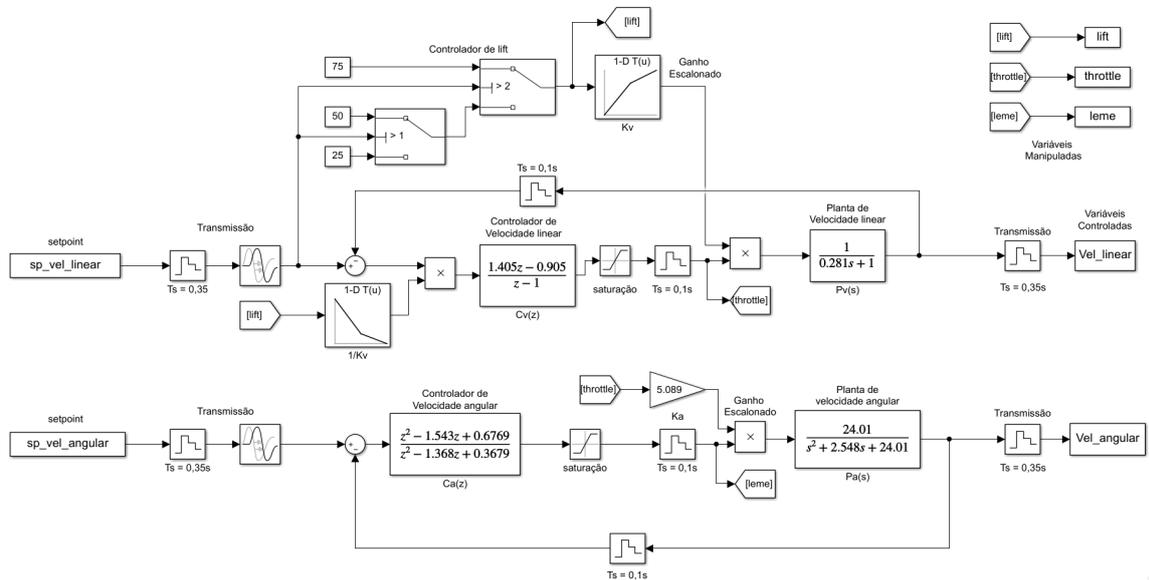


Figura 3.27: Modelo em simulink do sistema de controle projetado.

É necessário tecer alguns comentários acerca do modelo implementado no Simulink. Em primeiro lugar, nota-se que os sinais de entrada apresentam um bloco de atraso de transporte, enquanto os de saída, não. Isso se deve ao fato de que, no protótipo implementado, os pares de dados de entrada e saída recebem o mesmo valor de tempo. A fim de modelar a latência de conexão entre a mudança de setpoint e a sua transmissão ao microcontrolador, foi necessário incluir esse bloco, conforme apresentado na seção 3.9. O controlador de lift, descrito na mesma seção, apresenta comportamento idêntico ao descrito, que é o de um controlador do tipo feed-forward (pré-compensador de referência). A saturação das variáveis manipuladas também foi implementada para este modelo, sendo, para o throttle, de 0 a 100%, e para o leme, de -90° a 90° , conforme a interface gráfica (seção 3.7) sugere. Os seguradores de ordem zero (ZOH) representam a discretização do sistema. Para o período de amostragem do microcontrolador, conforme apresentado na seção 3.5, tem-se $T_s = 100\text{ms}$. Já no processo de comunicação com a interface, tem-se um tempo de amostragem levemente variável e dependente da qualidade da conexão, fixado para fins de validação em 350ms.

Estando concluída a modelagem necessária à validação do sistema, o próximo passo é a testagem do protótipo, a fim de verificar sua capacidade de seguir as referências estabelecidas perante às restrições propostas. É pertinente observar aqui que tais restrições definidas nas seções 3.12.1 e 3.12.2 são não apenas objetivos propostos para este trabalho, como também sugestões de diretrizes didáticas para adoção do docente na abordagem da planta num contexto de sala de aula, uma vez que tangibilizam a percepção do aluno sobre os objetivos de controle que devem ser observados.

Com o intuito de testar o protótipo sob a gama de pontos de operação identificados, deve ser percorrido todo o range de controlabilidade modelado. Para isso, são realizados degraus de referência subsequentes a fim de observar o comportamento dos controladores. Na figura 3.28 podem ser visualizados os resultados de um desses testes.

Observa-se que o sistema real é representado pelas curvas azul-escuras e vermelhas, enquanto o sistema modelado, pelas curvas em ciano e verde. Infelizmente, a transmissão de dados entre microcontrolador e computador é realizada a uma taxa de amostragem menor do que a utilizada para o controle. Em razão disso, a resolução sobre as variáveis manipuladas não permite visualizar com grande detalhamento o comportamento transitório após as transições de referência. Apesar disso, os controladores aparentam atender aos objetivos, não apresentando sobre oscilação significativa, não saturando em nenhum momento dos testes e reduzindo o tempo de acomodação do sistema.

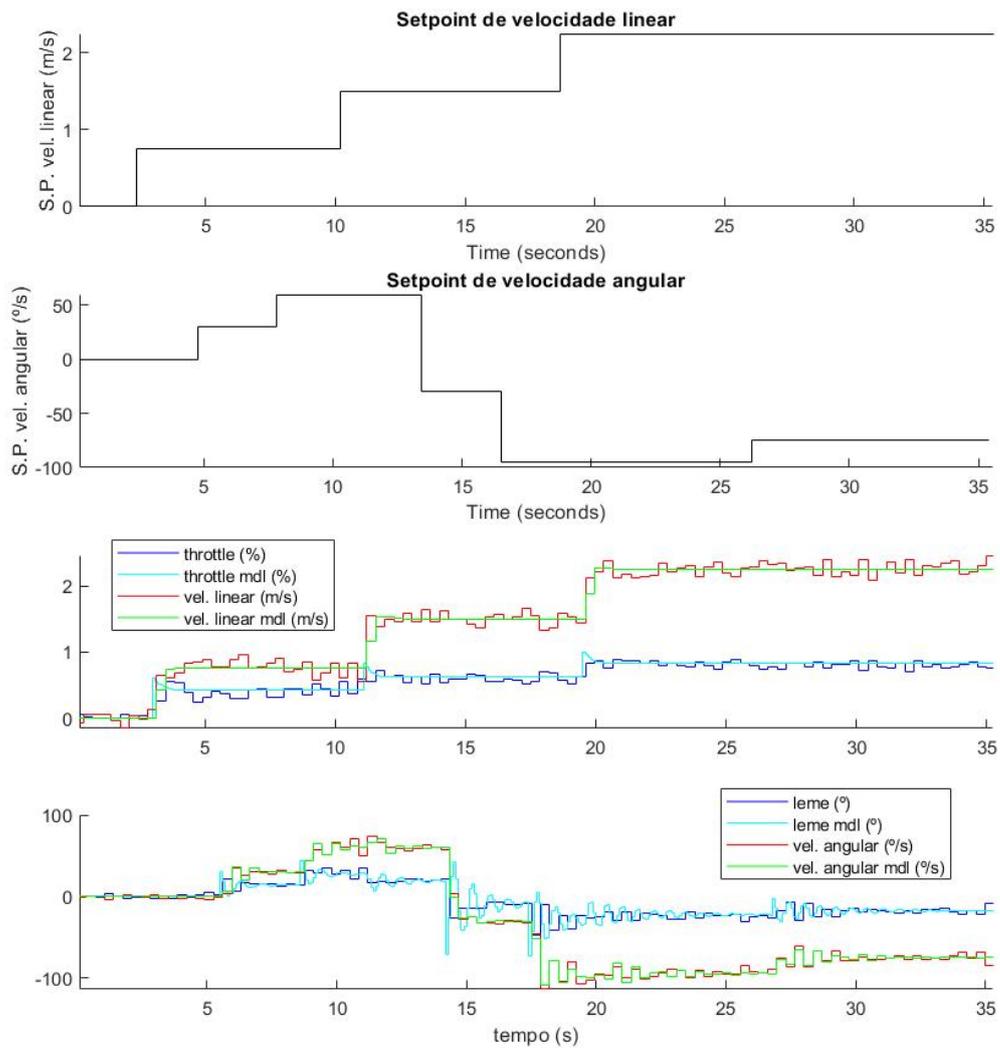


Figura 3.28: Ensaio para validação do sistema controlado.

Capítulo 4

Conclusões

4.1 Observações e Propostas de Continuidade

Ao longo do desenvolvimento deste protótipo, algumas possibilidades de melhoria foram identificadas. Porém, devido a limitações de tempo e recursos, não puderam ser implementadas nesta primeira versão. Ainda assim, são ideias interessantes que merecem ser comentadas, pois podem embasar futuras melhorias.

Em primeiro lugar, acerca da medição de velocidade linear, é importante frisar que o valor escalar dessa grandeza obtido com exatidão limitada pelo módulo GPS não foi a solução mais adequada. Conforme apresentado na seção 3.2.3, um sensor óptico pode fornecer a velocidade vetorial em duas dimensões, com exatidão superior e maior frequência de amostragem. Todavia, esse tipo de sensor apresenta limitações devido a sua fragilidade mecânica, necessidade de contato próximo com a superfície e baixa robustez diante de superfícies empoeiradas, o que não é desejável. Sendo assim, uma outra alternativa seria a adaptação de um mouse mecânico para essa função. Além de ser mais robusto e apresentar acurácia semelhante, seria um sensor mais simples e prático de integrar ao projeto.

Em segundo lugar, é importante comentar as limitações da comunicação via wi-fi empregada no projeto. Apesar de conferir grande comodidade de utilização, bastando que o computador e o protótipo estejam conectados a uma mesma rede, a escolha por esse tipo de comunicação apresenta duas principais desvantagens. Mesmo uma conexão wi-fi de boa qualidade apresenta uma latência variável e muitas vezes significativa, a qual é especialmente crítica para uma planta com transitórios rápidos como esta. Outra questão é a baixa confiabilidade dessa forma de comunicação sem fio, que limita severamente o alcance para utilização do protótipo. Uma alternativa interessante, apresentada por Consten [5], é a utilização de transmissores e receptores AM, o que aumenta muito o alcance e garante uma latência constante de comunicação. Com essa modificação, seria possível trazer a implementação da rotina de controle para o próprio ambiente do MATLAB, deixando apenas a leitura dos sensores e comando dos atuadores no microcontrolador. Essa melhoria tornaria o uso da ferramenta mais didático, facilitando a experiência de aprendizado em laboratório.

Por fim, alguns comentários sobre a construção do protótipo em si. Os motores utilizados, embora práticos por já terem turbinas acopladas, estão superdimensionados em relação à massa do protótipo, o que acarreta um consumo de bateria excessivo. Ademais, seria interessante desenvolver uma forma mais prática de trocar a saia plástica que o envolve, haja visto que o atrito contínuo com a superfície leva a sua deterioração.

4.2 Considerações Finais

O desenvolvimento de um protótipo funcional é uma atividade naturalmente complexa e multidisciplinar, uma vez que envolve tanto um domínio robusto dos conteúdos teóricos quanto o traquejo e a destreza necessários para a realização de sua construção física. Tratando-se de um protótipo de uma planta didática, esses dois aspectos são essenciais, pois, além de funcionalidade, deve-se ter em vista sua praticidade de uso, sua robustez física e seu propósito como ferramenta de ensino.

Ao adotar um *Hovercraft* como objeto de estudo, buscou-se propositalmente afastar o aluno dos exemplos clássicos de sistemas de controle, trazendo-o para uma situação de aprendizado completamente nova. Outra razão para essa escolha é que a formação técnica do engenheiro de controle deve abranger não somente o contexto de aplicações industriais, mas também o desenvolvimento de bens de consumo e, em especial, de veículos automatizados, uma forte tendência do mercado mundial. Além disso, um importante propósito pretendido com este projeto é o de atizar a curiosidade e o desejo de aprender nos alunos, tornando o processo de formação acadêmica mais leve e estimulante.

Levando-se em conta a natureza multifacetada desse projeto, é natural que, sob alguns aspectos, ele não atenda às expectativas de um produto final. Seu objetivo foi, primordialmente, demonstrar a riqueza de possibilidades a serem exploradas no ensino de Engenharia de Controle. Sendo assim, esse projeto deve ser compreendido como um mínimo produto viável de uma proposta de ensino mais desafiadora, engajadora e cativante.

Referências Bibliográficas

- [1] Norhafizan Ahmad, Raja Ariffin Raja Ghazilla, and Nazirah M. Khairi. Reviews on various inertial measurement unit (IMU) sensor applications. *International Journal of Signal Processing Systems*, 1, 2013.
- [2] Limited Beitian Co. BN-220 GPS module + antenna datasheet.
- [3] M.J. Caruso. Applications of magnetic sensors for low cost compass systems. *Position Location and Navigation Symposium, IEEE*, pages 177–184, 2000.
- [4] Shenzhen Ai-Thinker Technology Co. Nodemcu-32s Datasheet.
- [5] H. J. L. M. Consten. Control of a model sized hovercraft. 2003.
- [6] Fierce Electronics. Compensating for tilt, hard-iron, and soft-iron effects.
- [7] Hobbywing Inc. Skywalkerv2 manual.
- [8] InvenSense Inc. PS-MPU-9250A-01-v1.1.
- [9] Sooyong Lee and Jae-Bok Song. Mobile robot localization using optical flow sensors. *International Journal of Control, Automation, and Systems*, 2:485–493, 2004.
- [10] link=Url: <http://www.towerpro.com.tw/product/sg5010-4/> TORQ PRO TOWER PRO. Servomotor SG-5010 Specifications.
- [11] K. Ogata. *Modern Control Engineering*. Prentice-Hall, Inc, Englewood Cliffs, New Jersey, USA, 1990.
- [12] B.E. Okafor. Development of a hovercraft prototype. *International Journal of Engineering and Technology*, 3, No. 3:276–281, 2013.
- [13] Chul Ki Song, Michael Uchanski, and J. Karl Hedrick. Vehicle speed estimation using accelerometer and wheelspeed measurements. *Society of Automotive Engineers, Inc.*, 2002.
- [14] Espressif Systems. ESP32-WROOM-32 Datasheet.
- [15] L. Yun and A. Bliault. *Theory and Design of Air Cushion Craft*. John Wiley & Sons Inc, 605 Third Avenue, New York, NY 10158-0012, 2000.