

PM

Navegação de um Barco Autónomo

PROJETO DE MESTRADO

Martim da Silva Abreu Fernandes

MESTRADO EM ENGENHARIA ELETROTÉCNICA - TELECOMUNICAÇÕES



UNIVERSIDADE da MADEIRA

A Nossa Universidade

www.uma.pt

dezembro | 2025

Navegação de um Barco Autónomo

PROJETO DE MESTRADO

Martim da Silva Abreu Fernandes

MESTRADO EM ENGENHARIA ELETROTÉCNICA - TELECOMUNICAÇÕES

ORIENTAÇÃO

João Dionísio Simões Barros

RESUMO

Neste trabalho desenvolve-se um sistema de navegação autónoma de um barco. No projeto do sistema de navegação são utilizadas metodologias de controlo para corrigir o ângulo de direção e a velocidade do barco para os valores desejados de maneira a percorrer uma trajetória programada. Constrói-se, também, um telecomando para o utilizador interagir com o barco.

Para projetar o controlador deduz-se um modelo representativo da física do barco e a partir dos modelos e controladores simula-se o comportamento dos sistemas de navegação autónomos. O barco autónomo ao percorrer a trajetória está constantemente a fazer a leitura do sensor de *Global Positioning System* (GPS) e verifica se já chegou ao local de destino, observando se a distância entre os pontos GPS é menor que uma margem pré-definida. E quando chegar ao ponto de referência passa para o próximo local, percorrendo a trajetória de referência.

Através destas metodologias, amostraram-se os parâmetros relevantes ao modelo, construindo um programa de simulação no MATLAB/SIMULINK e desenvolveu-se o circuito do barco e telecomando, utilizando microcontroladores para o processamento, sensores para amostragem de dados, rádios para comunicação, *driver* de motor para acionar o motor e periféricos de entrada para interagir com o utilizador.

Os resultados evidenciam que o sistema de navegação funciona, com um desvio inferior a 1 metro; o modelo físico deduzido é válido, apresentando resultados em concordância com a análise teórica e simulação. A utilização destes circuitos e código constituem uma solução válida para um sistema de navegação autónomo de um barco.

Palavras-chave:

Barco autónomo, sistema de navegação marítima, modelo físico do barco, telecomando, trajetória com GPS, controlo de direção.

ABSTRACT

This work describes the development of an autonomous boat navigation system. The navigation system design uses control methodologies, to correct the boat's steering angle and speed to the desired values, in order to follow a programmed trajectory. A remote control is also built for the user to interact with the boat.

To design the controller, a representative model of the boat's physics is deduced, and from these models and controllers, the behavior of autonomous navigation systems is simulated. As the autonomous boat travels along its trajectory, it constantly reads data from the Global Positioning System (GPS) sensor and checks if it has reached its destination, observing whether the distance between GPS points is less than a predefined margin. When it reaches a reference point, it moves on to the next location, following the reference trajectory.

Through these methodologies, the parameters relevant to the model were sampled, a simulation program was built in MATLAB/SIMULINK, and the boat and remote control circuit was developed, using microcontrollers for processing, sensors for data sampling, radios for communication, a motor driver to activate the motor, and input peripherals to interact with the user.

The results show that the navigation system works, with a deviation of less than 1 meter; the deduced physical model is valid, presenting results in agreement with the theoretical analysis and simulation. The use of these circuits and code constitutes a valid solution for an autonomous boat navigation system.

Key-words:

Autonomous surface vehicle, maritime navigation system, boat physical model, remote control, GPS trajectory, direction control system.

AGRADECIMENTOS

Um agradecimento especial ao meu orientador, o Prof. João Dionísio Barros, por todo o tempo que despendeu para a orientação do projeto da dissertação de mestrado, pelos esclarecimentos de dúvidas, pelos bons conselhos, sugestões e conhecimento que me transmitiu, pela compreensão das dificuldades que surgiram ao longo do desenvolvimento do trabalho, pelo fornecimento de bibliografia no começo do trabalho, pelo fornecimento de material necessário para a realização do projeto, pela exigência e rigor que impôs e pela sua disponibilidade ao longo de todo o projeto.

À minha família gostaria de agradecer pela sua compreensão pelo tempo gasto e a minha indisponibilidade ao longo do desenvolvimento do projeto, pelo carinho, pela amizade pelo apoio e convívio que me deram.

Aos meus colegas de curso e amigos, gostaria de agradecer pela força que me deram, pelo apoio, pelo convívio, pelo interesse e curiosidade sobre o projeto e por todos os momentos que partilhamos.

A todos os professores da área de Eletrotécnica – Telecomunicações e da área de Física da Universidade da Madeira gostaria de agradecer, pelas boas explicações e transmissão de conhecimentos que serviram de base para muitos tópicos de interesse nesta dissertação.

E finalmente ao Eng. Filipe Santos gostaria de agradecer, pelo grande apoio e transmissão de conhecimentos práticos que me forneceu, pela ajuda na montagem e planeamento do circuito, pelo fornecimento dos componentes eletrónicos necessários e pela experiência que recebi graças ao apoio dele. A todos os referidos um muitíssimo obrigado.

LISTA DE ABREVIATURAS E TERMINOLOGIA

9DOF	Nove graus de liberdade (<i>Nine Degrees Of Freedom</i>)
AC	Corrente alternada (<i>Alternating Current</i>)
AIS	Sistema de identificação automática (<i>Automatic Identification System</i>)
DC	Corrente contínua (<i>Direct Current</i>)
DSC	Chamada seletiva digital (<i>Digital Selective Calling</i>)
FM	Modulação de frequência (<i>Frequency Modulation</i>)
GPS	Sistema de posicionamento global (<i>Global Positioning System</i>)
I2C	Circuito Inter-Integrado (<i>Inter-Integrated Circuit</i>)
ITU	União Internacional das Telecomunicações (<i>International Telecommunication Union</i>)
PWM	Modulação de largura do impulso (<i>Pulse-Width Modulation</i>)
SPI	Interface periférica série (<i>Serial Peripheral Interface</i>)
UART	Recetor-transmissor assíncrono universal (<i>Universal Asynchronous Receiver-Transmitter</i>)
UTM	Projeção universal transversal de Mercator (<i>Universal Transverse Mercator projection</i>)
VHF	Frequência muito alta (<i>Very High Frequency</i>)

LISTA DE SÍMBOLOS MAIS FREQUENTES

$\sim v$	Velocidade média
$\sim \tau$	Binário motor médio
Acl_{Xs}	Aceleração no eixo dos Xs
Acl_{Xs}'	Aceleração calibrada no eixo dos Xs
Acl_{Ys}	Aceleração no eixo dos Ys
Acl_{Ys}'	Aceleração calibrada no eixo dos Ys
Acl_{Zs}	Aceleração no eixo dos Zs
Acl_{Zs}'	Aceleração calibrada no eixo dos Zs
b	Atrito do ar
B	Campo magnético
C_1	Constante equivalente da força diferencial
C_2	Constante equivalente da força comum
C_t	Ganho da velocidade de rotação
C_v	Constante de atrito da velocidade linear
C_ω	Constante de atrito da velocidade de rotação
\vec{F}	Força
\vec{F}_c	Força centrípeta
$Gyro_{Xs}$	Velocidade de rotação no plano perpendicular a Xs
$Gyro_{Xs}'$	Velocidade de rotação calibrada no plano perpendicular a Xs
$Gyro_{Ys}$	Velocidade de rotação no plano perpendicular a Ys
$Gyro_{Ys}'$	Velocidade de rotação calibrada no plano perpendicular a Ys
$Gyro_{Zs}$	Velocidade de rotação no plano perpendicular a Zs
$Gyro_{Zs}'$	Velocidade de rotação calibrada no plano perpendicular a Zs
i	Corrente
J	Momento de inércia
k_b	Constante <i>back-electromotive force</i>
k_d	Constante derivativa do controlador de direção
k_p	Constante proporcional do controlador de direção
k_t	Ganho da corrente
k_w	Constante proporcional do controlador de velocidade de rotação
k_{wi}	Constante integral do controlador de velocidade de rotação
K	Ganho do modelo de <i>Nomoto</i>
l	Comprimento do condutor
L	Indutância
m	Massa
Mag_{Xs}	Campo magnético no eixo dos Xs
Mag_{Xs}'	Campo magnético calibrado no eixo dos Xs
Mag_{Ys}	Campo magnético no eixo dos Ys
Mag_{Ys}'	Campo magnético calibrado no eixo dos Ys
Mag_{Zs}	Campo magnético no eixo dos Zs
Mag_{Zs}'	Campo magnético calibrado no eixo dos Zs
r	Raio da terra
R	Resistência do motor
S	Área de secção

s	Frequência complexa, transformada de <i>Laplace</i> do tempo
t	Tempo
t_r	Tempo de reação
\vec{T}	Força aplicada no barco pelo motor
\vec{T}_1	Força aplicada no barco pelo motor 1
\vec{T}_2	Força aplicada no barco pelo motor 2
\vec{T}_{com}	Força comum aplicada no barco pelos motores
\vec{T}_{dif}	Força diferencial aplicada no barco pelos motores
T_{ds}	Constante de tempo do modelo de <i>Nomoto</i>
$UTME_{atual}$	Coordenada UTM do local atual este
$UTME_{ref}$	Coordenada UTM do local pretendido este
$UTMN_{atual}$	Coordenada UTM do local atual norte
$UTMN_{ref}$	Coordenada UTM do local pretendido norte
V	Tensão de entrada no motor
v	Velocidade
V_{bat}	Tensão da bateria
v_{max}	Velocidade pretendida máxima
v_{ref}	Velocidade pretendida
α	Atrito aquático
β	Ganho do binário de motor
Δt	Intervalo de tempo
$\Delta\theta$	Erro do ângulo
$\Delta\lambda$	Diferença entre longitudes
$\Delta\varphi$	Diferença entre latitudes
θ	Ângulo de direção atual
θ_{ref}	Ângulo de direção desejada
μ_{dif}	Variável de controlo intermédia diferencial
μ_{com}	Variável de controlo intermédia comum
ρ	Raio de aceitação
τ	Binário motor
τ_c	Binário de carga
τ_g	Binário motor gerado
φ_1	Latitude atual
φ_2	Latitude do local pretendido
ψ_{pitch}	Ângulo de inclinação
ψ_{roll}	Ângulo de enrolamento
Ω	Velocidade de rotação dos motores
ω	Velocidade de rotação atual do barco
ω_{ref}	Velocidade de rotação pretendida do barco

ÍNDICE

Resumo	i
Abstract	iii
Agradecimentos	v
Lista de abreviaturas e terminologia.....	vii
Lista de símbolos mais frequentes	ix
Índice.....	xi
Capítulo 1 Introdução.....	1
1.1 Motivação	2
1.2 Objetivos.....	3
1.3 Organização e conteúdos	4
Capítulo 2 Revisão da bibliografia	7
2.1 Barcos	8
2.2 Navegação	10
2.3 Monitorização	13
2.4 Sensores	14
2.5 Comunicação.....	18
2.6 Motores.....	21
2.7 Sistemas de acionamento	26
2.8 Controlo de barcos autónomos.....	28
2.9 Navegação autónoma.....	32
2.10 Considerações finais sobre a revisão da bibliografia.....	34

Capítulo 3 Especificações do barco autónomo 35

3.1 Sistema do barco autónomo.....	36
3.2 Controlo de direção do barco autónomo.....	37
3.3 Comando de velocidade linear do barco autónomo.....	38
3.4 Funcionalidades do barco autónomo.....	39
3.5 Considerações finais sobre as especificações do barco	42

Capítulo 4 Dedução do modelo físico e projeto dos controladores do barco autónomo 43

4.1 Dedução do modelo do barco autónomo.....	44
4.2 Projeto dos controladores do barco autónomo	49
4.3 Considerações finais sobre o modelo do barco e projeto de controladores.....	54

Capítulo 5 Simulação do sistema de navegação do barco autónomo 55

5.1 Implementação do diagrama de blocos no SIMULINK/MATLAB.....	56
5.2 Simulação do sistema de navegação	62
5.3 Considerações finais sobre as simulações.....	70

Capítulo 6 Implementação e testes do barco autónomo 71

6.1 Escolha do barco, dos motores e <i>drivers</i>	72
6.2 Sensores para obtenção da posição e direção do barco autónomo	73
6.3 Microcontroladores e rádios do barco autónomo.....	74
6.4 Telecomando do barco	76
6.5 Circuito global do barco	78

6.6 Comunicação e funcionalidades do barco autónomo	81
6.7 Comando do barco	82
6.8 Teste de mudança de direção do barco autónomo	83
6.9 Teste de seguimento de trajetória pré-definida pelo barco autónomo	87
6.10 Teste de alcance	90
6.11 Considerações finais sobre a implementação e testes do barco autónomo.....	92
Capítulo 7 Conclusões	93
7.1 Conclusões gerais	94
7.2 Propostas para trabalhos futuros.....	100
Referências	101
Apêndice A Dedução e simplificação das equações do modelo do barco autónomo	105
A.1 Dedução das equações do modelo do barco	106
A.2 Determinação dos parâmetros dos controladores.....	108
Apêndice B Testes de determinação dos parâmetros do barco	109
B.1 Teste dos motores elétricos do barco autónomo	110
B.2 Testes da hidrodinâmica do barco	111
B.3 Testes para determinar os parâmetros dos motores.....	116
B.4 Determinação dos parâmetros apresentados no Apêndice B	118

Apêndice C Configuração do sistema do barco autónomo..... 121

C.1 Amostragem dos sensores do barco autónomo e calibração do sensor de direção	122
C.2 Comunicação com o telecomando do barco	130
C.3 Controlo do barco autónomo.....	139
C.4 Programa final do barco autónomo	150
C.5 Simulação do barco autónomo no MATLAB/SIMULINK	164

Apêndice D Visualização da reação do barco..... 167

D.1 Vídeos do controlo manual do barco.....	168
D.2 Vídeos de reação do sistema de controlo	168
D.3 Vídeos da trajetória pré-definida do barco autónomo num lago....	168
D.4 Vídeos da trajetória pré-definida do barco autónomo no mar	168

Capítulo 1

INTRODUÇÃO

Neste capítulo apresenta-se a motivação para a realização de sistemas de controlo autónomo de um barco para aplicações em amostragem de dados marítimos. Definem-se os objetivos a concretizar e descreve-se a organização e conteúdo do relatório em cada capítulo.

1.1 Motivação

Portugal é um país com uma grande tradição marítima, foi através do mar que Portugal cresceu. A recolha de dados marítimos é de extrema importância e valor para ajudar na sua conservação, no auxílio à pesca através da deteção de cardumes, em operações de busca e salvamento, para fins militares, entre outros.

Inicialmente, a recolha de dados marítimos não era de grande importância, os barcos realizavam o transporte, pesca e comércio necessários localmente e a influência do conhecimento dos dados marítimos nessas aplicações era pouco importante.

À medida que os sectores da indústria, comércio e pesca em alto mar sentiram maior necessidade de aumento de produtividade e de qualidade, para serem economicamente competitivos, as tripulações dos barcos começaram a mapear rotas, locais de pesca e usar a posição das estrelas para navegação, de maneira a apresentar melhor desempenho e rendimento na navegação marítima. Tal aumentou o interesse e exigência pelos dados marítimos. Atualmente, usa-se dados marítimos para a conservação do mar, previsão do tempo e auxílio à pesca.

A tendência atual de colocação de boias no mar para obtenção dos dados possui a desvantagem que só retirar esses dados num determinado local, para além de necessitar da reposição da energia gasta e indisponibilidade do serviço durante um certo tempo. O uso dos barcos comerciais com recolha de dados cobre muito pouco espaço marítimo, devido ao seu uso primário ser para tráfego e comércio. Os sensores com balões cobrem uma vasta área, mas não é possível controlar a trajetória. Por isso, em relação ao parâmetro de custo por área coberta, a utilização de um barco de pequenas dimensões, autónomo, que possa ser produzido em massa e que transmite os dados a longa distância, otimiza a recolha de dados marítimos relativamente às soluções anteriormente utilizadas.

1.2 Objetivos

O crescente aumento do interesse pelos barcos autónomos e pela recolha de dados marítimos tem sido um dos motivos de pesquisa de novas soluções para melhoria da qualidade e redução dos custos dos equipamentos responsáveis por tal.

Neste contexto os objetivos desta dissertação são os seguintes:

- 1) Rever a bibliografia relevante sobre barcos autónomos de maneira a entender o princípio de funcionamento de cada componente, especificamente a bibliografia sobre:
 - a. Tipos de barcos;
 - b. Navegação;
 - c. Monitorização;
 - d. Sensores;
 - e. Comunicação;
 - f. Motores;
 - g. Sistemas de acionamento;
 - h. Controlo;
 - i. Navegação autónoma.
- 2) Definir as especificações do barco autónomo para poder realizar:
 - a. Navegação controlada por telecomando;
 - b. Navegação pré-definida;
 - c. Controlo de direção;
 - d. Controlo de velocidade;
 - e. Transmissão de dados entre barco e utilizador.
- 3) Dedução do modelo físico do barco.
- 4) Projeto dos controladores do barco autónomo.
- 5) Simulação do sistema de navegação e comando do barco.
- 6) Implementação do sistema do barco autónomo, nomeadamente:
 - a. Escolha dos motores e *driver*;
 - b. Sensores para obtenção da posição e direção;
 - c. Microcontroladores;
 - d. Rádio;
 - e. Telecomando;

Capítulo 1 - Introdução

- f. Circuito global.
- 7) Testes de simulação e experimentais do barco autónomo:
- a. Testes de comunicação e funcionalidades;
 - b. Teste do controlo manual;
 - c. Testes de mudança de direção;
 - d. Testes de navegação entre dois pontos;
 - e. Testes de navegação entre N pontos;
 - f. Teste do alcance das comunicações;
 - g. Comparação entre simulação e experimental.

1.3 Organização e conteúdos

Esta dissertação está organizada em 7 capítulos (Introdução, Revisão da bibliografia, Especificações do barco autónomo, Dedução do modelo físico e projeto de controladores do barco autónomo, Simulação do sistema de navegação do barco autónomo, Implementação e testes do barco autónomo, Conclusões), Referências e Apêndices.

No capítulo 1, Introdução, faz-se um enquadramento do tema e da importância dos dados marítimos e dos sistemas de navegação autónomos nos barcos. Referem-se as principais desvantagens da utilização dos outros dispositivos utilizados para a obtenção de dados marítimos e a solução proposta para otimizar o custo por área coberta, sendo essa a motivação para o desenvolvimento deste projeto. Definem-se os objetivos a concretizar, apresentando também a organização do relatório e conteúdos de cada capítulo.

No capítulo 2, Revisão da bibliografia, apresentam-se os tópicos relevantes ao projeto. Dando a entender a base de funcionamento dos barcos, mostrando exemplos de como se controlam os barcos, apresentam-se diversos tipos de sensores e que tipo de dados são importantes monitorizar no mar. Descreve-se também a comunicação no mar e métodos de controlo utilizados nos barcos, finalizando com exemplos de sistemas de navegação autónomos.

No capítulo 3, Especificações do barco autónomo, propõem-se os tipos de componentes, lógica de circuitos e programas utilizados, de maneira a apresentar uma solução que enquadre os objetivos definidos para este trabalho. O problema é subdividido em partes e resolvendo cada uma delas, consegue-se

Capítulo 1 - Introdução

junta-las e apresentar uma solução. Separa-se, então, o problema nas seguintes partes: desenvolvimento do sistema do barco autónomo; projeto do controlo de direção do barco autónomo; projeto do comando da velocidade linear do barco autónomo; implementação das funcionalidades do barco autónomo.

No capítulo 4, Dedução do modelo físico e projeto dos controladores do barco autónomo, deduz-se o modelo físico do barco, que descreve como é que a partir de uma diferença de tensão aplicada nos motores se causa uma rotação no barco, e como a partir de uma tensão comum aplicada nos motores se causa uma velocidade linear no barco. Para tal dedução apresentam-se os modelos simplificados de cada parte e é realizada a junção de todos os modelos do sistema do barco autónomo. Projetam-se, também, os controladores do barco, impondo os tempos de reação pretendidos e realiza-se o dimensionamento dos parâmetros dos controladores lineares a partir de um modelo simplificado de primeira ordem do barco.

No capítulo 5, Simulação do sistema de navegação do barco autónomo, simula-se o sistema através do modelo e controlador propostos do capítulo anterior. Para tal utiliza-se o programa MATLAB/SIMULINK, para construir o diagrama de blocos equivalente do sistema de navegação do barco e, com o valor das constantes dimensionadas, obtém-se os resultados da simulação, guardando os dados de direção, posição, velocidade de navegação do barco e velocidade de rotação ao longo do tempo. Também se simula o funcionamento do barco autónomo em diversos modos de funcionamento, verificando se o controlo responde a variações no sistema.

No capítulo 6, Implementação e testes do barco autónomo, faz-se o projeto do barco autónomo, realizando a escolha dos materiais utilizados para a construção do sistema do barco autónomo, e o projeto dos circuitos eletrónicos. Para avaliar o desempenho são realizados testes a cada parte do sistema, verificando o seu funcionamento, tal como testes globais, numa piscina, num lago e no mar, para analisar resultados e comparar com a simulação e teoria.

No capítulo 7, Conclusões, é feito um sumário do trabalho realizado, realçando as conclusões mais importantes. Dá-se ênfase ao trabalho realizado na construção de um sistema de navegação autónoma de um barco, e apresentam-se trabalhos de investigação futuros.

Capítulo 1 - Introdução

Em apêndice demonstram-se as deduções realizadas para a obtenção do modelo e parâmetros do barco autónomo, fazem-se os testes para obter os parâmetros do modelo deduzido do barco, apresenta-se o código necessário para a implementação do barco autónomo tal como o código da simulação e disponibilizam-se recursos adicionais, como vídeos, para auxiliar a visualização dos resultados apresentados nesta dissertação.

Capítulo 2

REVISÃO DA BIBLIOGRAFIA

Neste capítulo é realizada uma revisão sobre a bibliografia relevante ao projeto. Descrevem-se os barcos em geral, como é feita a navegação marítima, que parâmetros se monitorizam no mar, que tipos de sensores são utilizados nos barcos, como é estabelecida a comunicação no mar e numa rede de sensores, como funcionam os motores dos barcos e respetivos sistemas de acionamento, quais os métodos de controlo utilizados nos barcos e exemplos de como é realizada a navegação autónoma em outros barcos. Dando, assim, a contextualizar os dispositivos e métodos utilizados para o desenvolvimento do projeto.

2.1 Barcos

Os barcos são instrumentos de navegação à superfície aquática em vias navegáveis, como por exemplo rios, lagos, mares e oceanos, contendo uma grande variedade de tipos e tamanhos dependentes do seu objetivo, como por exemplo transporte de pessoas e mercadoria, depositar isco para peixes, pescar e, no caso particular desta dissertação, realizar navegação autónoma.

Os barcos podem ser categorizados pelos seus meios de propulsão, que podem ser propulsionados por velas, a remo, mas mais importante e mais utilizado atualmente é por motores, que podem ser de combustão, por vapor ou elétricos [1].

Os barcos necessitam de ter um controlo sobre a sua direção e velocidade linear, atuando nos meios de propulsão, por exemplo abrindo e fechando velas, modificando o local onde é aplicada a propulsão ou variando o nível de rotações dos motores [1].

Os barcos precisam também de um sistema de comunicação entre o comandante e o acionamento dos meios de propulsão, podendo transmitir um sinal elétrico com um periférico *joystick* e com um circuito elétrico, para atuar nos motores, ou através da rotação de um leme variar a posição das velas, ou transmitindo os dados de controlo através de um telecomando para controlar remotamente o barco [1].

Um barco quanto mais peso tiver, para a mesma estrutura, mais fundo entra na água. Por isso a importância de baixar ao máximo a densidade do barco de maneira a que se torne mais estável, possa levar mais carga e não afunde. Para isso um barco contém grandes áreas com espaço vazio de maneira a garantir a sua flutuabilidade [1].

Existem várias classes de barcos de maneira a garantir a segurança tendo em conta os diferentes meios navegáveis definidos pela *Recreational Craft Directive* da União Europeia [2]. A classe A é autorizada a navegar em quaisquer águas, classe B é limitada a navegar no mar, a classe C limitada a navegar na costa marítima e classe D limitada a navegar em rios, canais e lagos. Cada classe implica que o barco garanta a estabilidade para várias condições de vento e ondas.

Capítulo 2 – Revisão da bibliografia

Os tipos de barcos mais importantes para este trabalho são os autónomos e por isso apresentar-se-á vários exemplos de barcos autónomos, explicando como funcionam nos seguintes parágrafos.

No trabalho [3] foi criado um barco autónomo denominado de *Roboduck-II*, esse barco tem 213 centímetros de comprimento e 71 centímetros de largura e tem uma altura no mastro de 104 centímetros, com um peso de 43 quilogramas. São utilizados dois motores DC (corrente contínua) capazes de o levarem a uma velocidade máxima de 1,55 metros por segundo. Utiliza, também, um comando manual pré-fabricado para telecomandar o barco. Esse trabalho teve como objetivo a criação de um barco autónomo para navegação em lagos. A estrutura do barco é demonstrada na Figura 2.1.



Figura 2.1 – Exemplo de barco autónomo *Roboduck-II* [3].

No trabalho [4] foi criado um barco autónomo denominado de *AquaSV*, esse barco é formado por dois flutuadores, para criar uma estrutura catamarã, utilizando dois caiaques de plástico *Bic Ouassou* que suportam uma carga máxima de 110 quilogramas.

Posteriormente, foi criada uma estrutura metálica para armazenar o circuito lógico do sistema, utilizando alumínio anodizado e algumas placas de aço galvanizado. Também é de notar que na utilização das partes metálicas no mar, estas têm de ser protegidas contra a corrosão causada pela água e por isso é que foi utilizado o metal referido anteriormente. São utilizados dois motores DC da Yamaha [5] para propulsionar o barco com uma força equivalente a 13,6 quilogramas para cada motor e o controlo manual para testar o sistema é

realizado em cima do barco. Esse trabalho teve como objetivo a criação de um barco autónomo para monitorização ambiental. A estrutura do barco está na Figura 2.2.



Figura 2.2 – Exemplo de barco autónomo AquaSV [4].

2.2 Navegação

Na navegação manual de um barco tem-se um local destino, definido pelas suas coordenadas, e o local atual do barco. O comandante vira e acelera o barco para atingir uma velocidade linear pretendida na direção do local de destino, necessitando de saber qual a diferença em graus entre a direção pretendida e a atual e qual a diferença entre a velocidade linear atual e a pretendida, corrigindo qualquer desvio. O comandante atua nos periféricos do barco para aplicar uma diferença no meio de propulsão que varia a velocidade linear e a direção atual do barco.

Por exemplo, a navegação de uma canoa tem como meio de propulsão uma pá e o comandante controla a canoa através da força e frequência com que empurra a água, sendo que a velocidade linear se controla empurrando de um lado do barco e depois do outro. Para controlar a direção o comandante empurra mais de um lado do que de outro. O comandante de uma canoa estima qual a

Capítulo 2 – Revisão da bibliografia

velocidade e direção com que pretende navegar e propulsiona a canoa nessa direção com essa velocidade, como mostra a Figura 2.3.

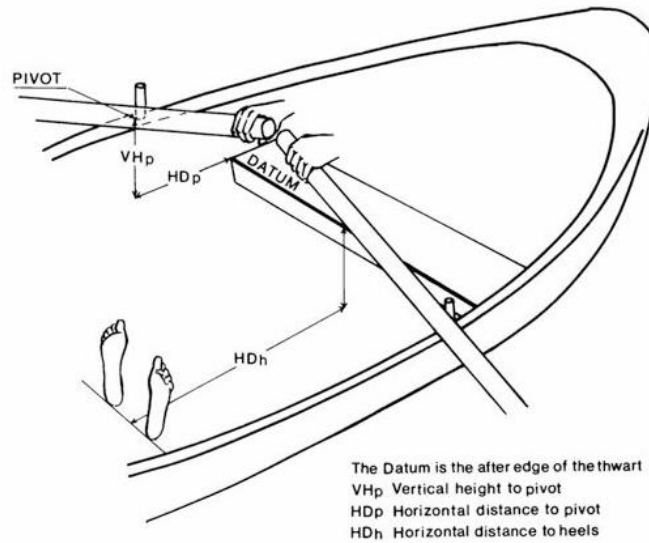


Figura 2.3 – Diagrama ilustrando a geometria da propulsão de uma canoa [1].

A navegação de um barco à vela tem como meio de propulsão o vento, que empurra a vela do barco causando movimento e o comandante controla, através de um leme, a direção entre a velocidade do vento e a vela, que influencia na velocidade do barco e na direção, e também controla o quão hasteadas estão as velas, controlando a velocidade do barco, através da variação de quantidade de vento que empurra o barco [1]. A Figura 2.4 demonstra os vários elementos do barco à vela.

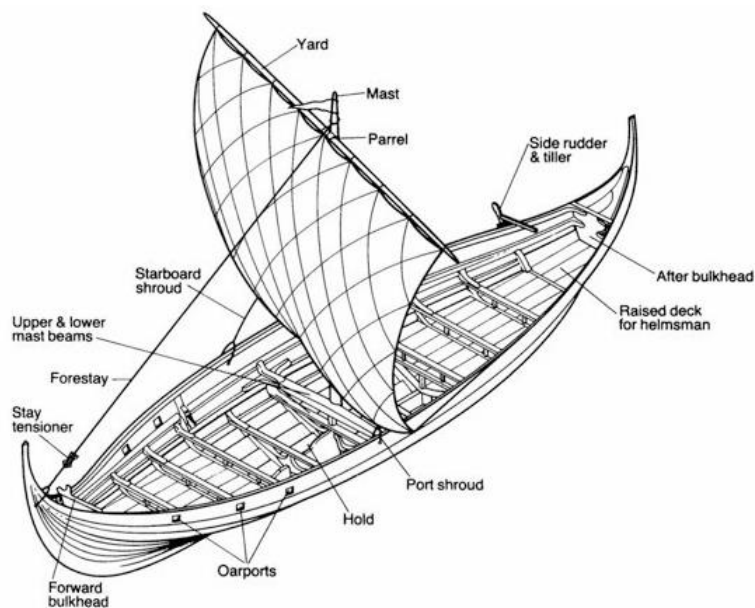


Figura 2.4 – Elementos de um barco à vela [1].

Capítulo 2 – Revisão da bibliografia

A navegação de um barco a motor tem como meio de propulsão turbinas e o comandante controla a velocidade de rotação do barco através de um leme e a velocidade através de um *joystick*. Esses periféricos enviam um sinal para um processador que indica aos motores como atuar através de um *driver* e garante que a velocidade de rotação e a velocidade do barco acompanham a referência através de um sistema de controle. Existem duas maneiras de atuar na propulsão do barco a motor, ou roda-se o motor e a diferença de ângulo cria a rotação, como mostra a Figura 2.5, ou a diferença de potência nos motores causa a rotação, como mostra a Figura 2.6.



Figura 2.5 – Barco a motor com atuação no ângulo [6].



Figura 2.6 – Barco a motor com atuação na diferença de potência de dois motores [7].

2.3 Monitorização

A monitorização dos oceanos utiliza diversos sensores para a obtenção de dados que podem ter por objetivo prever eventos futuros, controlar a qualidade da água, prever o tempo com meteorologia ou procura de objetos, pessoas ou animais, para proteção, observação e resgate. Tendo em conta estas aplicações a amostragem e interpretação dos dados marítimos é uma indústria valiosa e o estudo sobre o processo de interpretação dos dados é importante para este trabalho.

No livro [8] utilizam-se os dados de temperatura, pH, condutividade, oxigenação, e quantidade de sólidos dissolvidos ou em suspensão, para analisar a qualidade da água. As variações de temperatura afetam a toxicidade da água porque, tanto para temperaturas altas ou baixas, a dissolução de certos metais devido à variação da temperatura torna a água mais poluída. O pH da água afeta a sua solubilidade e por isso ao baixar o pH da água aumenta a toxicidade pois o nível de contaminação por certos metais aumenta. O aumento da salinidade diminui a toxicidade da água pois diminui a solubilidade da água. Com isto em conta, ao arranjar sensores que amostram os dados anteriores, consegue-se monitorizar a qualidade da água.

No livro [9] é descrito o processo de previsão do tempo através da utilização de modelos matemáticos computacionais, observando os padrões de evolução de diversas variáveis ao longo do tempo e usando, por exemplo, inteligência artificial para prever o comportamento no futuro, tendo em conta os dados de temperatura, humidade, ondulação, pressão atmosférica e marítima. Através da monitorização da velocidade e da direção do vento em várias localizações consegue-se prever a evolução da temperatura ao longo do dia, a quantidade de precipitação, a formação de tempestades, entre outros fenómenos.

No artigo [10] utilizou-se um sensor de imagem infravermelho para detetar um objeto no mar, sendo que o mesmo princípio pode ser utilizado para detetar pessoas em operações de busca e salvamento, baleias e golfinhos ou navios. As ondas marítimas causam perturbações no algoritmo de deteção de objetos, sendo o objeto detetado através do contraste que faz em relação ao mar plano.

As ondas têm um padrão gradiente muito específico e é isso que é filtrado nesse trabalho, de maneira a obter somente a monitorização do objeto. Logo, com sensores de imagem, conseguem-se fazer buscas por objetos, animais ou pessoas no mar.

2.4 Sensores

Um sensor é um dispositivo que produz um sinal com o propósito de detecção e quantificação de um fenómeno físico. Os sensores detetam eventos ou mudanças no ambiente e enviam a informação para outros dispositivos eletrónicos, normalmente para um processador de um computador para gravação, leitura ou interpretação dos dados [11].

Os sensores não conseguem replicar idealmente a função de transferência, daí terem diversos parâmetros que descrevem as suas características, como a sensibilidade, ruído, saturação, resolução, precisão, entre outros.

Os parâmetros mais importantes dos sensores são a exatidão e a precisão. A exatidão, indica quanto o valor do sensor está perto do valor real e a precisão indica qual a variação dos dados do sensor ao medir o mesmo valor.

Um exemplo de sensor é o GPS (*Global Positioning System*) [12]. O GPS é um sensor de posição global baseado em comunicações com satélites, obtendo dados sobre a localização e tempo no recetor. O sistema GPS recebe, periodicamente, pacotes com informação sobre o tempo do satélite. Com isso calcula quanto tempo demorou entre a comunicação do satélite e o recetor GPS. Obtém-se uma superfície esférica de possibilidade de localização, com uma certa variação.

Através do conhecimento da velocidade de transporte da informação, aproximadamente igual à velocidade da luz, obtém-se a distância do recetor ao satélite, como mostra a Figura 2.7. Tendo em conta essa característica, quantos mais satélites o recetor GPS detetar, mais preciso é o conhecimento da localização, realizando a interseção entre todos os espaços possíveis que correspondem a cada distância ao satélite, como mostra a Figura 2.8 [12].

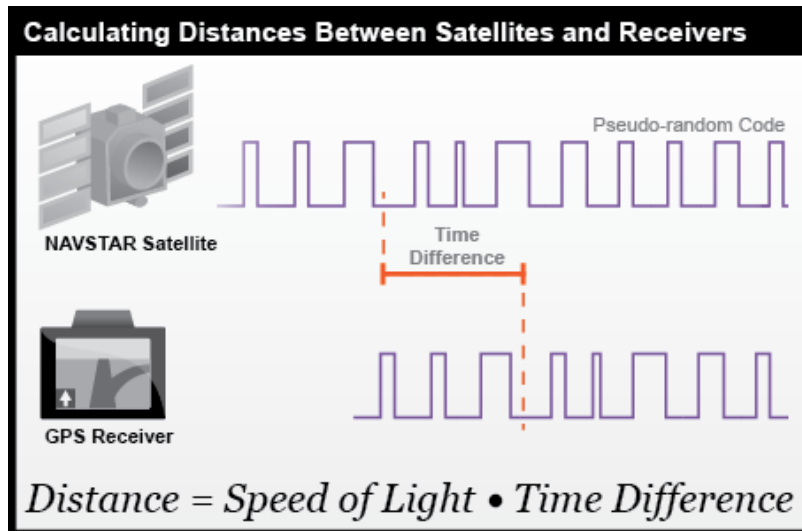


Figura 2.7 – Estimação da distância entre o satélite e recetor através da amostragem do tempo [13].

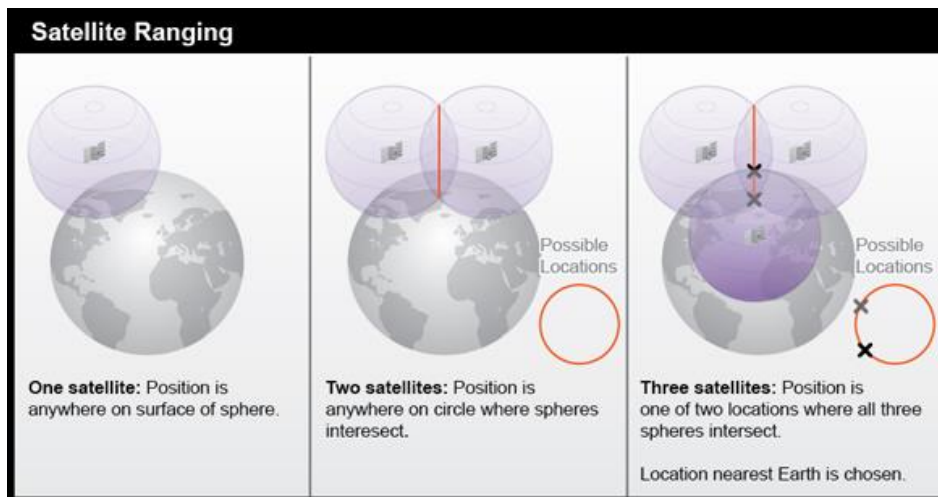


Figura 2.8 – Obtenção de coordenadas GPS através da interseção dos dados de distância entre cada satélite [13].

Outro exemplo de sensor é o magnetómetro, que utiliza o efeito de Hall para medir a intensidade do campo magnético numa dada direção. Num sensor efeito de Hall, os eletrões movem-se dentro de uma placa condutora plana inserida num campo magnético. A placa possui dois contactos laterais ligados a um voltímetro. São colocados dois outros contactos nas extremidades superior e inferior da placa ligados a uma fonte. Devido ao campo magnético a força de deflexão desloca os eletrões em movimento para um lado da placa que se torna mais negativo que o outro lado, produzindo uma diferença de potencial transversal, sendo que o sinal e amplitude deste potencial dependem da magnitude do campo magnético naquela direção e da corrente elétrica, como mostra o diagrama da Figura 2.9 [11].

Capítulo 2 – Revisão da bibliografia

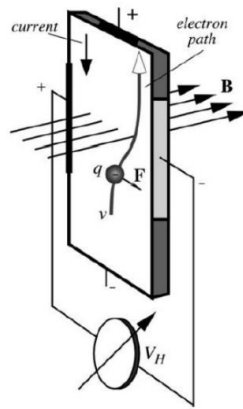


Figura 2.9 – Magnetômetro com um sensor de efeito de Hall [11].

Um acelerômetro utiliza molas ligadas a uma massa para converter a aceleração em deslocamento. Para amostrar esse deslocamento pode-se utilizar, por exemplo, um condensador variável. Deslocando uma das placas varia o valor da capacitância e depois com um circuito que converte capacitância em tensão consegue-se medir a intensidade da aceleração, numa direção, em função da diferença de potencial, como mostram as Figura 2.10 e Figura 2.11 [11].

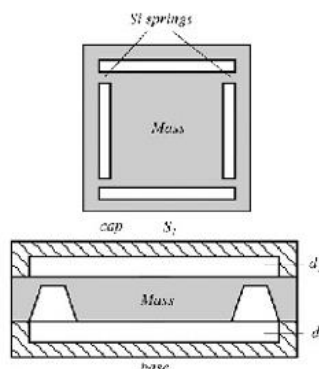


Figura 2.10 – Condensador de capacitância variável em distância [11].

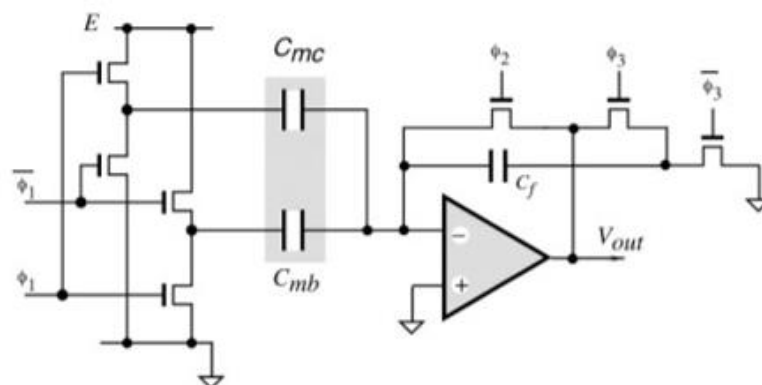


Figura 2.11 – Circuito de conversão de capacitância para tensão [11].

Um giroscópio ótico utiliza o desvio de Doppler para obter uma frequência de sinal equivalente a uma velocidade de rotação [11]. Para tal é enviado um sinal em direções opostas, sendo que um dos sinais terá de percorrer uma maior distância, pois à medida que o sinal se propaga a rotação roda a fibra no mesmo sentido efetivamente aumentado a distância, e o outro sinal terá de percorrer uma menor distância, fazendo com que eles cheguem no misturador com frequências diferentes. O misturador funciona como um diferencial nesta configuração, eliminando o sinal em modo comum e enviando o sinal diferencial, causado pela velocidade de rotação, como mostra a Figura 2.12. Seguidamente, é usado um circuito elétrico para converter o sinal luminoso em tensão, para medir a velocidade de rotação em função de uma diferença de potencial [11].

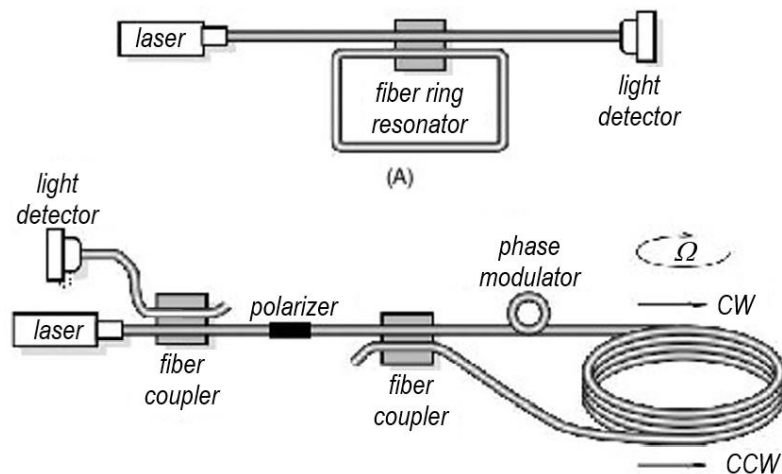


Figura 2.12 – Circuito ótico para determinação da velocidade de rotação [11].

Uma câmera pode ser vista como um sensor universal, pois através da amostragem de imagens consegue-se interpretar as variáveis que se pretendem monitorizar, através do mesmo princípio pelo qual a nossa visão consegue observar os fenómenos a acontecerem. Por exemplo, em operações de busca e salvamento, os nossos olhos estão constantemente à procura da pessoa a ser resgatada e ao interpretar as imagens capturadas por uma câmera, através de por exemplo inteligência artificial, é efetivamente realizado o mesmo processo.

Uma câmera é composta por um conjunto de sensores de imagem que realizam o processo de conversão de energia dos fotões de um feixe de luz para energia elétrica e para tal utiliza-se silício pois a energia necessária para transitar os eletrões de estado é parecida à energia dum fotão na gama da luz visível. Logo ao expor silício à luz consegue-se obter uma corrente equivalente à

intensidade de luz capturada utilizando um fotodíodo [14]. Para conseguir capturar na totalidade uma imagem aplicam-se filtros para capturar feixes de luz com as três cores básicas, vermelho, verde e azul e ao juntar essas cores, utilizando um certo padrão de fotodíodos, recupera-se a imagem, como mostra a Figura 2.13.

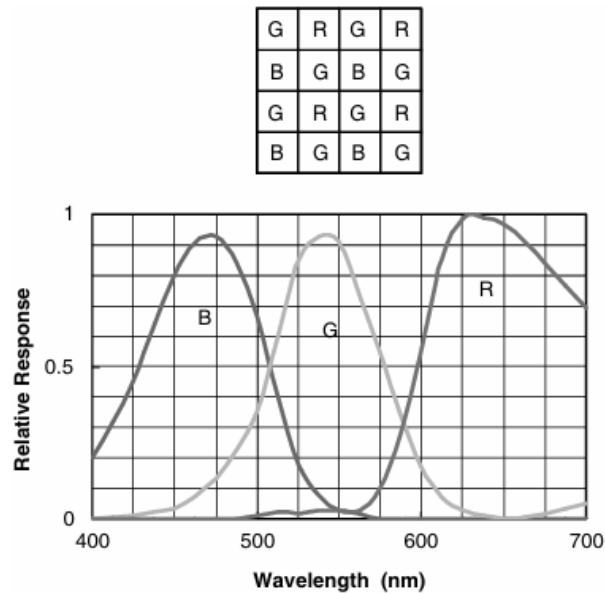


Figura 2.13 – Distribuição do espectro visível com um padrão de fotodíodos RGB [14].

A temperatura é uma variável que afeta muitas características de vários sensores, podendo muitos sensores serem também utilizados para amostrarem a temperatura. De maneira a simplificar a amostragem, normalmente, utiliza-se um termistor, que varia o valor da resistência elétrica em função da temperatura, e através de um divisor de tensão, por exemplo, obtém-se uma tensão variável em função da temperatura [11].

2.5 Comunicação

Um sistema de telecomunicações permite a transmissão de informação a longas distâncias através de meios eletrônicos, por exemplo através de cabos coaxiais, radiofrequência, satélite, fibra ótica ou outras tecnologias de comunicação. Estes métodos de transmissão podem ser divididos em relação às suas características de taxa de transmissão de dados, mobilidade, probabilidade de erro, distância máxima que garante qualidade de serviço, modulação, múltiplo acesso, multiplexagem, entre outros. Um sistema de telecomunicações é

composto por um transmissor, que converte a informação em um sinal, um canal físico, que transporta o sinal enviado pelo transmissor, e um recetor, que recebe o sinal do canal físico e o volta a converter em informação útil [15].

Os sistemas de telecomunicações nos barcos, para o contexto dos dados transmitidos, precisam de ter as seguintes características: o sistema tem de permitir variações na posição do transmissor ou recetor, para o barco se mover e continuar a existir comunicação; precisa de garantir qualidade de serviço para distâncias na ordem de quilómetros, para ter um sistema de transmissão direta do comando, ou seja, sem necessitar de retransmissão; a taxa de transmissão de dados e comandos é baixa, na ordem dos segundos, pois a taxa de amostragem dos dados, no qual aplica-se controlo precisa de ser mais rápida que a atuação, é na ordem das centenas de milissegundos, e a velocidade do sistema de navegação, que pode demorar horas entre percursos, é igualmente baixa; caso exista controlo manual por telecomando a taxa de transmissão de dados aumenta para a ordem dos milissegundos, para não ter latência na ordem da taxa de atualização do controlo dos motores, na ordem das centenas de milissegundos.

Neste contexto, os sistemas mais adequados para as telecomunicações num barco são de baixa radiofrequência ou por satélite, tendo em conta que são sistemas móveis cuja lentidão na resposta não afeta o funcionamento do barco e garantem qualidade de serviço a longas distâncias [16] [17].

Assim, os sistemas padrão de telecomunicações marítimas utilizam rádios FM (*frequency modulation*) nos canais VHF (*very high frequency*), cuja frequência varia entre 156 MHz a 174 MHz, tal como definido pela ITU (*International Telecommunication Union*). A potência de transmissão é limitada a 25 W o que geralmente permite comunicações a distâncias de 100 km. Isso permite comunicação de voz direta sem nenhum sistema adicional. Basta transmitir a onda sonora num canal e se outro barco estiver a escutar essa frequência consegue ouvir a transmissão. Para adicionar serviços extra, como identificação e alertas para resgate são utilizados outros padrões como o AIS (*Automatic Identification System*) e DSC (*Digital Selective Calling*) [18].

Os módulos *Xbee Sub-1 GHz* [19] são um exemplo de rádio de baixa radiofrequência, com sensibilidade de receção de -110 dBm, frequência de

transmissão entre 800 MHz a 900 MHz e potência de transmissão de 13 dBm. Isto permite com que em espaço livre e linha de vista o rádio consiga comunicar na ordem de 10 km. Na prática, tendo em conta que a antena do barco está muito perto do plano de massa, estima-se que se consiga comunicar entre 100 m a 1 km [19]. A vantagem dos *Xbee* é que com a plataforma XCTU consegue-se configurar o módulo em diversos modos de funcionamento e é capaz de realizar diversos testes [19]. Na Figura 2.14 é apresentado um exemplo de um rádio *Xbee*.



Figura 2.14 – Rádio *Xbee* [19].

O módulo *LoRa SX1276* [20] é outro exemplo de um rádio de baixa radiofrequência, com sensibilidade de receção de até -148 dBm, frequência de transmissão entre 137 MHz a 1020 MHz e potência de transmissão máxima de 20 dBm. Isto permite que em espaço livre e linha de vista o rádio consiga comunicar na ordem de 100 km [20]. Na prática estima-se que se consiga comunicar entre 1 km a 10 km. A vantagem dos rádios *LoRa* é a modulação da mensagem que permite o longo alcance [20]. Na Figura 2.15 é apresentado um exemplo de um rádio *LoRa*.

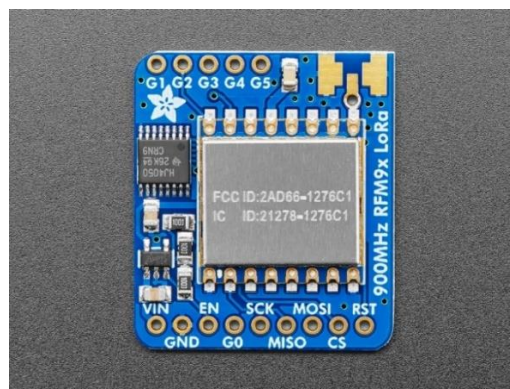


Figura 2.15 – Rádio *LoRa* [21].

O módulo *nRF905* [22] é um rádio de baixa radiofrequência, com sensibilidade de recepção de -100 dBm, frequência de transmissão entre 433 MHz a 915 MHz e potência de transmissão de 10 dBm. Isto permite que em espaço livre e linha de vista o rádio consiga comunicar na ordem dos 10 km [22]. Na prática estima-se que se consiga comunicar na ordem de 1 km, tendo em conta imperfeições. A vantagem do rádio *nRF905* é a sua simplicidade, uma vez que não tem funcionalidades extra, tem tudo de ser programado, o que permite um acesso direto com maior velocidade [22]. Na Figura 2.16 é apresentado um exemplo de um rádio *nRF905*.



Figura 2.16 – Rádio *nRF905* [23].

2.6 Motores

Os motores são máquinas que convertem um tipo de energia em energia mecânica, sendo maioritariamente utilizado como um intermediário na produção de energia elétrica através de outras fontes de energia. As fontes de energia podem ter diversas origens, como: energia potencial, exemplificada pelos motores utilizados na geração de energia hidroelétrica; energia térmica, como nas centrais geotérmicas; energia química, obtida através de reações químicas, como a combustão interna; energia potencial elétrica, que pode ser gerada utilizando bobinas e aplicando uma corrente variável, criando uma variação de fluxo magnético que resulta em movimento; e energia nuclear, como a proveniente da fissão nuclear [24].

Capítulo 2 – Revisão da bibliografia

Na física Newtoniana, o binário motor τ é dado pelo momento de inércia, J , vezes a aceleração angular, que é a derivada da velocidade de rotação em função do tempo, $\frac{d\Omega}{dt}$, como mostra a equação (2.1) [25]:

$$\tau = J \frac{d\Omega}{dt} \quad (2.1)$$

A partir desta equação adiciona-se um termo de perdas devido à resistência do ar b e decompõe-se o binário em binário gerado τ_g e binário de carga τ_c , obtendo assim a equação clássica que descreve a variação da velocidade de rotação em função do binário motor gerado, dada pela equação (2.2) [26].

$$J \frac{d\Omega}{dt} = \tau_g - \tau_c - b\Omega \quad (2.2)$$

Esta equação é aplicada quando o motor está a rodar no ar. Mas como a viscosidade da água é muito maior que a do ar, o termo de perdas não é modelado corretamente pela equação anterior, daí ser necessário um novo modelo para quando o motor estiver a rodar dentro da água. Por exemplo o modelo proposto por Yoerger [27], que diz que o termo de perdas é dependente da velocidade de rotação ao quadrado, como mostra a equação (2.3), onde β e α são constantes dependentes das características da turbina e τ é o binário motor aplicado, ou seja, o binário motor gerado menos o binário motor de carga [27].

$$\frac{d\Omega}{dt} = \beta\tau - \alpha\Omega|\Omega| \quad (2.3)$$

Um motor de combustão é um motor baseado na geração de energia térmica através de energia química, onde acontece a combustão de um combustível com um oxidante, queimando o combustível e gerando calor e pressão, que aplica uma força mecânica no motor, fazendo-o rodar. Para tal existe um tubo de injeção do combustível no motor e um tubo de escape, que abrem e fecham as suas válvulas, através da variação da pressão aplicada, para controlar a rotação do motor, dependente do fluxo de massa do ar com combustível. Os sensores de ar, combustível, velocidade de rotação do motor, pressão, temperatura e velocidade do fluido servem para aplicar controlo dessas próprias variáveis do motor com *feedback*. Quanto maior o combustível injetado por ciclo for, maior o binário motor produzido pelo motor, pois é aplicada mais

força no motor devido ao aumento do produto pressão com volume deslocado, e quanto mais rápido for o ciclo de injeção do combustível e escape dos gases, maior será a velocidade de rotação do motor, pois estão sincronizados [28]. Na Figura 2.17 visualiza-se o diagrama deste tipo de motor de combustão.

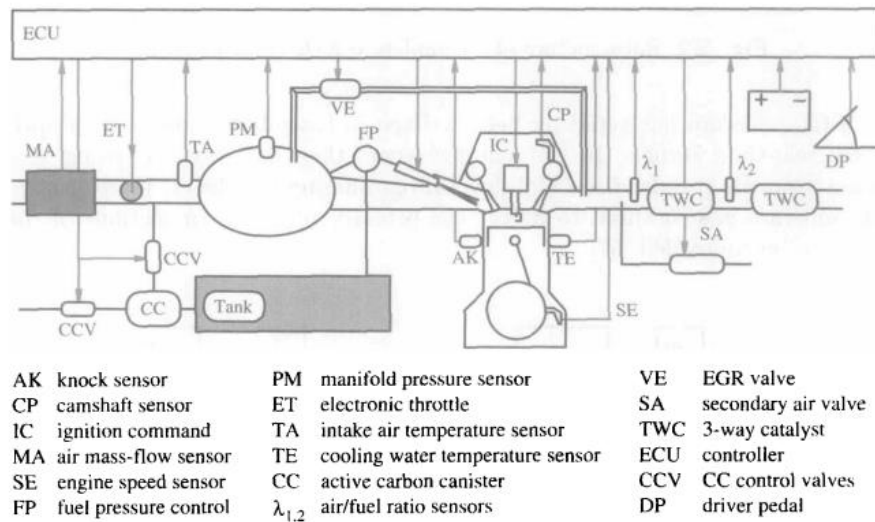


Figura 2.17 – Estrutura típica de um sistema de motor de combustão [28].

Um motor elétrico é um motor que converte energia elétrica em energia mecânica. A rotação do motor é causada pela passagem de corrente num enrolamento, através da força de Lorentz dada pela equação (2.4), considerando que o campo elétrico no condutor do enrolamento é nulo, sendo F a força, i a corrente, l o comprimento do condutor perpendicular ao campo magnético B e \times o produto externo [29].

$$F = il \times B \quad (2.4)$$

Um motor elétrico utiliza bobinas que geram um campo magnético através da passagem de uma corrente, dependente da indutância L , área de secção do fio S e número de voltas do fio N , como mostra a equação (2.5) [29].

$$Li = N * B * S \quad (2.5)$$

Os motores elétricos têm diferentes características dependendo da forma de onda do sinal enviado e da estrutura metálica. Os três principais tipos de motores elétricos são: motores AC (corrente alternada) síncronos e assíncronos e motores DC [30].

Capítulo 2 – Revisão da bibliografia

Os motores AC assíncronos têm o enrolamento do rotor curto-circuitado, fazendo com que as correntes no rotor, que provocam o movimento, sejam induzidas pela variação do campo magnético criado pelas correntes no estator.

O modelo de circuito elétrico equivalente é apresentado na Figura 2.18 [26], onde s é o deslizamento, ou seja, a diferença relativa entre frequências do estator e do rotor, V_1 é a tensão aplicada nos terminais do motor, R_1 é a resistência do estator, X_1 é a reactância de fugas, I_1 é a corrente do estator, I_ψ é a corrente de magnetização, R_c é a resistência de perdas, X_m é a reactância de magnetização, E_2 é a força eletromotriz induzida, X_2 é a indutância do rotor e R_2 é a resistência do rotor [26].

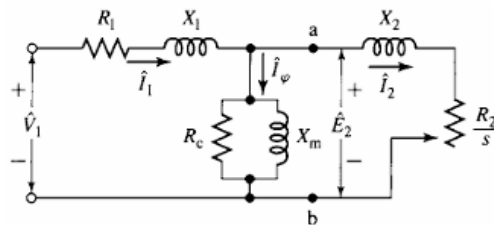


Figura 2.18 – Circuito equivalente de um motor AC assíncrono [26].

Os motores AC síncronos precisam de ter as bobinas do rotor e estator alimentadas externamente, necessitando que exista sincronismo entre as correntes do estator e do campo magnético do rotor para que um campo magnético siga o outro. A velocidade de rotação do motor é igual à frequência equivalente do sinal de entrada, tendo em consideração o número de pólos.

A passagem de correntes no rotor e no estator causa um efeito de indução que influencia o sistema. Esse efeito causa variações no valor de indutância das bobinas e diferença de tensão, como mostra o modelo de circuito elétrico equivalente da Figura 2.19, onde V_a é a tensão AC aplicada, X_s é a indutância do motor, R_a é a resistência de perdas do motor, I_a é a corrente do motor e E_{af} é a força eletromotriz induzida [26].

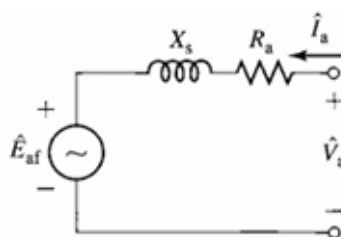


Figura 2.19 – Circuito equivalente de um motor AC síncrono [26].

Capítulo 2 – Revisão da bibliografia

Os motores DC têm como princípio de funcionamento a indução de um campo magnético constante que para fazer o motor rodar, alterna entre conduzir e não conduzir corrente, através de escovas que fecham o circuito caso toquem num material, comutador. Assim, alternar a alimentação dos enrolamentos do rotor faz com que mesmo fornecendo uma tensão DC, consiga-se rodar o motor [26]. O motor DC precisa de ser magnetizado no estator através da ligação à fonte DC para seguir o campo magnético girante.

Ao rodar o motor, acontece um efeito de indução mútua que baixa a queda de tensão na bobina, dependendo da velocidade de rotação. O modelo de circuito elétrico equivalente na Figura 2.20, onde V é a tensão DC de entrada na armadura, i_a é a corrente da armadura, R_a é a resistência de perdas na armadura, L_a é a indutância na armadura, V_f é a tensão de entrada no estator, R_f é a resistência do estator, L_f é a indutância do estator, i_f é a corrente do estator e e é o efeito da força eletromotriz induzida [30].

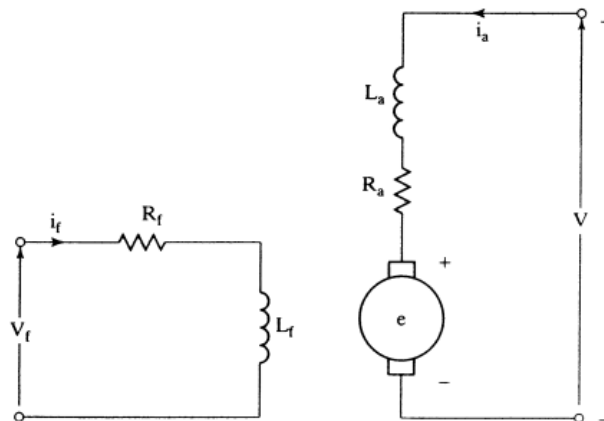


Figura 2.20 – Circuito equivalente de um motor DC [30].

Também é de notar que o binário gerado pelos motores elétricos é dado por uma constante a multiplicar pelas correntes que passam na armadura e estator [26].

Um exemplo de um motor elétrico DC para um barco é o Yamaha M12 [5], que utiliza um eixo de rotação pequeno, para maximizar a velocidade de rotação que determina a força equivalente aplicada, como mostra o artigo [27], e é apresentado na Figura 2.21.



Figura 2.21 – Motor Yamaha M12 [5].

Um exemplo de um motor de combustão interna para um barco é o Honda BF2.3 [31], que é um pouco maior que o motor elétrico Yamaha M12, mas consegue fornecer muito mais potência, 1,7 kW comparando com 300 W do motor M12. No entanto têm-se menos controlo sobre o motor, sendo esse motor apresentado na Figura 2.22.



Figura 2.22 – Motor Honda BF2.3 [31].

2.7 Sistemas de acionamento

Para poder controlar os motores DC é necessário um circuito de acionamento (*driver*) de motor, sendo a ponte-H, representada na Figura 2.23, o *driver* mais universal para motores elétricos DC. Para outros motores, nomeadamente os trifásicos, utilizam-se pontes de 3 braços. Este circuito permite o acionamento de vários níveis de tensão na entrada dos motores, podendo ser expandido o conceito também para motores AC. No caso de motores DC, a ponte-H aplica três possíveis níveis de tensão no motor, V_{dc} , 0 V ou $-V_{dc}$. Pode-se aplicar um sinal PWM (*Pulse-Width Modulation*) na entrada desses interruptores para que aparenta na entrada do motor uma tensão média proporcional ao *duty-cycle* que ocupa cada um dos níveis. Com isto pode-se acionar o motor a rodar num sentido e no sentido contrário e com controlo da

velocidade de rotação [30]. Os díodos aparecem no circuito para quando os interruptores estiverem todos abertos e o motor estiver a rodar, a corrente do motor ir para a bateria. Na seguinte Figura 2.23, S_1 , S_2 , S_3 e S_4 são interruptores, V_{dc} é a tensão DC e v_a é a tensão de entrada do motor DC. O par motores DC com ponte-H oferecem uma maior simplicidade no circuito, comparativamente aos outros sistemas, observáveis nas Figuras das secções 2.6 e 2.7.

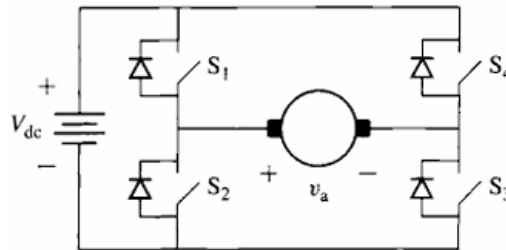


Figura 2.23 – Ponte-H para um motor DC [26].

Aplicando o conceito da ponte-H para os motores AC, obtém-se o conversor de três braços apresentado na Figura 2.24, que implica gerar as sinusóides de entrada em cada fase do motor. Para tal, utiliza-se o mesmo conceito da ponte-H no motor DC através de três níveis de tensão em cada fase, V_{dc} , 0 V ou $-V_{dc}$, consegue-se controlar a amplitude da tensão aplicada em cada fase através do *duty-cycle*, atuando nos transístores T_1 a T_6 , utilizando um condensador para balancear a tensão e, assim, no lado DC o sistema fornece corrente contínua e no lado AC o sistema consome corrente alternada. Os díodos aparecem no circuito para quando os interruptores estiverem todos abertos e o motor estiver a rodar, a corrente do motor ir para a bateria. Para construir a senoide de cada fase, varia-se o valor do *duty-cycle* ao longo do tempo em cada uma das fases de maneira a seguir a forma de onda [30].

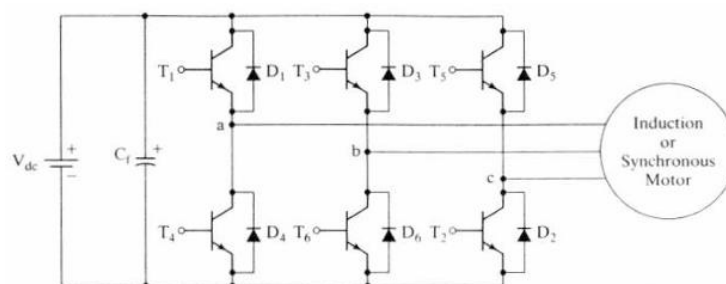


Figura 2.24 – Ponte de três braços para acionamento de motor AC [30].

Para motores de combustão interna, podem ser utilizados dois servos para controlar a quantidade de mistura de ar e combustível que entra e sai no motor, uma ignição para realizar a combustão e mais um servo para controlar o rácio de mistura entre ar e combustível, abrindo mais ou menos a entrada do combustível [28].

Os servos podem ser controlados através de *drivers* de motores elétricos e a ignição é acionada por semicondutores eletrónicos de potência. É importante o sincronismo que o sistema de controlo tem de manter para conseguir que o motor rode de maneira controlada. É de ter em conta que para além da velocidade de rotação é controlado o fluxo de ar em rotação. Para impor a velocidade de rotação do motor é amostrada a velocidade de rotação atual e aplicado um sistema de controlo que atua na quantidade de combustível a ser injetada. A pressão dentro do motor é amostrada e controlada, atuando na quantidade de ar injetado e, assim, influenciando o rácio de combustível com o ar. O sistema de escape dos gases é controlado dependendo da quantidade de combustível e ar em função de uma referência de fluxo constante, variando a abertura das válvulas do tubo de escape. Na Figura 2.25 apresenta-se o diagrama de controlo de um motor de combustão interna.

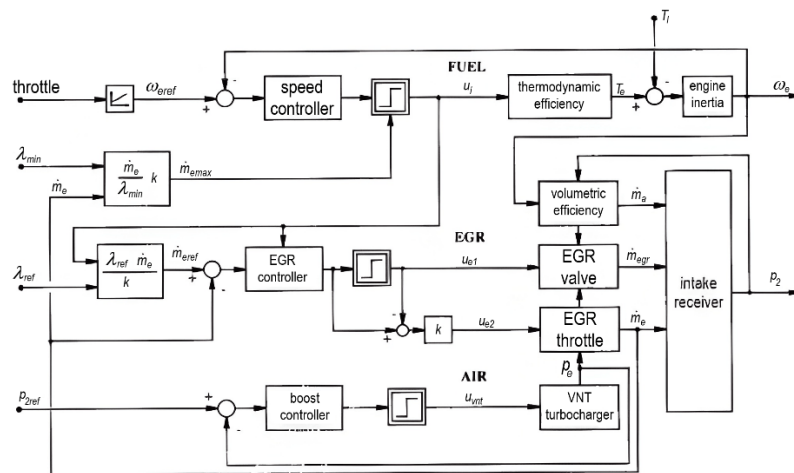


Figura 2.25 – Estrutura de controlo de um motor de combustão interna [28].

2.8 Controlo de barcos autónomos

Um sistema de controlo é um conjunto de dispositivos ou componentes que gerem, comandam, direcionam e regulam o comportamento de outros dispositivos ou sistemas com o objetivo de atingir uma saída desejada [32].

Capítulo 2 – Revisão da bibliografia

Existem dois tipos principais de sistemas de controlo: sistemas de comando em malha aberta e sistemas de controlo em malha fechada. Os sistemas de comando em malha aberta leem uma entrada de referência e enviam um sinal para os atuadores tentarem impor esse valor de referência, sem verificar a saída. Os sistemas de controlo em malha fechada medem a saída, comparam com a entrada de referência e enviam um sinal para os atuadores com o objetivo de tentarem corrigir o erro entre a referência e o valor atual. A vantagem dos sistemas em malha fechada é que ao longo do tempo as características do sistema se modificam, mas como o sistema atua sobre o erro, vai sempre tentar impor o valor do erro a zero, enquanto que em malha aberta as variações das características do sistema podem causar uma mudança na saída [32].

Num sistema de controlo existem quatro componentes principais, a referência, que é o sinal que se pretende que a saída do sistema acompanhe, o sensor, que mede o valor da saída e manda o *feedback* para o controlador, o controlador, que toma decisões tendo em conta a referência e o valor da saída e o atuador, que afeta o valor da saída através dos comandos enviados pelo controlador [32].

Para representar os sistemas de controlo utiliza-se normalmente a transformada de *Laplace*, com o objetivo de facilitar a visualização do diagrama de blocos e analisar a estabilidade e reação do sistema aos estímulos.

Nos barcos, o sistema de controlo da direção é responsável por aplicar uma diferença de potência nos motores ou rodar a posição dos motores, para impor uma velocidade de rotação no barco. Essa velocidade de rotação provoca uma variação da direção do barco ao longo do tempo. Para controlar este efeito amostra-se a direção e compara-se com a pretendida (o valor de referência). O sistema de controlo observa o erro e envia o comando correspondente, proveniente do compensador, para os motores.

Um tipo de controlo não linear utilizado para controlar a direção do barco é o controlo por histerese. Como, por exemplo, é apresentado no trabalho [33], onde o objetivo do controlo autónomo é atracagem do barco numa estação de carregamento e por isso o barco anda sempre para a frente até encontrar um obstáculo. Quando encontra o obstáculo tem de se desviar ou para a esquerda

ou para a direita, dependendo da forma do obstáculo. Para contornar os obstáculos o sistema de controlo por histerese liga um dos motores e desliga o outro, e caso o obstáculo passe a obstruir mais o outro lado, troca o acionamento ligando o outro motor e desligando o que estava ligado. Controlando o barco por histerese, começando por andar para a frente e se estiver um obstáculo próximo, verifica qual o lado que está mais livre e roda o barco para essa direção, como mostra o fluxograma da Figura 2.26. A vantagem deste método de controlo é a simplicidade de implementação.

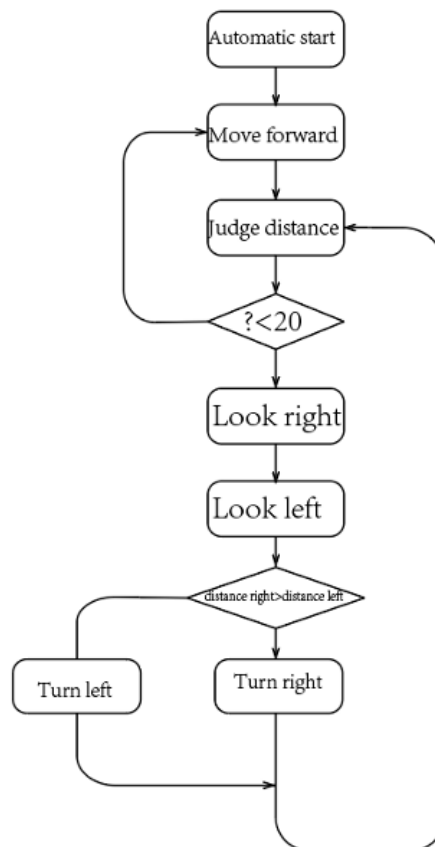


Figura 2.26 – Fluxograma do controlo por histerese de um barco autónomo [33].

Outro tipo de controlo utilizado para controlar a direção do barco é o controlo proporcional-derivativo. Como, por exemplo, é apresentado no trabalho [3], que aproxima o modelo do sistema de velocidade de rotação do barco a um sistema de primeira-ordem. Esse trabalho indica que a não utilização do controlo integral é dependente do facto do sensor de direção, que precisa de estimar essa variável, a partir de um filtro de *Kalman*, causa em certos intervalos de tempo um ruído constante. Logo integrar esse ruído cria mais instabilidade no sistema. O facto das perturbações no sistema empurrarem o barco para o outro lado da

referência mostra a desvantagem da não utilização do controlo integral. A existência de um erro estacionário constante não desvia o barco da trajetória pretendida. Este método de controlo compara o ângulo atual com o ângulo de referência, obtendo um erro que entra no sistema de controlo, multiplicando o erro pelo valor do K_p e subtraindo a esse valor, o valor da velocidade de rotação multiplicado pelo K_d que é o termo derivativo do controlador. Esse sinal de acionamento é a entrada num sistema de primeira-ordem que determina a velocidade de rotação e ao integrar obtém-se o ângulo, como mostra o diagrama da Figura 2.27. A vantagem deste método de controlo é uma maior precisão do acompanhamento da trajetória em comparação com os outros sistemas.

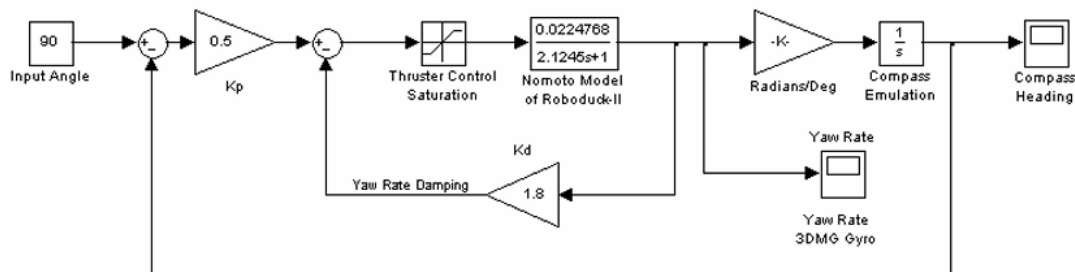


Figura 2.27 – Sistema de controlo proporcional-derivativo da direção do barco [3].

Como outro exemplo de controlo da direção do barco, é usado uma tabela de endereçamento, como é apresentado no trabalho [34]. Para simplificar o sistema assumem-se, somente, os três níveis de acionamento de uma ponte-H, motor ligado para a frente, motor ligado para trás ou motor desligado. Para a construção da tabela aplica-se os comandos para cada motor, M_L e M_R , $\{-1, 0, 1\}$. Para determinar quais os comandos que o sistema de controlo envia é calculado o erro de posição, e_p , e o erro de direção, e_ψ . Tendo em conta isto o sistema de controlo amostra o ângulo de direção do barco, ψ , e a posição do barco, p_x e p_y , aplicando operações matemáticas para calcular o erro de posição e erro de direção. Caso tenha um erro de posição pequeno, ou seja, um valor absoluto menor que 3 metros o barco só tem de corrigir a direção o que implica que os motores têm comandos opostos para não variar a posição do barco. Caso tenha um erro de direção pequeno, ou seja, um valor absoluto menor que 5° , o barco só tem de corrigir a posição o que implica que os motores têm comandos iguais para navegarem na direção pretendida. Nos outros casos para corrigir o

ângulo e direção o controlador só atua no motor correspondente para corrigir tanto o ângulo como a direção, sendo no outro motor enviado o comando de 0. O diagrama de blocos do sistema é apresentado pela Figura 2.28 e é utilizada a Tabela 2.1 para determinar o comando enviado ao barco. Este método é um meio termo entre os dois apresentados anteriormente, tendo maior precisão que a pura histerese e maior simplicidade que a implementação de um controlador linear.

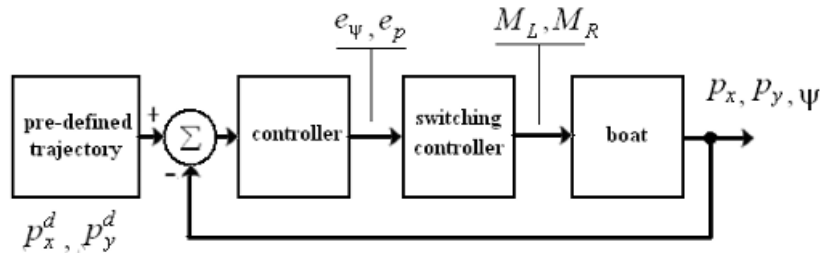


Figura 2.28 – Diagrama de blocos do sistema de controlo recorrendo a uma tabela de endereçamento da direção do barco [34].

Tabela 2.1 – Tabela de endereçamento utilizada para implementar o controlador [34].

	$e_\psi (^{\circ})$			
				M_L, M_R
+90	0 , -1	-1 , 1	1 , -1	0 , 1
+5	-1 , -1	0 , 0	0 , 0	1 , 1
0	-1 , -1	0 , 0	0 , 0	1 , 1
-5	-1 , 0	1 , -1	-1 , 1	1 , 0
-90				
	-3	0	+3	$e_p (m)$

2.9 Navegação autónoma

A diferença entre a navegação manual e a autónoma é a substituição do comandante do barco por um sistema de controlo automático, que deve traçar a rota desejada tendo em conta as suas entradas. Na navegação autónoma de um barco tem-se um local de destino com as coordenadas que definem a trajetória e o local atual do barco. O sistema de controlo automático vira e acelera o barco para uma velocidade pretendida na direção do local destino, necessitando de saber qual a diferença em graus entre a direção pretendida e a atual e qual a diferença entre a velocidade atual e a pretendida, corrigindo qualquer desvio. O

processador atua nos periféricos do barco, através de um sistema de controlo, para aplicar uma diferença no meio de propulsão que varia a velocidade e direção atual do barco.

Para além do sistema de controlo, motores, acionamento, estrutura do barco e sensores é essencial descrever a lógica necessária para, a partir de duas coordenadas GPS, traçar a trajetória enviando os dados de direção pretendida para o sistema de controlo e mudar para o próximo ponto, até chegar ao destino através de uma condição de paragem. Também é possível definir a trajetória de maneira contínua, por exemplo, através da constante atualização da referência de posição entre pontos.

No trabalho [3] utilizam o sistema de coordenadas UTM (*Universal Transverse Mercator projection*), que divide o mundo em 60 zonas verticais, permitindo aproximar a superfície esférica da Terra num mapa, o que permite aplicar diretamente as operações do plano de coordenadas cartesianas nos pontos UTM. Tendo em conta isso, para calcular o ângulo desejado é feito o arco-tangente da diferença entre os dois pontos como mostra a equação (2.6) [35], sendo θ_{ref} o ângulo desejado, $UTME_{ref}$ a coordenada do local destino para este, $UTMN_{ref}$ a coordenada do local destino para norte, $UTME_{atual}$ a coordenada do local atual para este e $UTMN_{atual}$ a coordenada do local atual para norte.

$$\theta_{ref} = \tan^{-1} \left(\frac{UTME_{ref} - UTME_{atual}}{UTMN_{ref} - UTMN_{atual}} \right) \quad (2.6)$$

Para o barco detetar a chegada ao destino, calcula-se a distância através do teorema de Pitágoras somando os quadrados da diferença entre cada coordenada dos pontos e verifica-se se é menor que um raio de aceitação, ρ , ao quadrado, como mostra a equação (2.7) [35].

$$(UTME_{ref} - UTME_{atual})^2 + (UTMN_{ref} - UTMN_{atual})^2 \leq \rho^2 \quad (2.7)$$

Pode-se aplicar a mesma ideia para o sistema de coordenadas GPS, obtendo a direção e a distância entre dois pontos GPS. Para tal aplicam-se as leis de *haversine* [36]. Tendo em conta isso, para calcular o ângulo desejado é calculado o ângulo entre o arco do polo Norte ao ponto atual e o arco do ponto atual ao ponto final, como mostra a equação (2.8). E utiliza-se a equação (2.9)

para verificar se a distância é menor que um raio de aceitação ao quadrado [36]. Sendo φ_1 a latitude do local atual, φ_2 a latitude do local de destino, $\Delta\varphi$ a diferença entre as latitudes do local atual e do local de destino, $\Delta\lambda$ a diferença entre as longitudes do local atual e do local de destino e r o raio da Terra.

$$\theta_{ref} = \tan^{-1} \left(\frac{\sin(\Delta\lambda) \cos(\varphi_2)}{\cos(\varphi_1) \sin(\varphi_2) - \sin(\varphi_1) \cos(\varphi_2) \cos(\Delta\lambda)} \right) \quad (2.8)$$

$$2r \sin^{-1} \left(\sqrt{\frac{1 - \cos(\Delta\varphi) + \cos(\varphi_1) \cos(\varphi_2) (1 - \cos(\Delta\lambda))}{2}} \right) \leq \rho^2 \quad (2.9)$$

2.10 Considerações finais sobre a revisão da bibliografia

Em resumo, para desenvolver o projeto do barco estudou-se, inicialmente, o princípio de funcionamento e navegação dos barcos, observando que os barcos precisam de ter um meio de propulsão, tendo como exemplos velas, pás e motores, uma maneira de controlar a direção e posição do barco, tendo como exemplos o próprio navegador através do local da remada das pás, um leme e um sistema autónomo, e uma maneira de comunicar entre o utilizador e o sistema, tendo como exemplos o leme e o telecomando.

Como o barco é autónomo em vez de o utilizador atuar diretamente na direção e posição do barco, utiliza-se um sistema de controlo automático, que necessita de sensores, para observar as variáveis que se pretendem controlar, atuando num sistema de acionamento para controlar o comportamento dos motores, utilizados como meio de propulsão, daí a importância do estudo desses tópicos.

Juntando, também, a isso o estudo de algoritmos de condições de paragem e de algoritmos de cálculo da direção. Para estabelecer a comunicação com o utilizador, estudaram-se rádios e sistemas de telecomunicações para comunicar no mar. Com o objetivo de permitir a possibilidade de recolha de dados marítimos, estudou-se também a monitorização no mar. Tendo a bibliografia revista, avançou-se com o desenvolvimento da estrutura lógica de programas e circuitos para apresentação de uma solução válida na construção de um sistema do barco autónomo.

Capítulo 3

ESPECIFICAÇÕES DO BARCO AUTÓNOMO

Neste capítulo definem-se as especificações do barco autónomo, os programas com as funcionalidades do barco autónomo e os circuitos do barco para que ele possa atingir os objetivos previamente definidos. São especificados os tipos de controladores utilizados para o controlo de direção e comando de velocidade linear do barco e também é descrita a forma de comunicação entre o telecomando e o barco. São definidas as funcionalidades do barco em navegação controlada por telecomando, em navegação pré-definida e de envio de coordenadas.

3.1 Sistema do barco autónomo

Para o desenvolvimento do sistema do barco autónomo é necessário utilizar diversos materiais. Utiliza-se um telecomando para estabelecer a comunicação entre o barco e o utilizador. Os motores propulsionam o barco, a bateria fornece energia aos motores através dos *drivers* de motor, que transformam um sinal de controlo num sinal de potência aplicado nos motores. Os rádios estabelecem a comunicação entre o telecomando e o barco. Os periféricos de entrada permitem ao utilizador o envio de comandos para o barco. Os sensores de posição e direção são utilizados para controlar a navegação através de *feedback*. Os microcontroladores executam toda a lógica do sistema e interligam as diversas partes do barco autónomo. Foi escolhido um barco elétrico pela sua simplicidade de montagem e interação com o microcontrolador.

O telecomando é constituído pelos periféricos de entrada, microcontrolador e rádio. O utilizador interage com os periféricos de entrada que enviam um sinal correspondente da funcionalidade pretendida para o microcontrolador do telecomando. O microcontrolador interpreta esse sinal e cria uma mensagem para enviar para o rádio transmitir para o barco. O telecomando é alimentado por uma porta *Universal Serial Bus*.

No barco autónomo, o rádio recebe os comandos do telecomando e envia-os para o microcontrolador do barco interpretar e seguir o fluxo de código do comando pretendido. O sensor de posição e o sensor de direção também enviam os dados amostrados para o microcontrolador. O microcontrolador ao receber a informação do comando a executar e os valores de posição e direção do barco calcula o valor do sinal de controlo para enviar ao *driver* de motor, que aciona os motores com o valor correspondente. O barco autónomo é alimentado através da bateria, com um conversor DC/DC para alimentar a parte lógica.

Tendo em conta estas especificações foi criado um diagrama de blocos de maneira a tornar mais visíveis as ligações, relações e funcionamento do sistema que caracteriza o barco autónomo, como mostra a Figura 3.1.

Capítulo 3 – Especificações do barco autónomo

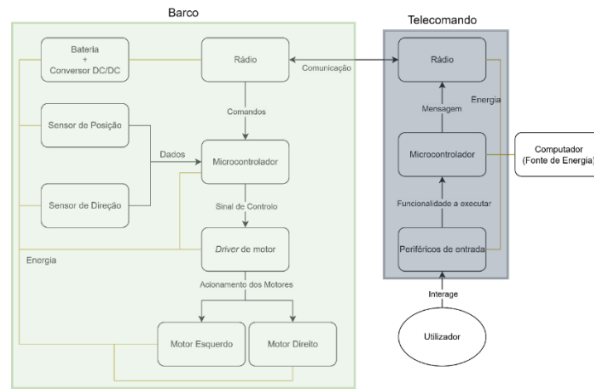


Figura 3.1 – Diagrama de blocos da estrutura geral do barco autónomo.

Tendo a descrição do funcionamento do telecomando, que interage com o utilizador e transmite os comandos, e a descrição do funcionamento do barco, que recebe os comandos, realiza a leitura dos sensores e atua nos motores, especificados. Procede-se para a próxima secção, onde faz-se a descrição do sistema de controlo da direcção do barco autónomo que se pretende programar no microcontrolador.

3.2 Controlo da direcção do barco autónomo

Para o desenvolvimento do sistema de controlo da direcção do barco é escolhida uma direcção pretendida e amostrada a direcção atual, fazendo a subtração angular entre elas para se obter o erro de direcção. Com esse erro o controlador de direcção deve calcular o valor da velocidade de rotação de referência do barco.

Realizando a subtração entre a velocidade de rotação do barco pretendida com a amostrada obtém-se o erro da velocidade de rotação do barco que servirá de entrada para o controlador da velocidade de rotação do barco, que calcula uma variável intermédia diferencial, denominada de μ_{dif} , representativa da diferença entre o *duty-cycle* dos sinais PWM aplicados nos motores, podendo ser vista como a diferença de tensão aplicada entre motores.

Essa variável intermédia diferencial soma num motor e subtrai noutro a uma variável intermédia comum, denominada de μ_{com} , que é calculada pelo comando de velocidade linear. Obtém-se assim um sinal PWM para atuação em cada motor, que entra no modelo físico do barco, como mostra o diagrama de blocos da Figura 3.2. Nesta figura considera-se θ_{ref} a direcção pretendida para o

Capítulo 3 – Especificações do barco autónomo

barco, ω_{ref} a velocidade de rotação do barco pretendida, μ_{dif} a variável intermédia diferencial para cálculo da atuação nos motores, μ_{com} a variável intermédia comum para cálculo da atuação nos motores, $PWM1$ o valor do sinal no motor esquerdo e $PWM2$ o valor do sinal no motor direito. Finalmente, ω é a velocidade de rotação do barco atual e θ a direção atual do barco.

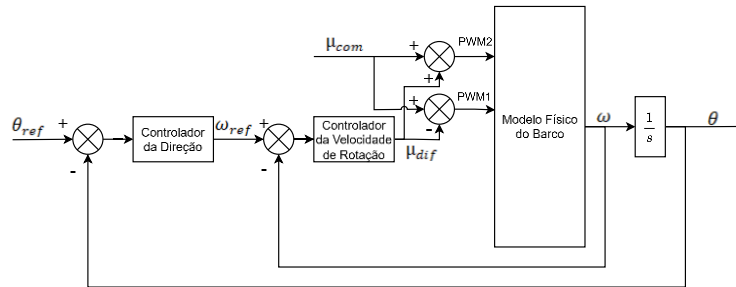


Figura 3.2 – Diagrama de blocos do sistema de controlo autónomo da direção do barco.

Com o controlo de direção do barco autónomo descrito, através da utilização de dois controladores, sendo que o controlador de direção calcula uma velocidade de rotação de referência e o controlador de velocidade de rotação calcula uma variável intermédia diferencial, equivalente à diferença do sinal PWM aplicado entre motores. Procedeu-se para a descrição do sistema de comando da velocidade linear do barco autónomo que se pretende programar no microcontrolador.

3.3 Comando da velocidade linear do barco autónomo

Para o desenvolvimento do sistema de comando da velocidade linear do barco autónomo é escolhida a velocidade linear pretendida. O comando de velocidade linear do barco calcula o valor PWM comum a aplicar nos dois motores, μ_{com} . O valor de μ_{com} é calculado através da multiplicação da velocidade linear pretendida por uma constante. De maneira a que a posição do barco acompanhe a referência, o vetor velocidade deve estar direcionado para o próximo local. Para tal ajusta-se a velocidade do barco pretendida ao longo da rotação segundo a equação (3.1), com o objetivo de corrigir o desvio numa rotação de 180° , sendo que nos primeiros 90° o barco desvia-se para um lado e nos últimos 90° o barco desvia-se para o lado contrário. Sendo v_{ref} a velocidade linear do barco pretendida, v_{max} a velocidade linear do barco pretendida máxima e $\Delta\theta$ o erro da direção em graus.

$$v_{ref} = v_{max} \left(1 - \frac{\Delta\theta}{90} \right) \quad (3.1)$$

Com o controlo de velocidade do barco autónomo descrito, através da multiplicação de uma constante pela velocidade de referência para se obter o sinal PWM comum a aplicar nos dois motores e variando a velocidade de referência segundo a equação (3.1), avança-se, de seguida, para a descrição das funcionalidades pretendidas no barco autónomo.

3.4 Funcionalidades do barco autónomo

As funcionalidades do barco autónomo são de comutação entre o modo de controlo manual e autónomo, transmissão das coordenadas do trajeto pretendido e transmissão da velocidade de rotação do motor atual para cada motor no controlo manual. Também, pretende-se que a comunicação do barco autónomo tenha um alcance de cerca de 1 quilómetro.

O controlo manual do barco funciona através da leitura dos periféricos de entrada do telecomando, cujo microcontrolador converte os sinais de entrada do telecomando escolhidos pelo utilizador numa mensagem para ser transmitida pelo rádio do telecomando. Nessa mensagem é transmitido o valor equivalente da velocidade de rotação de cada motor, juntamente com a indicação de qual o comando a executar. O barco recebe essa mensagem através do rádio, e o microcontrolador do barco faz o processo inverso, convertendo a mensagem no comando equivalente de controlo manual dos motores. Na Figura 3.3 apresenta-se a lógica do algoritmo descrito anteriormente.

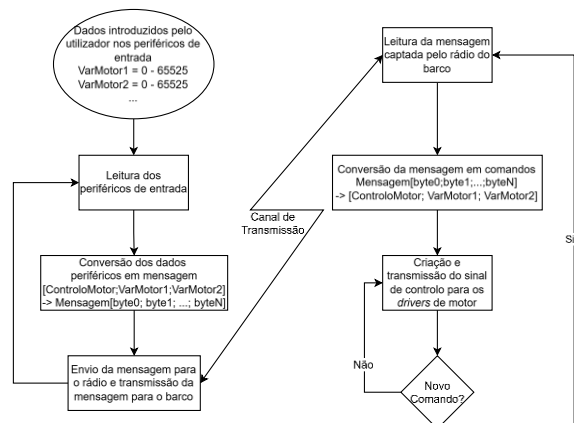


Figura 3.3 – Fluxograma do código executado pelos dois microcontroladores no controlo manual do barco autónomo.

Capítulo 3 – Especificações do barco autónomo

O controlo autónomo do barco funciona, primeiramente, com a leitura das coordenadas dos locais introduzidos pelo utilizador. O microcontrolador do barco recebe as coordenadas do destino pretendido e recolhe as coordenadas da posição atual através do sensor de posição. Com base nesses dados, calcula a distância e a direção desejada para o barco. Adicionalmente, o microcontrolador do barco obtém a direção atual, através do sensor de direção, e calcula o erro, que será utilizado como entrada no sistema de controlo de direção. Após a atuação é calculada a condição de paragem para verificar se o barco já chegou ao destino através da comparação da distância com uma margem de 10 metros. Se a distância for inferior à margem o programa avança para o próximo local pretendido. No final volta-se a repetir o algoritmo amostrando novamente o local atual, como mostra o fluxograma da Figura 3.4.

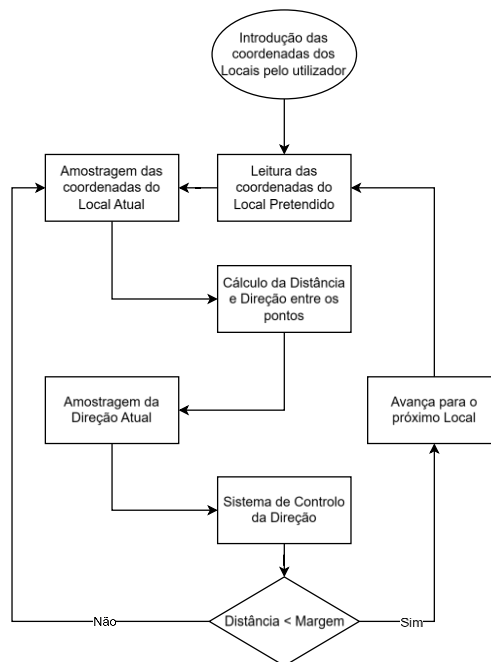


Figura 3.4 – Fluxograma do código executado pelo microcontrolador do barco na navegação autónoma.

Para poder trocar entre o modo manual e autónomo, transmite-se para o barco um comando de inicialização do modo autónomo quando for feita a troca de manual para autónomo e para trocar de autónomo para manual utiliza-se o comando de transmissão dos valores de velocidade de cada motor, para realizar o controlo manual do barco.

No telecomando o programa entra num ciclo realizando a leitura dos periféricos de entrada e verificando qual o modo de funcionamento. Se for o

Capítulo 3 – Especificações do barco autónomo

modo manual realiza a leitura dos periféricos de velocidade e constrói a mensagem de controlo manual, se for o modo autónomo constrói a mensagem de controlo autónomo, terminando o ciclo com a transmissão da mensagem para o barco através do rádio.

No barco o programa entra num ciclo realizando a leitura da mensagem captada pelo rádio e verificando qual o comando recebido. Se for recebido o comando de controlo manual o programa faz a conversão da mensagem para o sinal de acionamento dos motores, se for recebido o comando de controlo autónomo o programa executa o cálculo do sinal de acionamento dos motores a partir do sistema de controlo. O programa termina o ciclo com o envio do sinal de acionamento para os *drivers* de motor, como mostra o fluxograma apresentado na Figura 3.5.

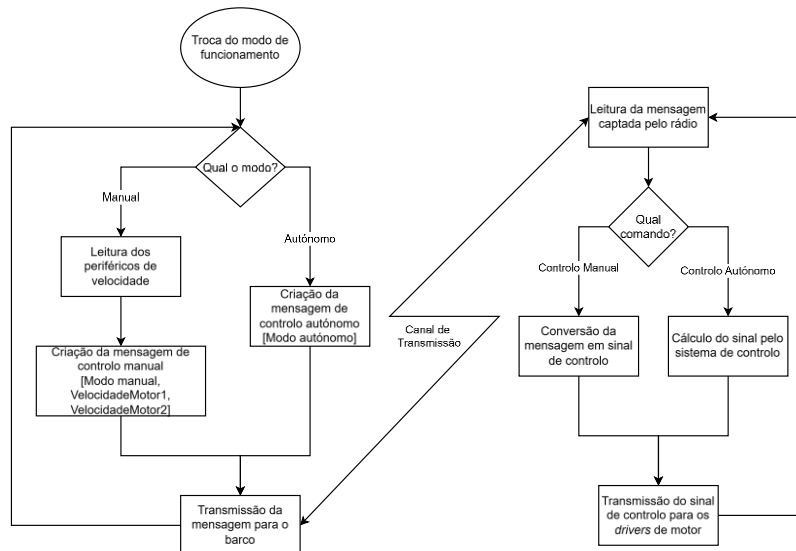


Figura 3.5 – Fluxograma do código executado pelos dois microcontroladores no modo manual e autónomo do barco.

Para transmitir as coordenadas do trajeto pretendido, o utilizador introduz as coordenadas no formato *float* e separam-se os *bytes* desses números para introduzir na estrutura da mensagem, que é um *array* de *bytes*. O programa do telecomando converte essas coordenadas numa mensagem e transmite a mensagem para o barco através do rádio. O programa do barco realiza a leitura da mensagem captada pelo rádio e faz a operação inversa, convertendo a mensagem em coordenadas do trajeto, finalizando o programa com a atualização do trajeto no barco, como mostra o fluxograma apresentado na Figura 3.6.

Capítulo 3 – Especificações do barco autónomo

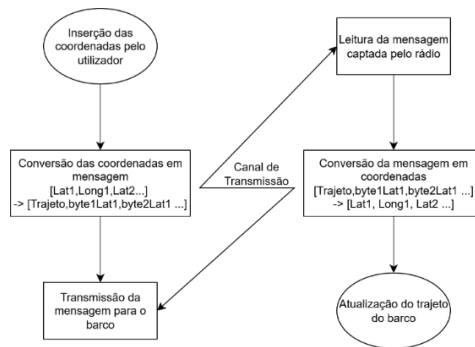


Figura 3.6 – Fluxograma do código executado no envio de coordenadas de novo trajeto.

3.5 Considerações finais sobre as especificações do barco

Em resumo, para desenvolver o sistema do barco autónomo especificaram-se quais os tipos de materiais necessários para a construção de uma solução válida. Utilizando para essa construção de solução válida um barco, dois motores, uma bateria, dois microcontroladores, sensores de posição e direção, dois rádios e *drivers* de motores.

Definiram-se os tipos de sistemas de controlo que se pretende implementar, realizando o controlo de direção do barco através do controlador da velocidade de rotação do barco, que recebe uma velocidade de rotação de referência e amostra a velocidade de rotação atual para, através do erro, calcular a variável intermédia diferencial μ_{dif} . Essa variável atua na diferença de PWM entre motores, e através do controlador da direção, que recebe uma direção de referência e amostra a direção atual para, através do erro, calcular a velocidade de rotação de referência. O comando da velocidade linear do barco realiza a multiplicação de uma constante com a referência para se obter a variável intermédia comum, que atua no PWM comum dos motores, equação (3.1).

Especificaram-se as funcionalidades do barco: de comutação entre o modo de controlo manual e autónomo, de transmissão das coordenadas do trajeto pretendido e de velocidade atual de cada motor no controlo manual. Utilizando o telecomando para comunicar a funcionalidade pretendida e escolhida pelo utilizador com o barco. O barco recebe as mensagens correspondentes aos comandos e executa a secção de código para realizar tal funcionalidade. Tendo as especificações definidas, avança-se com a dedução do modelo físico do barco, projetando os controladores através desse modelo.

Capítulo 4

DEDUÇÃO DO MODELO FÍSICO E PROJETO DOS CONTROLADORES DO BARCO AUTÓNOMO

Neste capítulo realiza-se a dedução do modelo físico representativo do movimento do barco em função da tensão elétrica aplicada nos motores. É utilizado um modelo simplificado de maneira a projetar os controladores do barco autónomo. Os modelos que caracterizam o comportamento das variáveis de estado do barco autónomo e os controladores são utilizados para a simulação e implementação prática do barco autónomo.

4.1 Dedução do modelo do barco autónomo

Para poder realizar a simulação do comportamento do barco autónomo é preciso ter um modelo representativo do barco. Começa-se a dedução desse modelo através da descrição do circuito elétrico implementado no barco. O microcontrolador do barco comunica com o rádio para obter os comandos a realizar, com os sensores de direção e posição para obter o *feedback* das variáveis de controlo, e o *driver* de motor converte os sinais de controlo acionamento dos motores. São utilizados motores DC com o circuito de acionamento de uma ponte-H.

O modelo do barco autónomo é constituído por quatro submodelos: o modelo do motor DC, para relacionar a tensão dos motores com o binário; o modelo de Yoerger, para interligar o binário gerado com a força equivalente aplicada por cada motor; um modelo equivalente das forças no barco, para transformar as forças de cada motor numa força perpendicular e numa força alinhada com a direção do barco; e um modelo hidrodinâmico do barco, para interligar a força perpendicular gerada com a velocidade de rotação do barco e a força alinhada gerada com a velocidade linear do barco.

O circuito de acionamento deste trabalho é apresentado na Figura 4.1. Utiliza-se um microcontrolador para acionar os transístores da ponte-H de cada motor. Para ligar a tensão da bateria ao motor 1, liga-se o par de transístores *T11* e *T14*, para ligar a tensão da bateria no sentido contrário ao motor 1, liga-se o par de transístores *T12* e *T13*. A mesma lógica aplica-se para o motor 2 ligando o par de transístores *T21* e *T24* para ligar diretamente e ligando o par de transístores *T22* e *T23* para ligar inversamente. Para não aplicar tensão nos motores desligam-se os transístores referentes a cada motor. Quanto mais tempo a bateria estiver ligada ao motor, maior a corrente da armadura, por causa do acumular de campo magnético na bobina.

Tendo em conta isso, atua-se na corrente da armadura de cada motor, variando a quantidade de tempo que o par de transístores fica ligado por ciclo. Com isso pode-se assumir que a tensão aplicada em cada motor é a média da tensão ao longo do tempo [30], ou seja, a tensão da bateria a multiplicar pelo *duty-cycle*, assumindo que só um par de transístores transita na rotação para a

frente ou rotação para trás. Através disso analisa-se o modelo do barco como a entrada de uma tensão média em cada motor e como saída a velocidade linear e velocidade de rotação do barco.

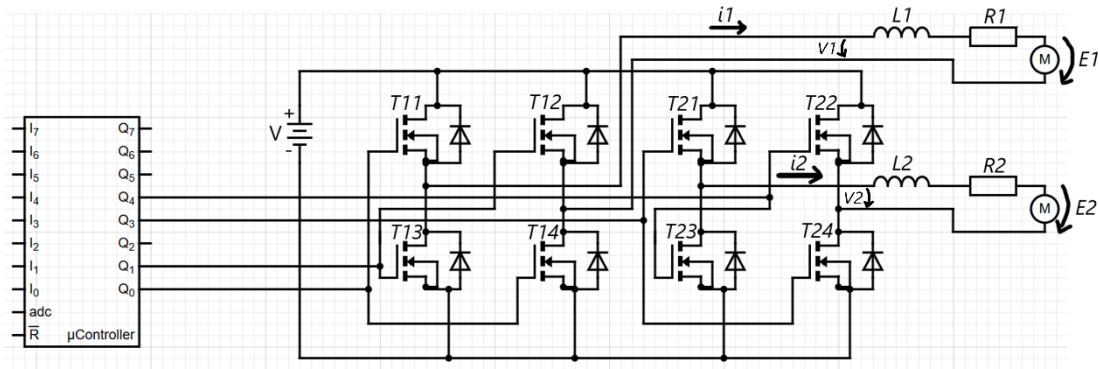


Figura 4.1 – Circuito de acionamento do barco autónomo.

O modelo do motor DC é constituído por um circuito equivalente que contém uma bobina, uma resistência e uma fonte de tensão causada pelo efeito de indução mútua, como é apresentado na Figura 2.20 no estator [30].

O somatório das tensões numa malha fechada é igual a zero, ou seja, a queda de tensão na bobina, que é o valor da sua indutância, L_j , a multiplicar pela derivada da corrente, $\frac{di_j}{dt}$, é igual à tensão de entrada no motor proveniente da ponte-H, V_j , a subtrair à queda de tensão na resistência de perdas do enrolamento, que é o valor da resistência, R_j , a multiplicar pela corrente, i_j , e a subtrair o efeito de indução mútua, que é igual à velocidade de rotação do motor, Ω_j , a multiplicar por uma constante, k_b , como mostra a equação (4.1) [30].

$$L_j \frac{di_j}{dt} = V_j - k_b \Omega_j - R_j i_j \quad (4.1)$$

O binário gerado, τ_{g_j} , equivale a uma constante, k_t , a multiplicar pela corrente, i_j , como mostra a equação (4.2) [30]. Sendo j o indicador do motor $\{1,2\}$.

$$\tau_{g_j} = k_t i_j \quad (4.2)$$

Aplicando a transformada de Laplace, às equações (4.1) e (4.2) e reorganizando os termos das equações em função das entradas e saídas, obtém-se a equação (4.3) do binário motor.

$$\tau_{g_j}(s) = k_t * \frac{V_j(s) - k_b \Omega_j(s)}{s + \frac{R_j}{L_j}} * \frac{1}{L_j} \quad (4.3)$$

O modelo de Yoerger, que descreve um motor a rodar dentro de água, indica que a variação da velocidade de rotação do motor ao longo do tempo é proporcional ao binário gerado e diminui por um fator quadrático da velocidade de rotação do motor, como mostra a equação (2.3) [27]. O equivalente de força aplicada pelo motor é proporcional ao quadrado da velocidade de rotação do motor [27].

Tendo isto em conta, obtém-se a equação (4.4), que descreve um motor a rodar dentro de água, sendo essa equação a transformada de Laplace da equação (2.3), e obtém-se a equação (4.5), que descreve a força equivalente do motor. Sendo j o indicador do motor $\{1,2\}$, T_j é a força equivalente gerada pelo motor, τ_c o binário de carga, C_t a constante que converte a velocidade de rotação em força, β e α as constantes que transformam o binário motor em rotação [27].

$$\Omega_j(s) = \frac{\beta(\tau_{g_j}(s) - \tau_c)}{s + \alpha|\Omega_j(s)|} \quad (4.4)$$

$$T_j = C_t \Omega_j |\Omega_j| \quad (4.5)$$

As constantes α , β e C_t são constantes dependentes dos parâmetros da turbina [27], que precisam de ser determinados.

Obtendo a força em cada motor, faz-se uma decomposição para obter o equivalente das forças no centro de massa do barco. A força \vec{T}_1 , gerada pelo motor esquerdo, e a força \vec{T}_2 , gerada pelo motor direito, afetam o barco na sua aceleração e velocidade de rotação.

Como os motores são simétricos, na perpendicular da direção do barco, as forças ao serem ambas aplicadas para a frente, subtraem-se no equivalente da força diferencial. Como os motores estão direcionados para a frente do barco, as forças ao serem ambas aplicadas para a frente, somam-se no equivalente da força comum. Parte de cada força é decomposta numa força perpendicular e uma força alinhada à direção do barco, por isso multiplicam-se as equações por constantes, C_1 e C_2 , que fazem esse equivalente. Sendo que essas constantes

têm de respeitar o teorema de Pitágoras, ou seja, a soma do quadrado de cada constante tem de ser igual a 1, para igualar o somatório das forças.

Na Figura 4.2 é apresentado um diagrama dos equivalentes das forças do barco. Obtém-se, também, as equações (4.6) e (4.7) sobre as forças equivalentes geradas, diferencial e comum, tendo em conta a descrição anterior. Sendo a força na direção do barco denominada de força comum \vec{T}_{com} , e a força perpendicular a essa, no plano de superfície, denominada de força diferencial \vec{T}_{dif} , sendo Cm o centro de massa, \vec{T}_{1com} a força comum aplicada pelo motor esquerdo, \vec{T}_{2com} a força comum aplicada pelo motor direito, \vec{T}_{1dif} a força diferencial aplicada pelo motor esquerdo, \vec{T}_{2dif} a força diferencial aplicada pelo motor direito e C_1 e C_2 as constantes que equivalem às forças, dependentes da estrutura do barco.

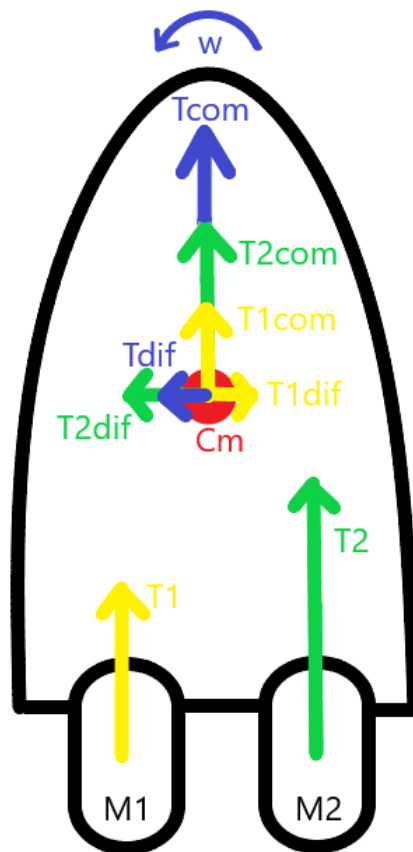


Figura 4.2 – Diagrama com o equivalente das forças do barco.

$$\vec{T}_{dif} = C_1(\vec{T}_1 - \vec{T}_2) \quad (4.6)$$

$$\vec{T}_{com} = C_2(\vec{T}_1 + \vec{T}_2) \quad (4.7)$$

Capítulo 4 – Dedução do modelo físico e projeto dos controladores do barco

O modelo hidrodinâmico do barco é obtido através das equações da Física [25] (4.8), que indica que a força aplicada num objeto, \vec{F} , é igual à massa, m , a multiplicar pela aceleração ou seja a derivada da velocidade, $\frac{dv}{dt}$, e (4.9), que indica que a força centrípeta, \vec{F}_c , é igual à massa a multiplicar pela velocidade linear do objeto, v , e a multiplicar pela velocidade de rotação do objeto, ω . A essas equações adiciona-se um termo de perdas devido ao atrito hidrodinâmico. Sendo esses valores caracterizados por um termo constante a multiplicar pela velocidade linear ao quadrado ou pela velocidade de rotação ao quadrado como mostram as equações (4.10) e (4.11) [25]. Nas equações de (4.8) a (4.11), \vec{F} é a força linear aplicada ao barco pelos motores, \vec{F}_c é a força centrípeta, m é a massa do barco, v é a velocidade linear do barco, ω a velocidade de rotação do barco, C_v o termo de perdas devido ao atrito hidrodinâmico na velocidade linear e C_ω o termo de perdas devido ao atrito hidrodinâmico na velocidade de rotação.

$$\vec{F} = m \frac{dv}{dt} \quad (4.8)$$

$$\vec{F}_c = mv\omega \quad (4.9)$$

$$\vec{T}_{com} - C_v v|v| = m \frac{dv}{dt} \quad (4.10)$$

$$\vec{T}_{dif} - C_\omega \omega|\omega| = mv\omega \quad (4.11)$$

Simplificando a última equação, obtém-se, a equação (4.12) da velocidade de rotação do barco com a dedução apresentada no Apêndice A, na secção A.1.

$$\omega(t) = \frac{-\vec{T}_{dif}(t)}{|\vec{T}_{dif}(t)|} * \frac{mv(t) - \sqrt{(mv(t))^2 + 4|\vec{T}_{dif}(t)|C_\omega}}{2C_\omega} \quad (4.12)$$

Integra-se a velocidade de rotação para obter a direção em graus θ , como mostra a equação (4.13), e com isto obtém-se o modelo do barco, composto pelos quatro submodelos apresentados anteriormente sendo eles: o modelo do motor DC, o modelo de Yoerger, o equivalente das forças e o modelo hidrodinâmico do barco. Esse modelo é apresentado em diagrama de blocos na Figura 4.3, tendo em conta as equações apresentadas anteriormente.

$$\theta(t) = \int \omega * dt \quad (4.13)$$

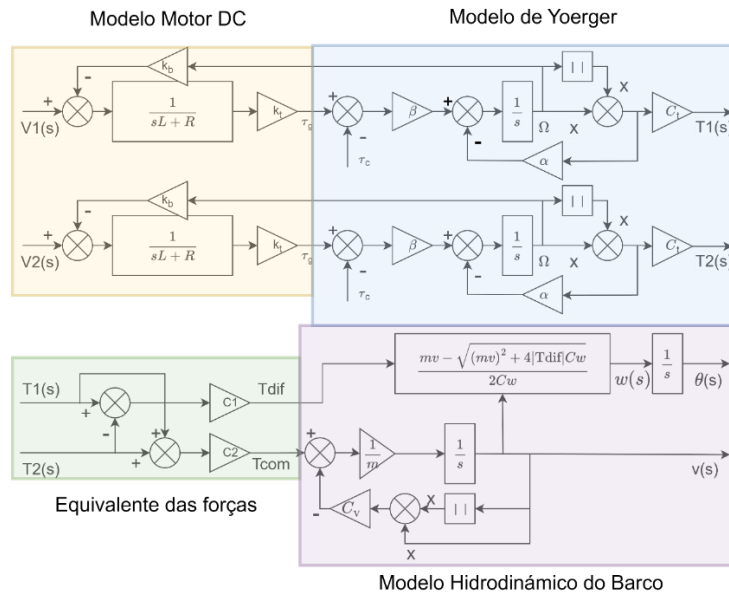


Figura 4.3 – Modelo físico do barco decomposto nos seus submodelos.

O modelo do barco é representado por um sistema não linear cujas grandezas de comando são as tensões V_1 e V_2 dos motores DC e as grandezas de saída, as variáveis que se pretendem controlar, que são a velocidade de rotação do barco ω , o ângulo de direção do barco θ e a velocidade linear do barco v . Na próxima secção serão desenhados os controladores para essas variáveis de saída, ou seja, o comando da velocidade linear, o controlador da velocidade de rotação e da direção. Com essas variáveis também será realizado o controlo de posição.

4.2 Projeto dos controladores do barco autónomo

Como se pode observar anteriormente, o modelo físico do barco deduzido é complexo e não linear. Esse facto torna difícil a projeção de controladores lineares no sistema, não se conseguindo ter uma análise prévia de como o sistema vai reagir. Por isso, utiliza-se a abordagem do trabalho [3] no projeto de controladores do barco autónomo. Essa abordagem indica que o sistema do barco autónomo pode ser simplificado por um sistema de primeira ordem num certo ponto de funcionamento, tendo como entrada a variável de atuação e saída a velocidade de rotação do barco. Esse modelo é denominado de modelo de *Nomoto* de primeira ordem.

Aplicando esse modelo a este sistema do barco autónomo, com a entrada w_{dif} , e adicionando o controlador da velocidade de rotação, obtém-se o diagrama de blocos do sistema apresentado na Figura 4.4, sendo K o ganho do modelo de *Nomoto* e T_{ds} a constante de atraso do modelo de *Nomoto*.

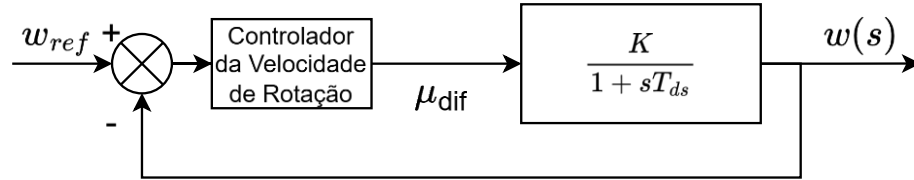


Figura 4.4 – Modelo de *Nomoto* de primeira ordem do barco autónomo com controlador de velocidade de rotação.

Para o controlador da velocidade de rotação utiliza-se um controlador proporcional-integral com o objetivo de corrigir perturbações constantes, sendo k_w a constante proporcional do controlador de velocidade de rotação e k_{wi} a constante integral do controlador de velocidade de rotação. Tendo em conta isso, apresenta-se a função de transferência do modelo equivalente da Figura 4.4, na equação (4.14).

$$\left(\omega_{ref}(s) - \omega(s)\right) \frac{k_w}{s} \left(s + \frac{k_{wi}}{k_w}\right) * \frac{K}{1+sT_{ds}} = \omega(s) \quad (4.14)$$

Reorganizando a equação (4.14), obtém-se a velocidade de rotação do barco em função da velocidade de rotação de referência do barco, como mostra a equação (4.15), com a dedução apresentada no Apêndice A, na secção A.1.

$$\omega(s) = \frac{k_w \left(s + \frac{k_{wi}}{k_w}\right) * \frac{\frac{K}{T_{ds}}}{s + \frac{1}{T_{ds}}}}{s + k_w \left(s + \frac{k_{wi}}{k_w}\right) * \frac{\frac{K}{T_{ds}}}{s + \frac{1}{T_{ds}}}} \omega_{ref}(s) \quad (4.15)$$

Para determinar os parâmetros do controlador de velocidade de rotação do barco, realiza-se o cancelamento polo-zero, igualando $\frac{k_{wi}}{k_w}$ a $\frac{1}{T_{ds}}$, o que simplifica a equação (4.15) para a equação (4.16).

$$\omega(s) = \frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} \omega_{ref}(s) \quad (4.16)$$

Para determinar k_w , projeta-se o tempo de subida. O tempo de subida de 10% a 90% do valor do estado estacionário é dado pela divisão da constante 2,2 com o polo dominante do sistema, ou seja, para projetar o tempo de subida utiliza-se a equação (4.17), sendo t_r o tempo de subida [32]:

$$t_r = \frac{2,2}{k_w * \frac{K}{T_{ds}}} \quad (4.17)$$

Com isto obtém-se as duas equações (4.18) (4.19), que determinam os parâmetros k_w e k_{wi} em função do tempo de subida e as restantes constantes que caracterizam o sistema.

$$k_w = \frac{2,2}{t_r * \frac{K}{T_{ds}}} \quad (4.18)$$

$$k_{wi} = \frac{k_w}{T_{ds}} \quad (4.19)$$

Com os parâmetros do controlador da velocidade de rotação determinados, simplifica-se o modelo e adiciona-se o controlador de direção do barco, obtendo o diagrama de blocos do sistema apresentado na Figura 4.5.

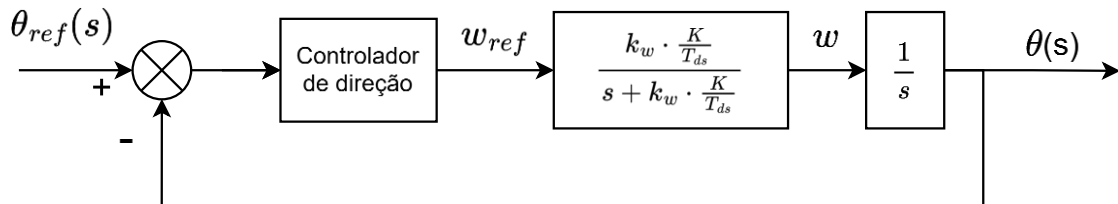


Figura 4.5 – Modelo simplificado do barco autónomo com o controlador de direção.

Para o controlador de direção do barco utiliza-se um controlador proporcional-derivativo com o objetivo de simplificar o controlo para um de sistema de primeira ordem, sendo a constante proporcional do controlador k_p e a constante derivativa do controlador k_d . Tendo em conta isso, apresenta-se a função de transferência do modelo equivalente da Figura 4.5, na equação (4.20).

$$\theta(s) = \frac{1}{s} * \frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} (\theta_{ref}(s) - \theta(s)) * k_d \left(s + \frac{k_p}{k_d} \right) \quad (4.20)$$

Reorganizando a equação (4.20), obtém-se o ângulo de direção do barco em função da sua referência, como mostra a equação (4.21), com dedução apresentada no Apêndice A, na secção A.1.

$$\theta(s) = \frac{k_d \left(s + \frac{k_p}{k_d} \right) \frac{k_w * T_{ds} \frac{K}{s + k_w * T_{ds}}}{s + k_d \left(s + \frac{k_p}{k_d} \right) \frac{k_w * T_{ds} \frac{K}{s + k_w * T_{ds}}} \theta_{ref}(s) \quad (4.21)$$

Para determinar os parâmetros do controlador da direção do barco, realiza-se o cancelamento polo-zero, igualando $\frac{k_p}{k_d}$ a $k_w * \frac{K}{T_{ds}}$, o que simplifica a equação (4.21) para a equação (4.22).

$$\theta(s) = \frac{k_d * k_w * T_{ds} \frac{K}{s + k_d * k_w * T_{ds}} \theta_{ref}(s) \quad (4.22)$$

Para determinar k_d , projeta-se o tempo de subida do sistema como mostra a equação (4.23), sendo t_r o tempo de subida.

$$t_r = \frac{2,2}{k_d * k_w * T_{ds} \frac{K}{s + k_d * k_w * T_{ds}}} \quad (4.23)$$

Com isto obtém-se as duas equações (4.24) (4.25), que determinam os parâmetros k_p e k_d em função do tempo de subida e restantes constantes que caracterizam o modelo.

$$k_d = \frac{2,2}{t_r * k_w * T_{ds} \frac{K}{s + k_d * k_w * T_{ds}}} \quad (4.24)$$

$$k_p = k_d * k_w * \frac{K}{T_{ds}} \quad (4.25)$$

Para estimar a constante k_v , que multiplica a velocidade de referência e obter a variável intermédia comum μ_{com} , relaciona-se a velocidade no ponto de operação médio $\sim v$ com o μ_{com} aplicado, esses valores foram determinados empiricamente e são apresentados no cálculo de k_v , como mostra a equação (4.26).

$$k_v = \frac{\sim \mu_{com}}{\sim v} = \frac{0,7}{0,58} \cong 1,2 \quad (4.26)$$

De maneira a fazer com que o barco siga a trajetória pretendida, é realizada a leitura da posição pretendida e amostrada a posição atual. Com isso é realizado o cálculo da direção entre coordenadas. A posição é o integral da velocidade tendo em conta a direção, aproximando à deslocação num plano XY, que depois se converte essas posições para coordenadas esféricas. É de notar que a integração da velocidade linear tem como valor inicial as coordenadas esféricas do início do trajeto e que a integração da velocidade de rotação do barco tem como valor inicial a direção do barco no início. Com isto obtém-se a Figura 4.6 representativa de todos os controladores e comandos do sistema do barco autónomo com as suas relações.

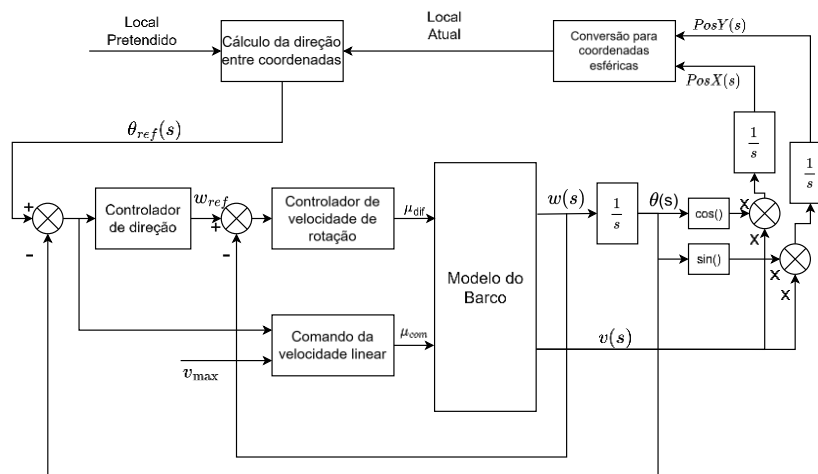


Figura 4.6 – Sistema de controlo de posição do barco autónomo.

Para projetar os valores das constantes de todos os controladores do barco autónomo, realizaram-se testes para determinação dos parâmetros, apresentados no Apêndice B. Considerou-se o valor de T_{ds} igual a 0,12 segundos (60% de 0,2 segundos), ao analisar o teste de reação nos motores apresentado no Apêndice B, na secção B.1 (Figura B.2). Considerou-se o valor de K igual a 0,9 (em radianos), devido ao teste da hidrodinâmica, que para um $\mu_{dif} = 1$ causa uma velocidade de rotação de 0,9 rad/s, quando é aplicado um degrau na tensão de entrada do motor direito apresentado no Apêndice B, na secção B.2 (Figura B.6 (a)).

Impôs-se um tempo de reação de 0,6 segundos para o controlador da velocidade de rotação do barco e um tempo de reação de 2 segundos para o controlador da direção do barco. Tendo em conta essas condições, utilizaram-se as equações (4.18) (4.19) (4.24) (4.25) para determinar os parâmetros dos

Capítulo 4 – Dedução do modelo físico e projeto dos controladores do barco

controladores do barco autónomo, como mostra o Apêndice A, na secção A.2, e apresentam-se os valores dos parâmetros obtidos na Tabela 4.1.

Tabela 4.1 – Constantes dos controladores do barco autónomo.

Descrição da constante	Nomenclatura da constante	Valor da constante
Constante proporcional do controlador de velocidade de rotação	k_w	0,5 (em radianos)
Constante integral do controlador de velocidade de rotação	k_{wi}	4 (em radianos)
Constante proporcional do controlador de direção	k_p	0,02 (em graus)
Constante derivativa do controlador de direção	k_d	0,005 (em graus)
Constante proporcional do comando de velocidade linear	k_v	1,2 (em segundos por metro)

4.3 Considerações finais sobre o modelo do barco e projeto de controladores

Em resumo, para realizar a dedução do modelo físico do barco autónomo, realizou-se a junção de 4 submodelos: o modelo do motor DC; o modelo de Yoerger; o equivalente das forças; e o modelo hidrodinâmico do barco. Utilizou-se uma simplificação do modelo de maneira a poder aplicar o método de determinação de parâmetros do controlador de cancelamento polo-zero e projetar o tempo de subida através de equações conhecidas dos sistemas lineares. Com isso, projetou-se o controlador da velocidade de rotação do barco, que permitiu projetar o controlador da direção do barco, avançando com o projeto do comando da velocidade linear do barco e demonstrando como é realizado o controlo de posição do barco. Tendo os controladores projetados, procede-se com a simulação do sistema, através da reprodução dos diagramas de blocos apresentados anteriormente, no *software* de simulação SIMULINK/MATLAB.

Capítulo 5

SIMULAÇÃO DO SISTEMA DE NAVEGAÇÃO DO BARCO AUTÓNOMO

Neste capítulo é realizada a simulação do sistema de navegação do barco. Começa-se pela criação do diagrama de blocos equivalente do sistema de navegação do barco, tendo como base os modelos e controladores do barco autónomo estudados no capítulo 4. Seguidamente, realiza-se a simulação do sistema, recolhendo os dados relevantes do barco autónomo: a direção, a posição, a velocidade linear, e a velocidade de rotação. Nos testes de simulação são analisados como é que os controladores projetados reagem aos valores de referência e as perturbações. Numa primeira parte é realizada a simulação do barco tendo em conta as características apresentadas no Apêndice B e numa segunda parte são realizadas simulações com a variação dos parâmetros de entrada: da eficiência em cada motor, da direção pretendida, da velocidade linear de referência, e da adição de ruído no sistema.

5.1 Implementação do diagrama de blocos no SIMULINK/MATLAB

Para simular o sistema de navegação do barco autónomo, juntaram-se os blocos dos controladores da velocidade de rotação, direção e velocidade linear do barco, ao modelo que caracteriza o comportamento do barco, para atuar na tensão média aplicada em cada motor, multiplicando a tensão da bateria com o *duty-cycle* do par de transístores que transita, tendo em conta a direção de rotação dos motores pretendida, sendo que o outro par fica desligado.

No SIMULINK/MATLAB, para implementar uma soma ou subtração utiliza-se o bloco *Sum*, para implementar uma multiplicação entre duas variáveis utiliza-se o bloco *Product* e para implementar um ganho ou multiplicação por uma constante utiliza-se o bloco *Gain*. O bloco *Abs* retorna o valor absoluto da entrada o que permite dar *feedback* quadrático tendo em conta a direção.

Para implementar o sistema de controlo de direção do barco no SIMULINK/MATLAB, amostra-se o ângulo atual, com um atraso de amostragem de 0,3 segundos, representado pelo bloco *Zero-Order Hold*, sendo esse atraso o tempo de processamento do microcontrolador para realizar um ciclo de controlo da direção. Subtrai-se o ângulo atual ao ângulo de referência para obter o erro de direção. Esse erro é multiplicado pelo k_p e subtraído pela multiplicação da velocidade de rotação atual com o k_d , implementado o controlador proporcional-derivativo, tendo em conta que a derivada da direção pretendida é aproximadamente zero em comparação com a derivada da direção atual.

Obtém-se, então, a velocidade de rotação de referência do barco e limita-se essa velocidade de rotação entre -0,8 rad/s e 0,8 rad/s através do bloco *Saturation*, de maneira a operar dentro dos limites físicos do barco tendo em conta que a velocidade de rotação máxima do barco é 0,9 rad/s, como apresentado no Apêndice B, na secção B.1. Subtrai-se a velocidade de rotação atual à velocidade de rotação de referência para se obter o erro de velocidade de rotação, que serve de entrada para o controlador proporcional-integral da velocidade de rotação.

Esse controlador é implementado com o bloco *Discrete PI Controller*, que utiliza o k_w e k_{wi} como constante proporcional e constante integral, respetivamente. Esse controlador adiciona um atraso de amostragem de 6

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

milissegundos, sendo esse atraso o tempo para o microcontrolador realizar um ciclo de controlo da velocidade de rotação, e limita a saída μ_{dif} entre -1 a 1, para limitar o *duty-cycle* máximo a 1. O diagrama de blocos do controlo da direcção é apresentado na Figura 5.1, com a cor verde.

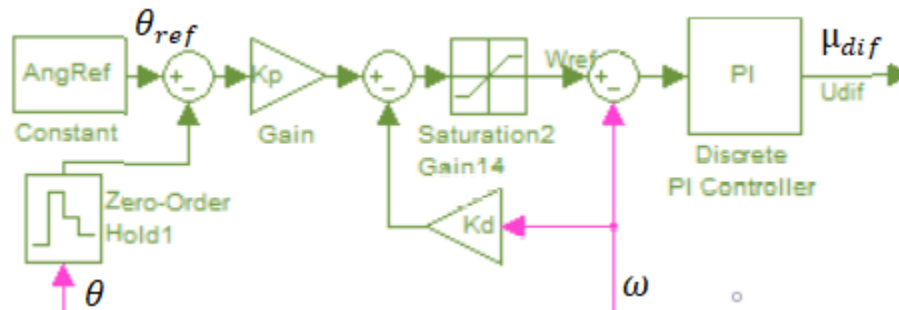


Figura 5.1 – Diagrama de blocos do controlo de direcção.

Para implementar o comando da velocidade linear do barco no SIMULINK/MATLAB, utiliza-se o erro do ângulo para ajustar a velocidade linear do barco autónomo, como mostra a equação (3.1). Esse valor de velocidade linear de referência é multiplicado pelo ganho k_v para calcular a saída μ_{com} . O bloco *Zero-Order Hold* impõe um atraso de atuação de 0,3 segundos, devido ao processamento, e o bloco *Saturation* limita o valor da saída μ_{com} entre -0,8 e 0,8, um pouco inferior ao *duty-cycle* máximo de 1. O diagrama de blocos de ajuste da velocidade linear do barco é apresentado na Figura 5.2, com a cor laranja.

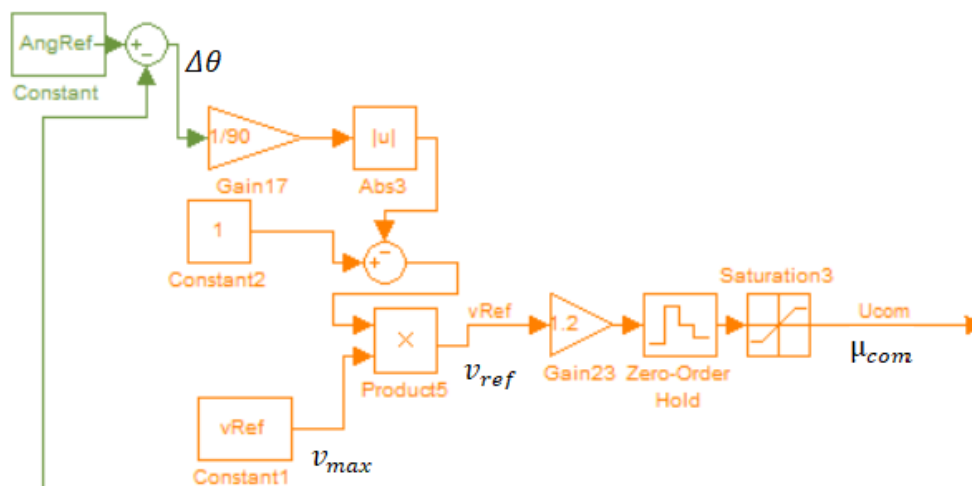


Figura 5.2 – Diagrama de blocos do comando da velocidade linear.

Para implementar o sistema de acionamento do motor no SIMULINK/MATLAB, soma-se o μ_{dif} com o μ_{com} para obter o *duty-cycle* do motor direito e subtrai-se o μ_{dif} ao μ_{com} para obter o *duty-cycle* do motor esquerdo.

Caso esse valor dê negativo significa que o *duty-cycle* é referente ao par de transístores que rodam o motor para trás, obtendo um *duty-cycle* igual a -1 vezes esse valor. Utiliza-se o bloco *Saturation* para limitar o valor da soma e subtração de μ_{dif} com o μ_{com} entre -1 a 1. Após esse bloco utiliza-se um bloco *Switch* com um bloco *Abs* para verificar se o valor absoluto do *duty-cycle* é acima de 0,5, que é o valor mínimo para o qual o motor começa a rodar, pois nos testes práticos o motor não acionava para valores inferiores. Multiplica-se a tensão da bateria com o valor de *duty-cycle* de cada motor para aproximar o circuito de acionamento da ponte-H, tendo em conta a direção. Obtém-se assim uma tensão equivalente de entrada para o modelo de cada motor, como mostra o diagrama de blocos do acionamento apresentado na Figura 5.3, com a cor vermelha.

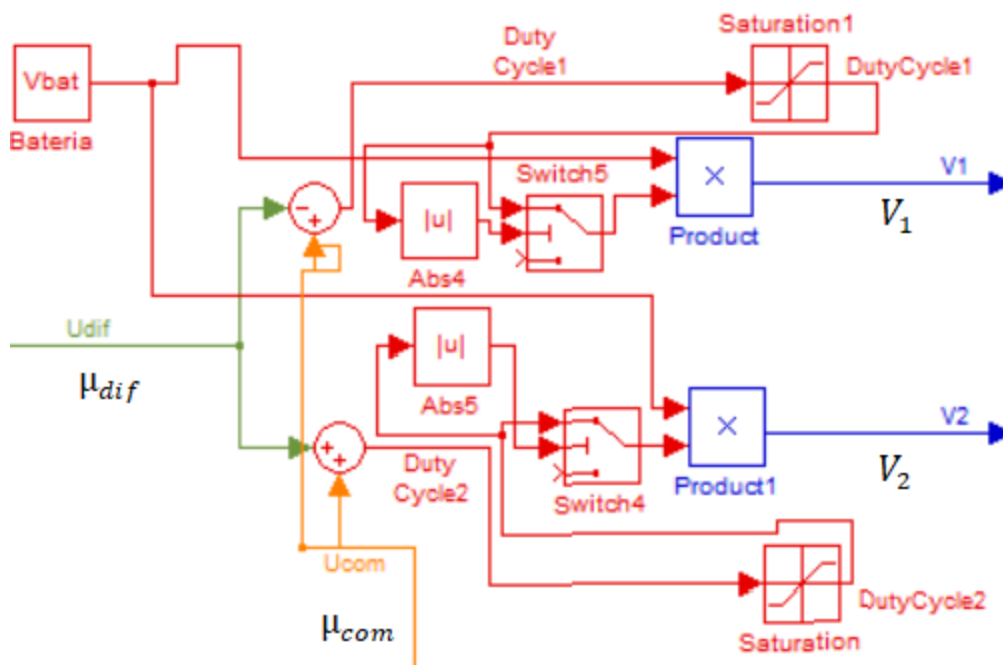


Figura 5.3 – Diagrama de blocos do acionamento dos motores.

Para implementar o modelo do motor DC, no SIMULINK/MATLAB, subtrai-se à tensão de entrada de cada motor, V_j , a sua força eletromotriz induzida, que é o valor da velocidade de rotação desse motor, Ω_j , multiplicado pela constante k_b . Esse valor de tensão passa por um sistema de primeira ordem, proveniente do modelo do motor DC, modelado através de uma bobina, L_j , e uma resistência, R_j [30]. Ao passar por esse sistema de primeira ordem, obtém-se a corrente que passa no motor, i_j , que ao multiplicar pelo ganho k_t resulta no binário motor gerado de entrada para cada motor, τ_{g_j} , sendo j o

indicador do motor {1,2}. O diagrama de blocos do modelo do motor DC é apresentado na Figura 5.4, com a cor azul escuro.

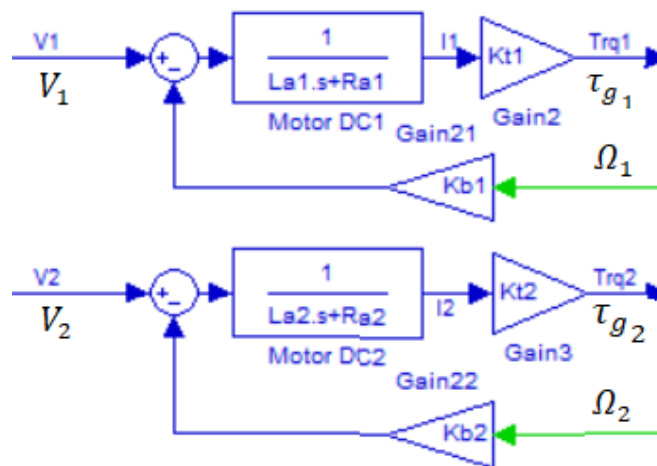


Figura 5.4 – Diagrama de blocos do modelo do motor DC.

Para implementar o modelo de Yoerger no SIMULINK/MATLAB, multiplica-se ao binário motor gerado de cada motor, τ_{g_j} , com o ganho β e subtrai-se a esse valor a multiplicação do quadrado da velocidade de rotação de cada motor com a constante de atrito α [27]. Ao integrar esse valor, com o bloco *Integrator*, obtém-se a velocidade de rotação de cada motor, que depois se multiplica pelo seu valor absoluto, com o bloco *Abs*, para obter o quadrado da velocidade de rotação de cada motor tendo em conta a respetiva direção.

O quadrado da velocidade de rotação multiplica pelo ganho C_t para se obter o valor da força equivalente gerada pelo motor, T_j . Tendo em conta a direção com que o motor puxa a água, para frente ou para trás, existe uma alteração na eficiência da conversão de velocidade de rotação do motor ao quadrado para a força aplicada ao barco. Para refletir essa diferença de eficiência foi adicionado mais um termo em cada motor para distinguir o modelo do barco caso esteja a andar para a frente ou para a trás, através da introdução de quatro constantes C_t com valores diferentes, sendo essas constantes C_{t11} , C_{t12} , C_{t21} e C_{t22} calculadas no Apêndice B, na secção B.3. Através do bloco *Switch* o simulador utiliza a constante apropriada correspondente se o motor está a andar para a frente ou para a trás. Obtém-se, assim, a força equivalente gerada em cada motor, como mostra o diagrama de blocos da Figura 5.5.

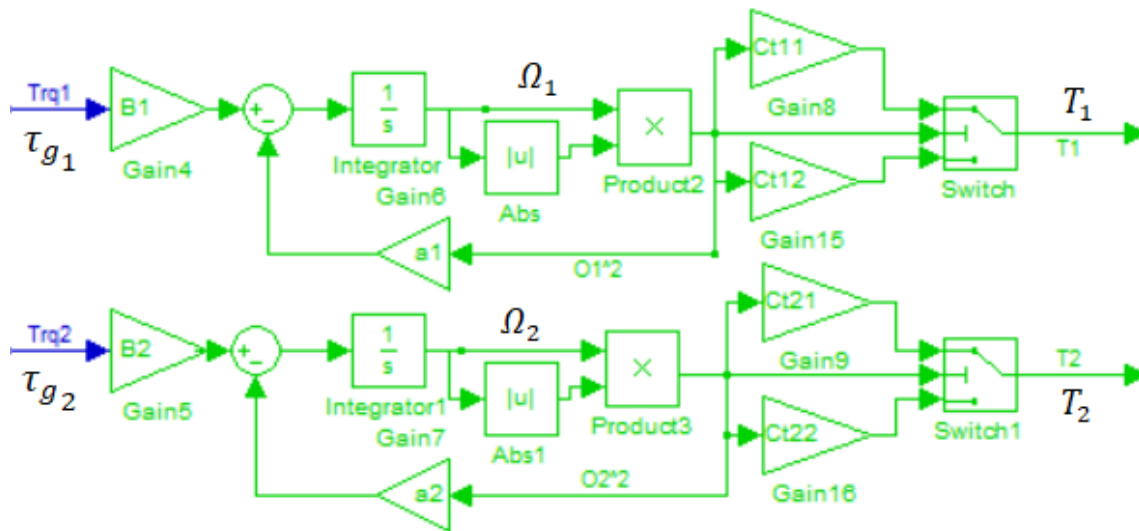


Figura 5.5 – Diagrama de blocos do modelo de Yoerger.

Para implementar o equivalente das forças, no SIMULINK/MATLAB, realizou-se a subtração da força gerada pelo motor direito, T_2 , com a força gerada pelo motor esquerdo, T_1 , multiplicando por C_1 para obter a força equivalente diferencial, T_{dif} . Para obter a força equivalente comum, T_{com} , realizou-se a soma das forças geradas pelos motores e multiplicou-se por C_2 . O diagrama de blocos do equivalente das forças é apresentado na Figura 5.6 com a cor azul claro.

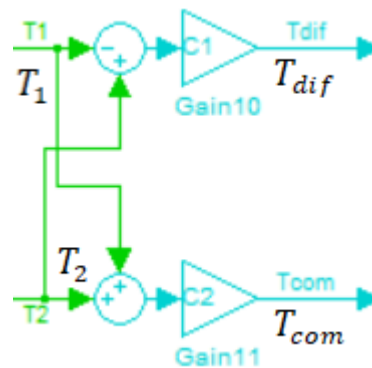


Figura 5.6 – Diagrama de blocos do equivalente das forças.

Para implementar o modelo hidrodinâmico do barco, dado pelas equações (4.10) e (4.11), no SIMULINK/MATLAB, subtrai-se ao valor da multiplicação da velocidade linear, v , quadrada com a constante de atrito C_v à força comum equivalente, T_{com} , para obter o resultante da força aplicada no movimento linear, F_t . Divide-se essa força pela massa, m , para obter a aceleração e integra-se, com o bloco *Integrator*, para obter a velocidade linear do barco. Para obter a

velocidade de rotação do barco utiliza-se um subsistema que implementa a equação (4.12), tendo em conta as entradas de força diferencial equivalente, T_{dif} , massa, m , velocidade linear, v , e constante de atrito C_{ω} . Ao integrar a velocidade de rotação do barco obtém-se o ângulo de direção do barco, como mostra o diagrama de blocos do modelo hidrodinâmico do barco, apresentado na Figura 5.7, com a cor rosa.

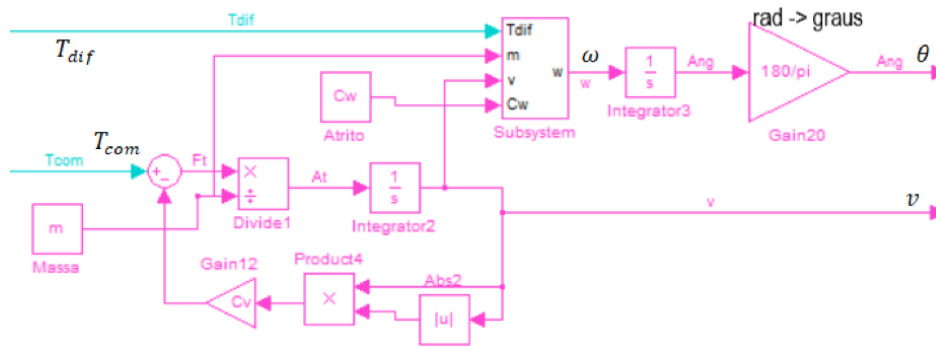


Figura 5.7 – Diagrama de blocos do modelo hidrodinâmico do barco.

Através dos dados de direção e velocidade do barco, também se pode observar a variação da posição ao longo do tempo. Aproximam-se as coordenadas esféricas para um plano de coordenadas XY, para pequenas distâncias, utilizando a técnica de *Dead Reckoning*. Ao integrar, com o bloco *Integrator*, o cosseno, com o bloco *Cosine*, do ângulo de direção, θ , multiplicado pela velocidade linear, v , estima-se a posição no eixo dos Xs e ao integrar o seno, com o bloco *Sine*, do ângulo de direção multiplicado pela velocidade linear estima-se a posição no eixo dos Ys, como mostra o diagrama de blocos apresentado na Figura 5.8. A variável T da Figura 5.8, serve somente para amostrar o tempo, integrando a constante 1 ao longo do tempo.

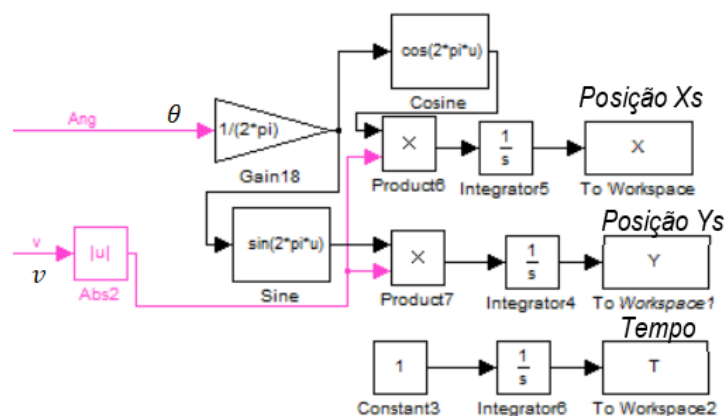


Figura 5.8 – Diagrama de blocos do cálculo da posição.

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

O diagrama de blocos completo da simulação é apresentado na secção C.5 do Apêndice C, na Figura C.19, com o respetivo código de simulação apresentado na Figura C.18.

Tendo o diagrama de blocos construído, a partir da implementação dos controladores da direção e velocidade de rotação do barco com o sistema de acionamento, foram, também, adicionados os atrasos e limites de atuação nesses modelos. Juntamente com a implementação dos modelos do motor DC, modelo de Yoerger, equivalente das forças e modelo hidrodinâmico do barco, consegue-se simular o modelo físico do barco. Procede-se, então, para a simulação do sistema e análise dos resultados.

5.2 Simulação do sistema de navegação

Para observar o comportamento do barco autónomo, pretende-se visualizar a variação do ângulo de direção do barco ao longo do tempo, a variação da velocidade de rotação do barco e da sua respetiva referência ao longo do tempo, a variação da velocidade linear do barco e da sua respetiva referência ao longo do tempo e a variação da posição ao longo do tempo.

Para simular o barco autónomo criou-se um ficheiro de constantes, no MATLAB, com a atribuição de cada constante da simulação no SIMULINK, apresentada no Apêndice C, na secção C.5 (Figura C.18). Nestes resultados de simulação o sistema foi caracterizado pelas constantes obtidas no Apêndice B, utilizando como entradas da simulação uma variação angular de 180° e uma velocidade linear máxima de referência de 0,5 m/s. Na Figura 5.9 observa-se a reação da direção (a), da velocidade de rotação de referência (b), da velocidade linear de referência (c) e da velocidade linear (d) ao longo do tempo.

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

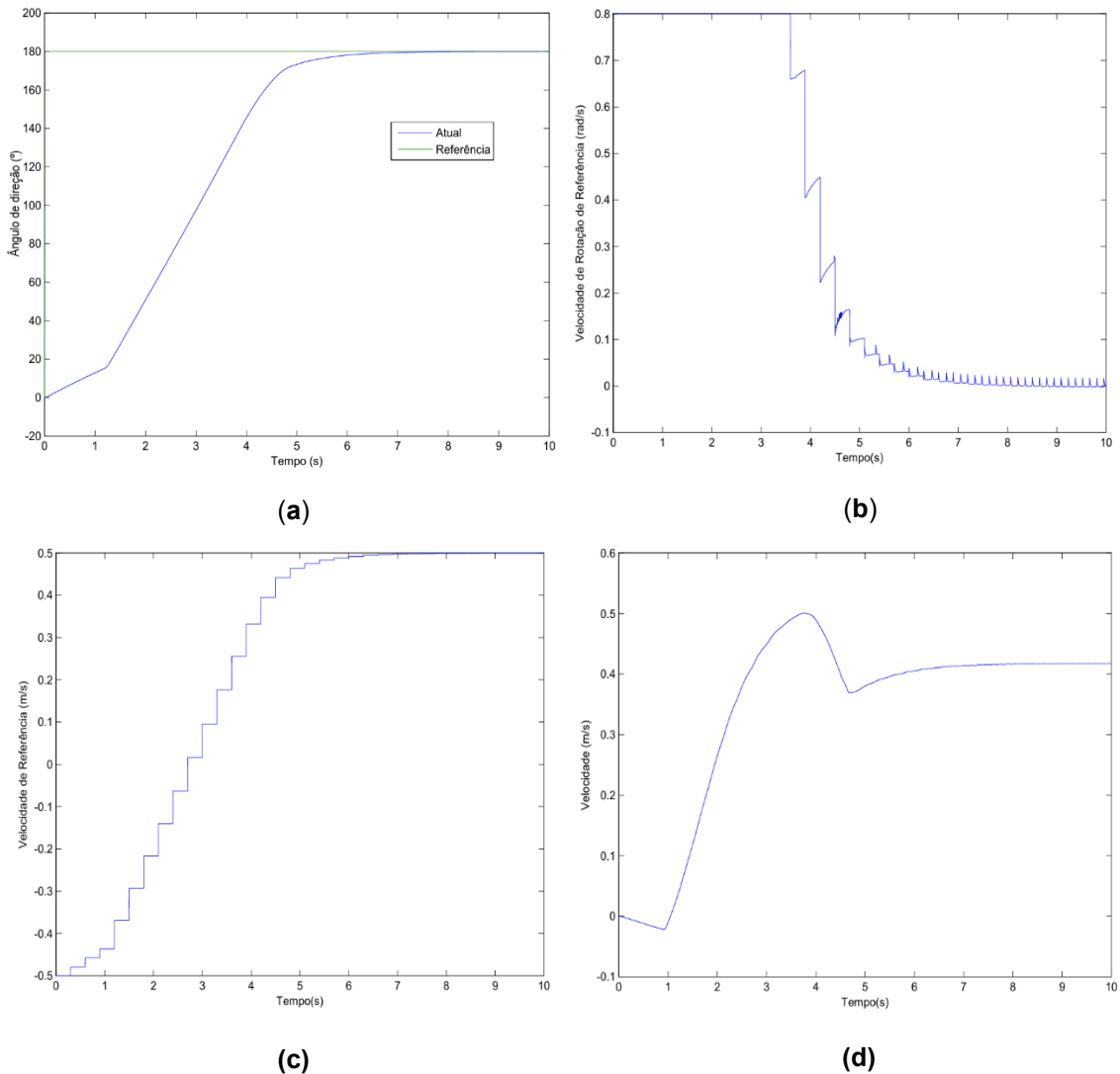


Figura 5.9 – Simulação do sistema para quando o barco faz uma rotação de 180°: **(a)** Reação do ângulo de direção; **(b)** Reação da velocidade de rotação de referência; **(c)** Reação da velocidade linear de referência; **(d)** Reação da velocidade linear.

Na reação do ângulo, observa-se que o tempo de subida de 10% a 90% do valor demora cerca de 3 segundos, estando o barco limitado fisicamente pela velocidade de rotação máxima. A reação da velocidade de rotação de referência não tem um comportamento mais linear devido ao atraso adicionado de 0,3 segundos.

Na reação da velocidade linear de referência, observa-se que o gráfico acompanha a variação do ângulo com um atraso de amostragem, impondo uma velocidade de referência para trás nos primeiros 3 segundos e uma velocidade de referência para a frente no restante tempo, devido à função de ajuste definida na equação (3.1).

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

Na reação da velocidade linear, observa-se que no primeiro segundo o barco tenta acompanhar a referência, mas não consegue devido às limitações físicas de eficiência de rotação do motor para trás. Entre 1 a 5 segundos o comando da velocidade linear não consegue acompanhar a referência devido à perturbação causada pelo controlo de direção, pois ao impor um diferencial nos motores, devido à diferença de eficiência de rotação para trás e para a frente, impõe uma força comum que provoca uma perturbação na velocidade linear. No restante tempo a velocidade linear acompanha a referência com um certo erro, devido a não existir *feedback* no comando da velocidade linear.

Também se simula a variação da velocidade de rotação do barco ao longo do tempo, comparando-a com a sua respetiva referência como mostra a Figura 5.10.

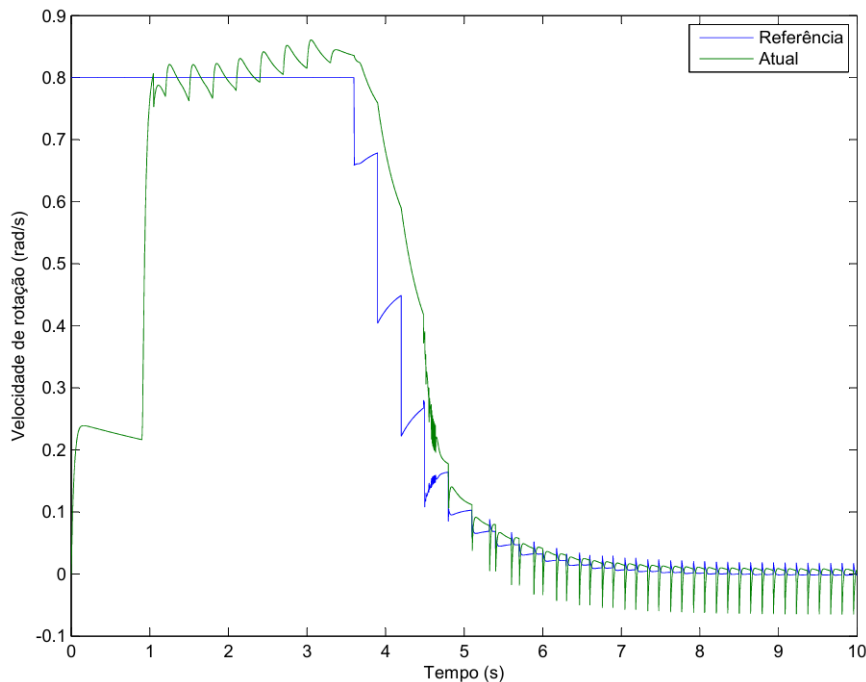


Figura 5.10 – Reação da velocidade de rotação em função do tempo.

Na reação da velocidade de rotação, no primeiro segundo a velocidade de rotação do barco não consegue acompanhar a referência pelo facto dos motores estarem a rodar para trás, o que os torna menos eficientes. No restante do tempo a velocidade de rotação atual acompanha a referência com um atraso temporal de 0,3 segundos.

De maneira a visualizar melhor a reação do sistema de controlo da velocidade de rotação, têm-se como referência uma velocidade de rotação

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

constante de 0,8 rad/s, aplicando uma entrada em degrau, e impôs-se μ_{com} a zero para não perturbar o sistema. Obtendo a reação apresentada na Figura 5.11.

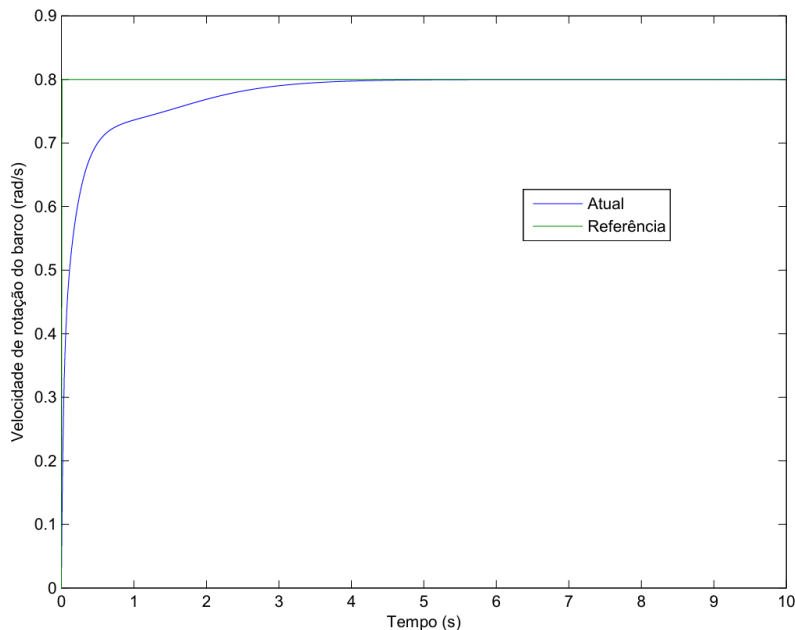


Figura 5.11 – Resposta do sistema a uma variação em degrau na velocidade de rotação de referência com uma amplitude de 0,8 rad/s.

Na reação da velocidade de rotação, observa-se que o tempo de subida é de cerca de 0,6 segundos, como projetado na teoria. Também se observa que no restante do tempo, o termo de momento linear, $m * v$, do barco domina o atraso em relação ao termo de atrito, C_{ω} , alterando a reação teórica do sistema.

Os resultados da Figura 5.12 mostram a variação da posição do barco ao longo do tempo num plano XY , em metros para uma rotação de 180° . Como o barco começa orientado no eixo dos Xs , o valor do eixo dos Ys é o desvio lateral do barco, ou seja, quanto se afastou lateralmente da posição inicial em relação à trajetória pretendida, e o eixo dos Xs é o deslocamento na direção pretendida, orientado na trajetória.

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

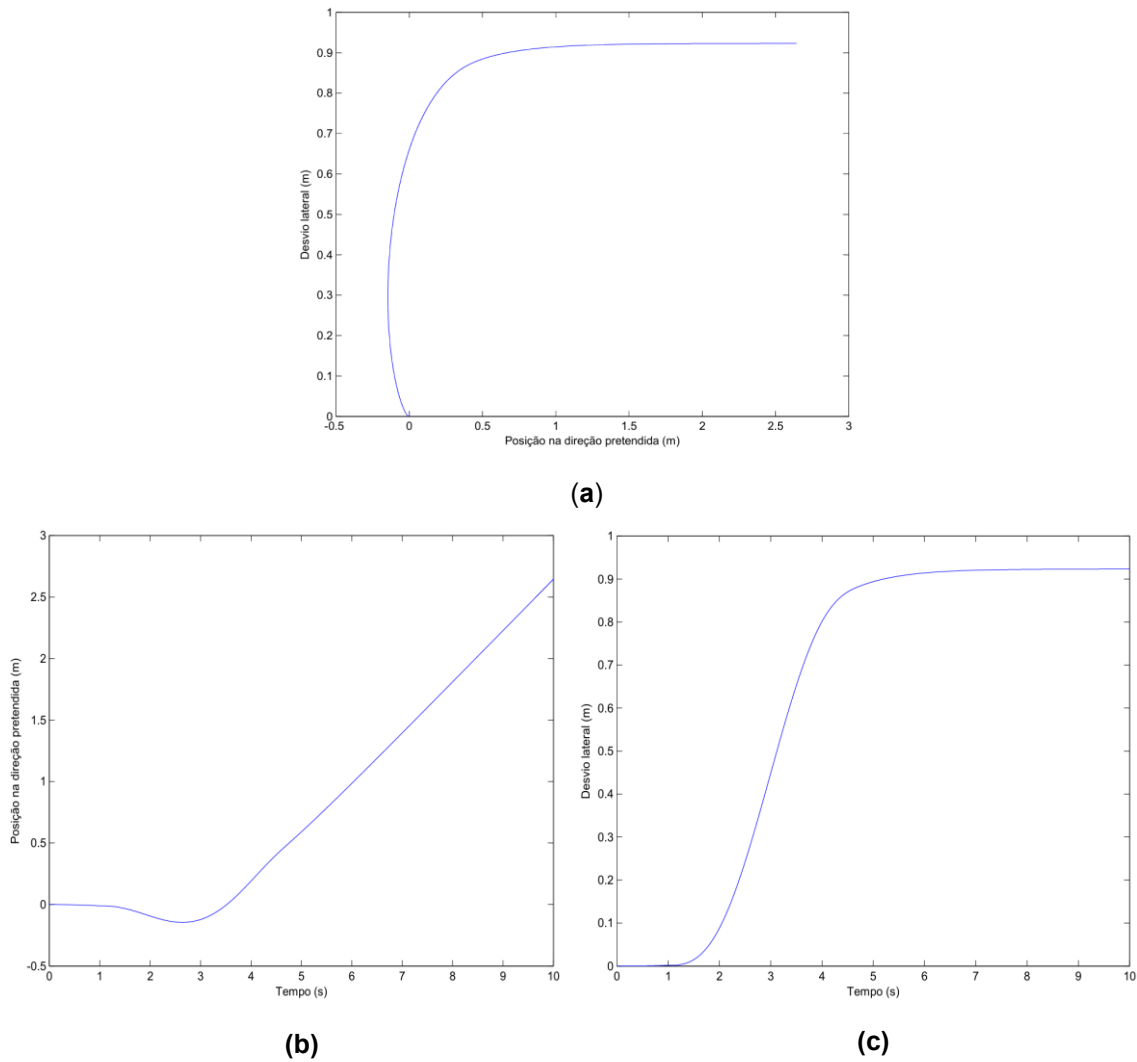


Figura 5.12 – Variação da posição do barco ao longo do tempo: **(a)** Trajeto do barco; **(b)** Posição do barco na direção pretendida; **(c)** Desvio lateral do barco.

Observa-se que para uma rotação de 180° , o barco apresenta um desvio de posição lateral inferior a 1 metro, (a) e (c), sendo que a posição no eixo dos Ys não ultrapassa esse valor. Com isso observa-se que o barco necessita que entre pontos exista pelo menos 1 metro de margem sem obstáculos laterais.

Também se observa que demora cerca de 3 segundos para o barco começar a mover-se na direção pretendida (b) e que a maior parte do desvio lateral acontece entre os 2 a 4 segundos (c), pois nos primeiros 2 segundos o barco realiza um movimento para trás, movimento de menor eficiência.

Para analisar o comportamento dos controladores do barco faz-se uma variação da direção de referência, para os ângulos de 90° , -90° e -180° , com uma velocidade linear inicial do barco de 0,5 m/s, como mostra a Figura 5.13, que apresenta a reação do barco para os diferentes ângulos de referência.

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

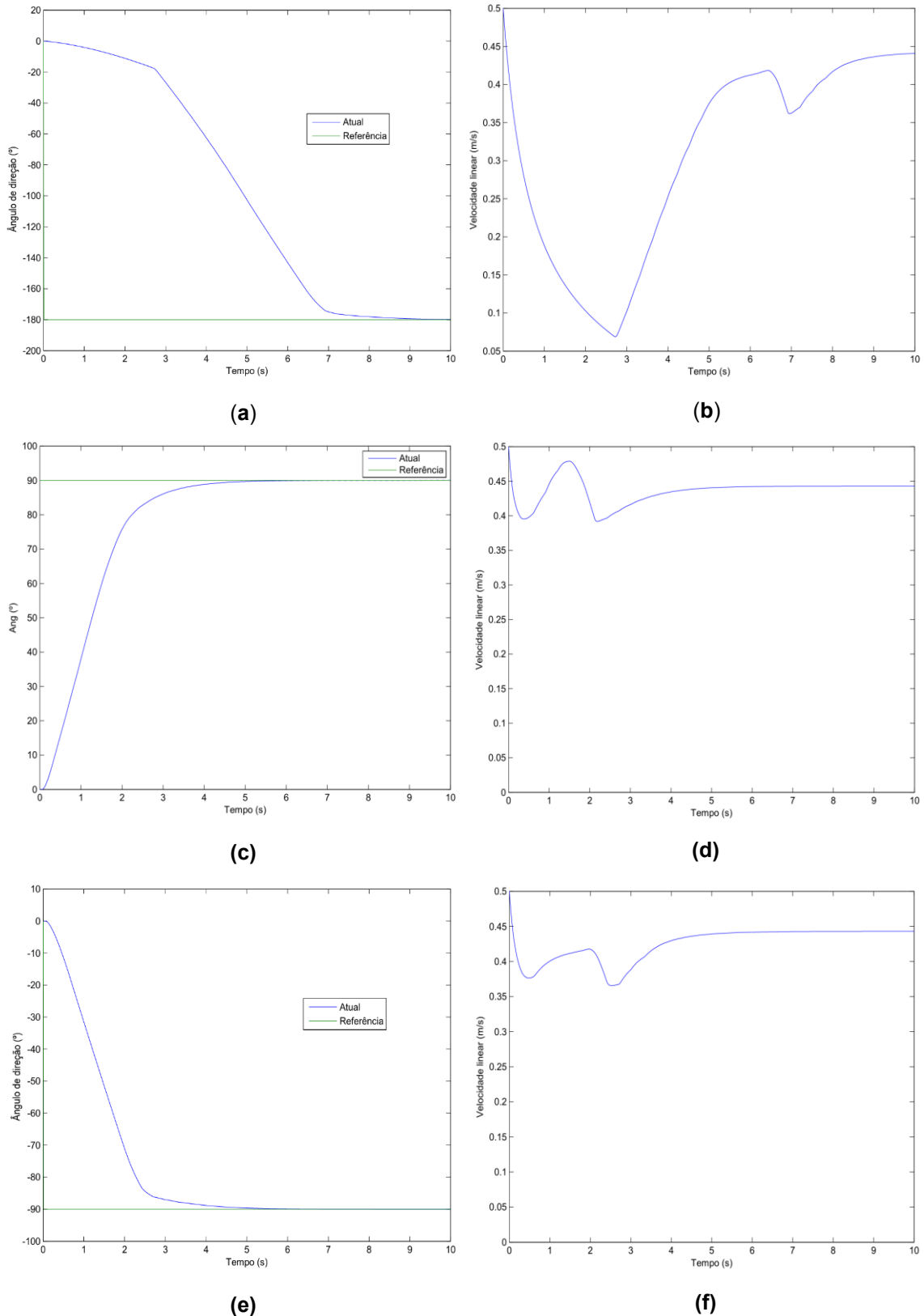


Figura 5.13 – Simulação do sistema do barco: **(a)** Reação do ângulo de direção para uma rotação de -180° ; **(b)** Reação da velocidade linear para uma rotação de -180° ; **(c)** Reação do ângulo de direção para uma rotação de 90° ; **(d)** Reação da velocidade linear para uma rotação de 90° ; **(e)** Reação do ângulo de direção para uma rotação de -90° ; **(f)** Reação da velocidade linear para uma rotação de -90° .

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

Observa-se que o sistema do barco consegue reagir para as diversas direções. As rotações de valor negativo perturbam menos a velocidade linear do que as rotações de valor positivo, devido ao facto dos motores do barco escolhido terem características diferentes na eficiência, sendo o motor esquerdo menos eficiente que o motor direito. Para as rotações de $\pm 90^\circ$, o tempo de subida é de cerca de 2 segundos, como o projetado na teoria. Na rotação de -180° observa-se que o barco demora mais tempo, cerca de 3 segundos, a realizar a rotação para trás em comparação com a simulação de 180° .

Isto deve-se ao facto da velocidade de rotação do barco ser perturbada pela velocidade linear inicial do barco, sendo que no início a força realizada pelo barco serve para simplesmente travar essa velocidade linear inicial. Denota-se que mesmo com motores diferentes o ângulo da direção do barco tende para o valor de referência, pois as perturbações que acontecem no barco são compensadas pelo termo integral do controlador de velocidade de rotação.

Para observar a importância do ajuste da velocidade linear quando o barco muda de direção, testou-se o caso em que o comando da velocidade linear é constante. Para tal criou-se uma sequência de transição da direção de referência de $0^\circ, 90^\circ, 180^\circ, 90^\circ, 0^\circ$ com 5 segundos em cada direção. Tendo em conta isso, simula-se o trajeto pretendido, o trajeto sem ajuste da velocidade linear e o trajeto com ajuste da velocidade linear do barco, como mostra a Figura 5.14, no plano XY ao longo do tempo.

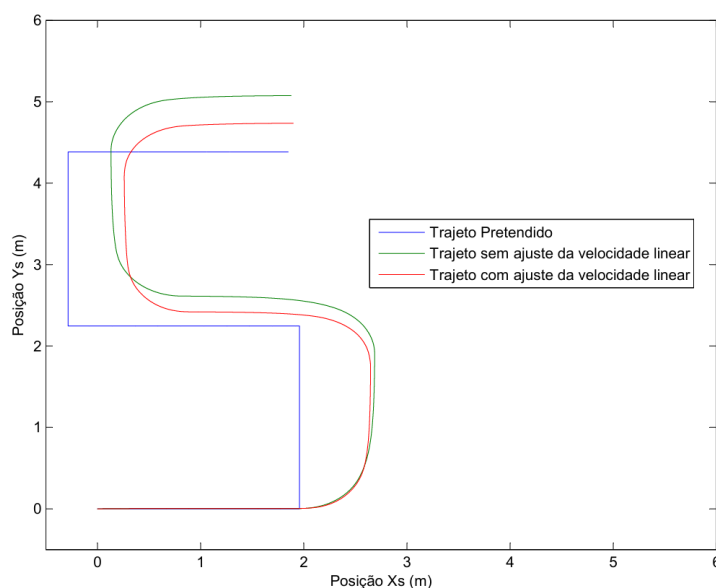


Figura 5.14 – Simulação da reação da posição do barco com e sem ajuste da velocidade linear.

Capítulo 5 – Simulação do sistema de navegação do barco autônomo

Observa-se que com o ajuste da velocidade linear, o barco acompanha melhor a referência. Para uma rotação de 90° , o desvio em relação ao trajeto pretendido, para ambos os casos, é cerca de 0,5 metros. A importância do ajuste da velocidade linear é melhor observada para uma rotação de 180° , onde o desvio no eixo dos Ys do trajeto com ajuste é cerca de metade do trajeto sem ajuste, sendo esse desvio cerca de 0,5 metros.

Realizam-se simulações ao barco autônomo quando é adicionado ruído na amostragem na leitura do ângulo do barco, para avaliar o efeito devido às imperfeições que podem surgir num sistema real, juntamente com uma perturbação na velocidade de rotação, devido às ondas marítimas. Estuda-se, assim o comportamento do sistema em condições mais próximas daquelas que se espera obter nas experiências práticas, com o barco autônomo real.

A velocidade de rotação é perturbada por um sinal descrito através de uma variável gaussiana discreta com potência de 0,1 W e tempo de amostragem de 0,3 segundos, obtendo-se a velocidade de rotação e a respetiva referência com perturbação. O sensor de direção capta ruído aditivo através de uma variável gaussiana discreta com potência de 0,1 W e tempo de amostragem de 0,01 segundos, obtendo-se o ângulo da direção com e sem ruído, para uma rotação de 180° . Os valores da potência e do tempo de amostragem das variáveis aleatórias foram escolhidos empiricamente tendo em conta o ruído e as perturbações observados na prática, tendo variações no ângulo devido ao ruído de $\pm 10^\circ$ e podendo acontecer perturbações fortes na velocidade de rotação do barco de $\pm 0,5$ rad/s. As características do sensor de direção indicam o tempo de amostragem, e o tempo de amostragem da perturbação adicionada na velocidade de rotação é igual ao atraso de processamento.

Na Figura 5.15 apresenta-se a reação do sistema com perturbação e ruído adicionados, observa-se a reação da velocidade de rotação e da respetiva referência (a) e a reação da direção do barco para o sinal amostrado de direção com e sem ruído (b).

Capítulo 5 – Simulação do sistema de navegação do barco autónomo

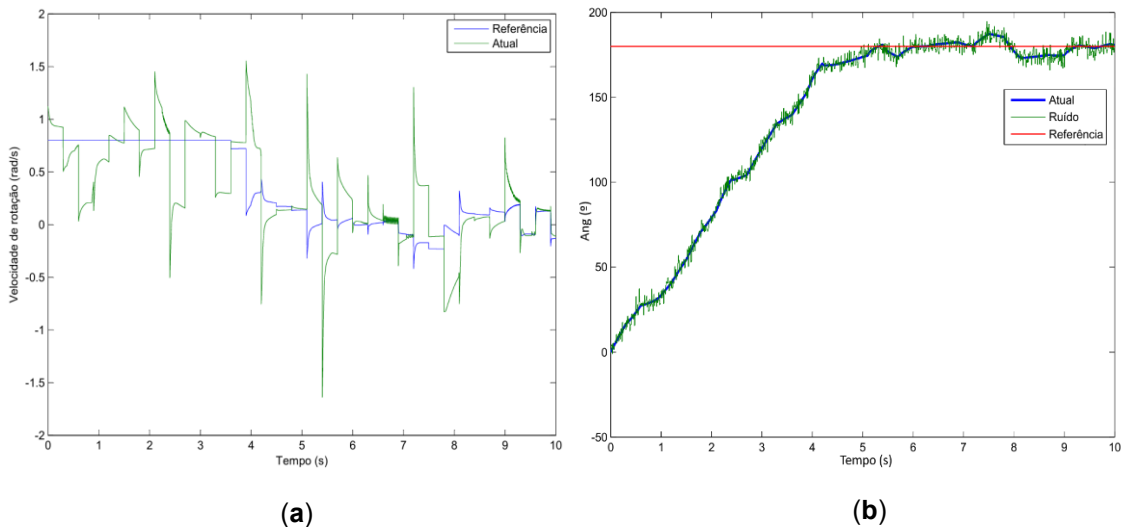


Figura 5.15 – Simulação do sistema com ruído e perturbação: **(a)** Reação da velocidade de rotação; **(b)** Reação da direção.

Observa-se que o sistema continua a reagir e a tender para o ângulo de referência de 180° . Observa-se que a velocidade de rotação de referência tenta contrariar a perturbação através do termo derivativo do controlador de direção, como por exemplo ao fim de 5 segundos, que ao detetar uma variação abrupta impõe na velocidade de rotação de referência uma variação no sentido oposto.

5.3 Considerações finais sobre as simulações

Em resumo, para realizar a simulação do barco autónomo, implementaram-se os 4 submodelos: o modelo dos motores DC; o modelo de Yoerger; o equivalente das forças; e o modelo hidrodinâmico do barco. Juntamente com a atuação do sistema, composta pelo controlador de direção, comando de velocidade linear e circuito de acionamento do sistema, adicionando os limites físicos e atrasos de processamento nestes componentes. Simulou-se a variação do ângulo de direção, da velocidade linear do barco, da velocidade de rotação do barco e da posição ao longo do tempo.

Observa-se que as reações projetadas na teoria do Capítulo 4, correspondem às reações obtidas na simulação do sistema, com um tempo de subida da velocidade de rotação de cerca 0,6 segundos e um tempo de subida do ângulo de direção para uma diferença de 90° de cerca de 2 segundos. Tendo a simulação realizada, procede-se para a implementação do sistema na prática e realização dos testes para validação do sistema do barco autónomo.

Capítulo 6

IMPLEMENTAÇÃO E TESTES DO BARCO AUTÓNOMO

Neste capítulo apresenta-se a escolha dos materiais utilizados e os circuitos desenvolvidos para implementar o sistema do barco autónomo. Escolhem-se os motores, e respetivos *drivers*, os sensores de posição e direção, os circuitos de processamento e condicionamento de sinais, os microcontroladores e o rádio no sistema de telecomunicações entre o comando e o barco, o desenvolvimento do circuito do telecomando e a escolha do barco. Também se apresentam os testes ao barco autónomo e analisam-se os resultados. Com os resultados obtidos verifica-se se os objetivos foram atingidos. São descritas as características da solução apresentada. Os testes são feitos ao sistema de comunicação, ao controlo manual do barco, de mudança de direção, de navegação autónoma e de alcance da comunicação entre o comando e o barco.

6.1 Escolha do barco, dos motores e *drivers*

Primeiramente, foi preciso adquirir a estrutura do barco e escolheu-se utilizar um barco pré-fabricado, do modelo *Bilbear Feeding Boat with GPS Carp* [37], apresentado na Figura 6.1; com o objetivo de reutilizar a estrutura do barco e os motores DC, modificando os restantes componentes. Escolheu-se utilizar este barco para simplificar o trabalho realizado.



Figura 6.1 – Barco escolhido para implementação do sistema autónomo.

Como o barco era pré-fabricado já incluía os motores adaptados à embarcação e os *driver* associados. O *driver* utilizado é o RZ7886 [38]. Foi necessário adquirir o *driver* para desenvolver o circuito de acionamento dos motores. É de notar que todos os circuitos foram desenvolvidos neste trabalho para substituir o sistema do fabricante do barco, pois o sistema do fabricante é um sistema sem possibilidade de configuração.

Realizando os testes aos motores, apresentados no Apêndice B, na secção B.1 e B.3, obtêm-se as características dos motores DC desse barco, e que são apresentados na Tabela B.3.

Após a escolha do barco, motores e *drivers*, procede-se à escolha dos sensores para a obtenção da posição e direção do barco autónomo.

6.2 Sensores para obtenção da posição e direção do barco autónomo

Para obter a posição do barco pretende-se utilizar um sensor que tenha baixo custo, uma exatidão e precisão na ordem de 10 metros, para se ter um certo nível de confiança da posição do barco, e que consiga amostrar a posição no máximo em 5 segundos. Daí utiliza-se o sensor de posição GY-GPSV3 NEO-M8N [39], que tem as características de baixo custo, precisão e exatidão de 3 metros e tempo de amostragem de 1 segundo.

Para obter a direção do barco pretende-se utilizar um sensor que tenha baixo custo, uma exatidão e precisão na ordem de 10° , para se ter um certo nível de confiança da direção do barco, e que inclua a amostragem da velocidade de rotação do barco. Daí utiliza-se o sensor de direção 9DOF (*Nine Degrees Of Freedom*) LSM9DS1 [40], que tem baixo custo, precisão na ordem de 10° e frequência de amostragem de 100 Hz.

Para estimar a direção do barco, utiliza-se um magnetómetro e o acelerómetro do sensor 9DOF para calcular os ângulos de rotação em três dimensões (X, Y e Z) [41]. Coloca-se o sensor 9DOF orientado com o eixo dos Ys alinhado na direção do barco e o eixo dos Zs alinhado com a gravidade da terra, como mostra a Figura 6.2.

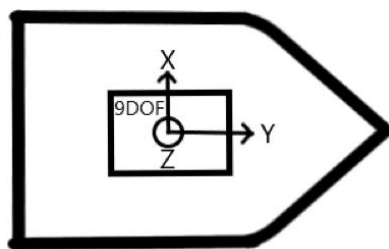


Figura 6.2 – Orientação do sensor 9DOF no barco (visto de cima).

O livro [41] indica que o ângulo de enrolamento, ψ_{roll} , ângulo Xs do sensor 9DOF com o plano marítimo, é dado pelo arco tangente da divisão da aceleração no eixo dos Xs, Acl_{Xs} , pela aceleração no eixo dos Zs, Acl_{Zs} , como mostra a equação (6.1). O ângulo de inclinação, ψ_{pitch} , ângulo Ys do sensor 9DOF com o plano marítimo, é dado pelo arco tangente da divisão da aceleração no eixo dos Ys, Acl_{Ys} , pela raiz quadrada da soma do quadrado da aceleração no eixo dos Xs com a aceleração no eixo dos Zs, como mostra a equação (6.2). O sensor

utiliza o valor da aceleração gravítica terrestre como referência para a determinação desses ângulos.

$$\psi_{roll} = arctg\left(\frac{Acl_{Xs}}{Acl_{Zs}}\right) \quad (6.1)$$

$$\psi_{pitch} = arctg\left(\frac{Acl_{Ys}}{\sqrt{(Acl_{Zs})^2 + (Acl_{Xs})^2}}\right) \quad (6.2)$$

Para estimar o ângulo da direção utiliza-se o campo magnético terrestre como referência, medido nas três dimensões através do magnetómetro. O livro [41] indica que o ângulo de direção do barco em relação ao norte magnético é dado pela equação (6.3). Sendo θ o ângulo da direção e Mag_{Xs} , Mag_{Ys} , Mag_{Zs} os valores do campo magnético no eixo dos Xs , Ys e Zs , respetivamente.

$$\theta = arctg\left(\frac{\sin(\psi_{roll}) * Mag_{Zs} + \cos(\psi_{roll}) * Mag_{Xs}}{\cos(\psi_{pitch}) * Mag_{Ys} + \sin(\psi_{pitch}) * (\cos(\psi_{roll}) * Mag_{Zs} + \sin(\psi_{roll}) * Mag_{Xs})}\right) \quad (6.3)$$

Os sensores foram calibrados através do programa *MotionCal* [42] e o código respetivo é apresentado no Apêndice C, na secção C.1. Na secção C.1 é também descrito o código de implementação e amostragem dos valores dos sensores.

Tendo os sensores de posição e direção escolhidos e calibrados, procede-se para a escolha dos microcontroladores e rádios utilizados para a implementação do barco autónomo.

6.3 Microcontroladores e rádios do barco autónomo

Para executar o processamento dos dados no barco e telecomando utiliza-se um microcontrolador que facilita a programação e interação com o sistema. Pretende-se, no microcontrolador, programar em linguagem Python, ter disponibilidade de bibliotecas de código para aceder aos sensores e sistema de comunicação, ter memória e processamento suficiente para executar o código e guardar os valores das variáveis em ficheiros e na memória do sistema para futura análise. Daí escolheu-se utilizar o microcontrolador Raspberry Pi Pico [43], por ter essas mesmas características, tendo 2 núcleos no processador e uma memória de 256 kB.

Para estabelecer a comunicação entre o barco e o telecomando pretende-se utilizar um rádio de baixo custo e com uma frequência de operação de 433 MHz para ter um maior alcance que as frequências de operação mais comuns de 2,4 GHz. Daí escolheu-se utilizar o rádio nRF905 [22], pelo seu baixo custo, frequência de operação de 433 MHz e -100 dBm de sensibilidade, juntamente com antenas de 433MHz [44], antenas monopolo que permitem captar o sinal de qualquer direção.

O código desenvolvido para estabelecer a comunicação é apresentado no Apêndice C, na secção C.2. O rádio do barco é inicializado no modo de escuta e o rádio do telecomando realiza a transmissão dos pacotes para o barco executar os comandos. O sistema constrói a mensagem do telecomando utilizando o primeiro *byte* da mensagem como o seletor do comando a executar, sendo 0x80 o comando do controlo manual, 0x40 o comando do controlo autónomo e 0x20 o comando de nova trajetória.

No controlo manual utilizam-se os 4 *bytes* seguintes da mensagem para transmitir a informação sobre o sinal PWM a aplicar a cada motor. No controlo autónomo é somente transmitido o comando sem conteúdo restante na mensagem. Na transmissão de nova trajetória, utilizam-se os restantes *bytes* da mensagem para transmitir as coordenadas GPS, podendo transmitir até 3 locais. É utilizado o protocolo SPI (*Serial Peripheral Interface*) [45] para estabelecer a comunicação entre o microcontrolador e o rádio, utilizando a sequência de comandos descritos na folha de características do rádio [22] para inicialmente o configurar e seguidamente transmitir ou receber mensagens. Na Figura 6.3 apresenta-se o formato da trama da mensagem para os diversos comandos como descrito anteriormente.

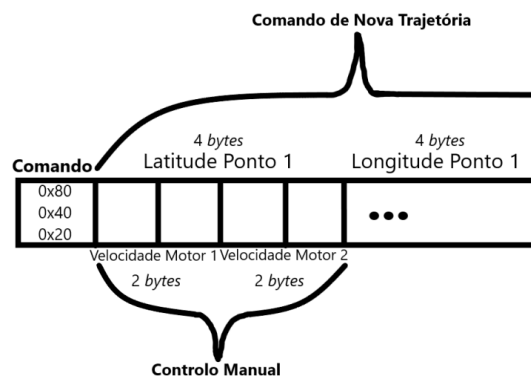


Figura 6.3 – Formato da trama da mensagem transmitida pelo telecomando.

Com os microcontroladores e rádios escolhidos, e a lógica do código de interação definida, avança-se para a implementação do telecomando do barco

6.4 Telecomando do barco

Utilizando o microcontrolador e o rádio descritos na secção 6.3, constrói-se o circuito do telecomando. Utilizam-se potenciómetros para indicar o valor equivalente de velocidade em cada motor, sendo o motor esquerdo dado pelo potenciómetro 1, *P1*, e o motor direito dado pelo potenciómetro 2, *P2*, variando a entrada entre 0 Volts e 3,3 Volts. O valor de 0 Volts corresponde a um valor de *duty-cycle* do sinal PWM aplicado aos motores de 1, no par de transístores de rotação para trás, o valor de 3,3 Volts corresponde a um valor de *duty-cycle* do sinal PWM aplicado aos motores de 1, no par de transístores de rotação para a frente, e o valor de 1,65 Volts corresponde a um valor de *duty-cycle* do sinal PWM aplicado aos motores de 0.

As tensões são lidas pelo conversor analógico-digital do microcontrolador, fazendo a leitura do valor equivalente em 16 *bits*. Também se utilizam dois botões, para juntar à funcionalidade de controlo manual as outras duas funcionalidades: trocar o modo de funcionamento utiliza-se o botão 1, *B1*; enviar coordenadas GPS para definir o percurso de referência do barco autónomo utiliza-se o botão 2, *B2*.

Para o rádio do telecomando entrar em modo de transmissão é preciso ligar logicamente a 1 os pinos TXEN e PWR, sendo o TXEN a entrada que indica se o rádio está em modo de transmissão ou modo de receção e PWR a entrada que indica se o rádio está em modo de adormecimento ou em modo de funcionamento.

O pino DR, quando em modo de receção, indica se o rádio já recebeu a mensagem para o controlador processar e o pino CE é ligado logicamente a 1 para transmitir a mensagem quando o rádio estiver em modo de transmissão.

Fornece-se a energia ao circuito através do microcontrolador, Raspberry Pi Pico, com os pinos de energia de 3V3_OUT e GND, ligando o microcontrolador ao computador, e realiza-se a comunicação com o rádio, nRF905, através do protocolo SPI [45], ligando os pinos do rádio (MISO, MOSI,

Capítulo 6 – Implementação e testes do barco autónomo

SCK e CSN) aos pinos GP10, GP11, GP12 e GP13 do microcontrolador, programados no código apresentado no Apêndice C, na secção C.2 (Figura C.12). O circuito equivalente é apresentado na Figura 6.4. Na Figura 6.5 visualizam-se os componentes de interação com o utilizador, potenciômetros *P1* e *P2*, e botões *B1* e *B2*, ligados ao microcontrolador e rádio através de uma placa de circuito impresso. Também é de notar que se liga o computador ao microcontrolador através dum cabo Universal Serial Bus para alterar as coordenadas e o código.

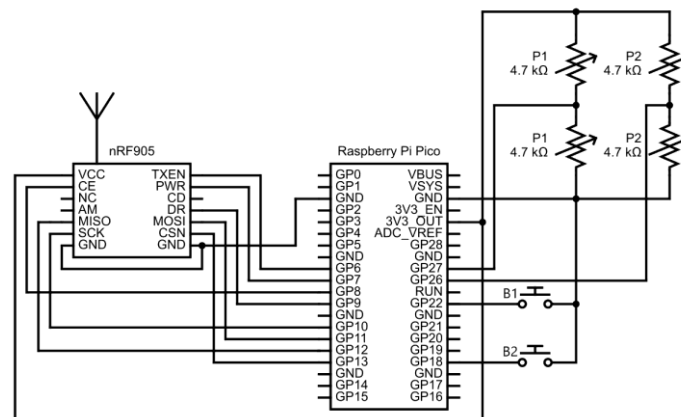


Figura 6.4 – Circuito equivalente do telecomando do barco autónomo.

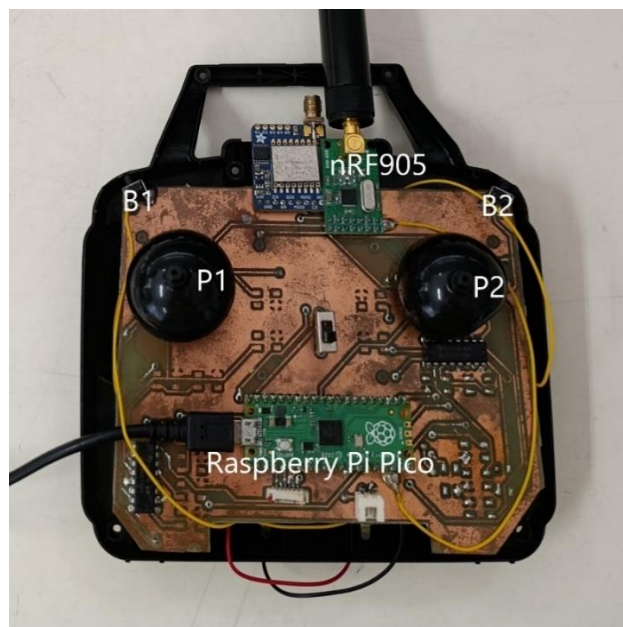


Figura 6.5 – Periféricos de entrada do telecomando do barco autónomo.

Tendo o telecomando do barco, com o respetivo circuito, implementado procede-se para a implementação do circuito do sistema do barco.

Apêndice C, na secção C.1. Para o sensor de GPS e o sensor 9DOF é somente preciso ligar os pinos de comunicação e energia. Na Figura 6.7 apresenta-se o circuito equivalente da amostragem dos dados dos sensores.

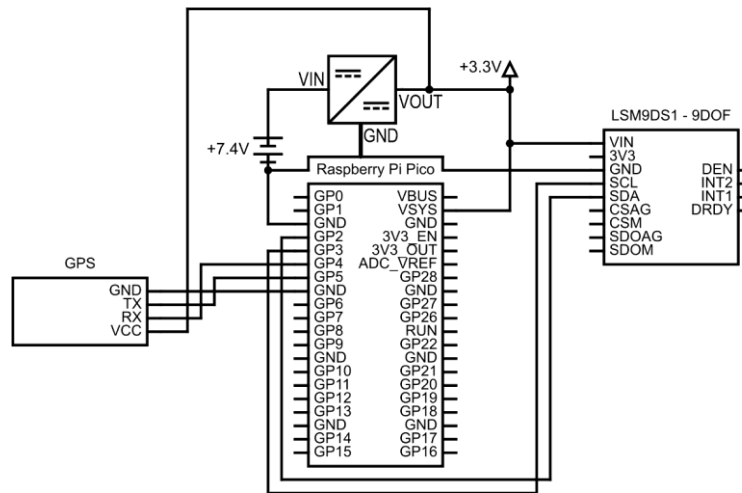


Figura 6.7 – Circuito equivalente da amostragem dos dados dos sensores *GPS* e *9DOF* do barco autónomo.

O microcontrolador do barco autónomo recebe os dados do telecomando e dos sensores GPS e 9DOF e atua no *driver* [38] dos motores. O comando dos motores do barco autónomo é feito através do envio de um sinal PWM nos interruptores da ponte-H. A ponte-H faz com que a bateria seja ligada e desligada dos motores tendo em conta o sinal PWM enviado, para impor um valor médio de tensão aos terminais dos motores e assim regular a velocidade de rotação do barco. No *driver* dos motores DC a entrada BI liga o motor para trás, ou seja, aplica o simétrico da tensão da bateria, e a entrada FI liga o motor para a frente, ou seja, aplica a tensão da bateria. Também é de notar que se liga diretamente a bateria ao *driver* e que a frequência da portadora do sinal PWM é de 1 kHz. Na Figura 6.8 apresenta-se o circuito de acionamento dos motores do barco autónomo, ligando a bateria e o conversor DC/DC ao Raspberry Pi Pico que comanda os *drivers* RZ7886 e respetivos motores.

Capítulo 6 – Implementação e testes do barco autónomo

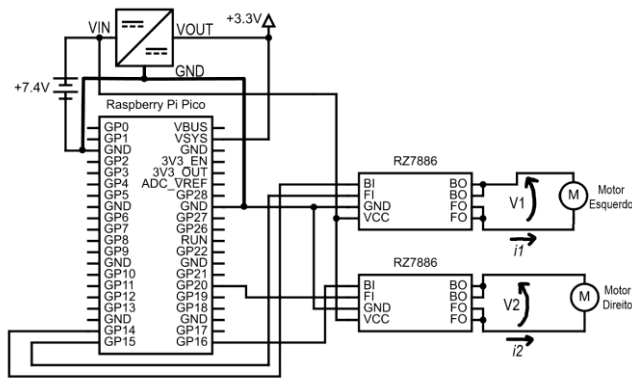


Figura 6.8 – Circuito equivalente do acionamento dos motores do barco autónomo.

O sistema do barco autónomo é constituído pelos circuitos do telecomando, constituído por microcontrolador, rádio e periféricos e o circuito do barco autónomo, constituído por bateria, conversor DC/DC, antena, microcontrolador, sensores (GPS e 9DOF), *driver* dos motores e motores DC. Os circuitos foram implementados em placas de circuito impresso. Juntando todas as partes, descritas da secção 6.1 à 6.5, obtém-se o sistema do barco autónomo, que em conjunto com o código global e o circuito equivalente global é apresentado no Apêndice C, na secção C.4 (Figura C.16 e Figura C.17). O circuito do barco e do telecomando é apresentado na Figura 6.9, com os componentes descritos anteriormente. Também é de notar que o sistema de controlo apresenta um atraso de 0,3 segundos por ciclo de atuação.



Figura 6.9 – Circuitos do barco autónomo e telecomando.

Tendo desenvolvido o sistema do barco procede-se para a realização dos testes do barco autónomo, começando pelos testes de comunicação.

controlo manual. Depois transmite-se a mensagem de entrada no controlo autónomo e seguidamente volta-se ao controlo manual. Finaliza-se o teste com a transmissão da mensagem de definição de nova trajetória. O código da transmissão das mensagens entre o telecomando e o barco é apresentado no Apêndice C, na secção C.2, obtendo-se na consola as mensagens apresentadas na Figura 6.12, que representam os comandos executados.

```
Shell <
bytearray(b't\x0cD \xa2\xa2\xa2\xa2\xd8')
Controlo Manual!
Controlo Manual!
Controlo Manual!
Controlo Autonomo
Controlo Autonomo
Controlo Autonomo
Controlo Manual!
Controlo Manual!
Controlo Manual!
Nova Trajetoria
Controlo Manual!
```

Figura 6.12 – Troca dos estados de funcionamento do barco visualizada na consola.

Observa-se que o sistema de comunicação do barco está a enviar os comandos para o barco e que são lidos e processados pelo microcontrolador do barco autónomo, seguindo a sequência descrita anteriormente. O comando de controlo manual e controlo autónomo repetem a mensagem enviada para a consola, porque estão em ciclo até o utilizador trocar com o botão *B1*. No comando de nova trajetória o programa executa o código uma vez para trocar os dados da trajetória, não repetindo a mensagem enviada para a consola.

Tendo o sistema de comunicação entre o telecomando e o barco a funcionar, procede-se para o teste do funcionamento do controlo manual do barco.

6.7 Comando do barco

Para testar o controlo manual, encheu-se um tanque de água e colocou-se o barco na água. Verifica-se que ao enviar os comandos pelo telecomando para o barco este acionava o motor correspondente, comandando o movimento do barco. Testaram-se ambos os motores ajustando os valores dos potenciômetros. Variando as combinações dos comandos enviados para os motores observou-se o correto funcionamento do comando manual, como mostra a Figura 6.13, com uma foto do barco e um diagrama mostrando o motor esquerdo a rodar para trás e mostrando o motor direito a rodar para a frente. Os

vídeos gravados com a reação do barco são apresentados no Apêndice D, na secção D.1, e comprovam o correto funcionamento.

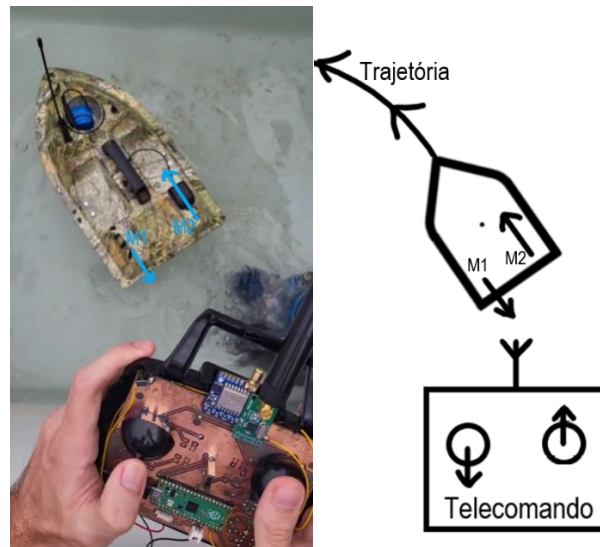


Figura 6.13 – Barco a reagir aos comandos do controlo manual do barco.

Os vídeos gravados comprovam que o barco para rodar 90° para a esquerda e para a frente, com o motor direito a rodar para a frente e o motor esquerdo a rodar para trás, demora cerca de 1,78 segundos. Para rodar 90° para a esquerda e para trás, com o motor direito a rodar para trás e o motor esquerdo desligado, demora cerca de 7,43 segundos. Para rodar 90° para a direita e para a frente, com o motor esquerdo a rodar para a frente e motor direito a rodar para trás, demora cerca de 2 segundos. Para rodar 90° para a direita e para trás, com o motor esquerdo a rodar para trás e o motor direito desligado, demora cerca de 10 segundos. A diferença entre os motores provem de maior desgaste de um deles em função do outro. O barco a andar para a frente tem uma velocidade de 0,6 metros por segundo e o barco a andar para trás barco uma velocidade de 0,2 metros por segundo.

Tendo o comando manual do barco validado, procede-se para o teste de mudança de direção do barco autónomo.

6.8 Teste de mudança de direção do barco autónomo

Para testar o funcionamento do sistema de controlo de direção do barco, montou-se uma piscina com aproximadamente 5 metros de diâmetro e visualizou-se a variação do ângulo de direção do barco quando é definida para

Capítulo 6 – Implementação e testes do barco autónomo

direção de referência o norte magnético e quando é colocado o barco na direção oposta à direção de referência e afastado 1 metro do centro da piscina na direção oposta à direção de referência. O procedimento de registo e a leitura dos dados foi feito ao adicionar uma linha de código de amostragem a cada ciclo do controlo de direção, guardando os dados como mostra o código global do Apêndice C, na secção C.4 (Figura C.16). A Figura 6.14 apresenta a piscina utilizada para realizar os testes dos sistemas de controlo de navegação do barco autónomo.



Figura 6.14 – Piscina montada para realizar os testes de controlo de direção.

Na secção D.2 do Apêndice D estão as hiperligações dos vídeos com os registos dos testes realizados nessa piscina para verificação do funcionamento do controlo da direção. Coloca-se o barco aproximadamente na direção oposta à direção de referência e observa-se a variação da direção ao longo do tempo utilizando os controladores definidos, no Capítulo 4, e utilizados na simulação, no Capítulo 5. Obtém-se na Figura 6.15, os resultados de simulação e experimentais da variação do ângulo de direção do barco.

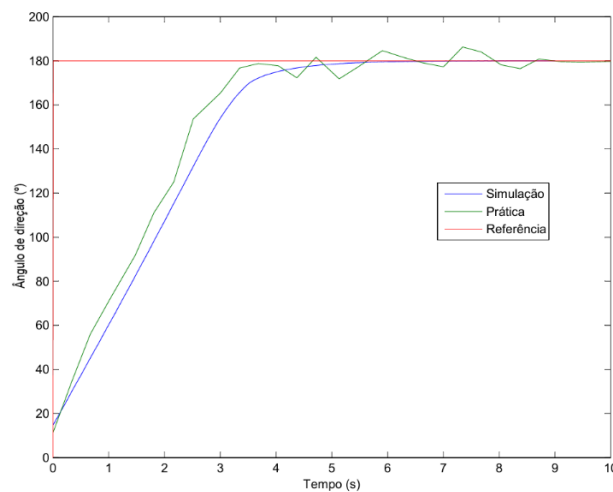


Figura 6.15 – Variação do ângulo de direção, em função do tempo, do barco autónomo com o controlo de direção.

Capítulo 6 – Implementação e testes do barco autónomo

Observa-se que o tempo de subida do ângulo de direção, de 10% a 90%, demora cerca de 3 segundos. O barco não realizou a rotação para trás devido a ter começado a reação prática do ângulo com 10°. No estado estacionário da reação prática o ângulo oscila devido ao ruído dos sensores e perturbações no sistema.

Observa-se que a reação da simulação e a reação da prática são semelhantes, tendo aproximadamente o mesmo tempo de subida.

Testa-se, também, a reação do barco para uma mudança de direção de 150°. Coloca-se o barco a percorrer um trajeto de dois pontos e regista-se a transição quando chega a um deles. Assim, o barco ao realizar uma mudança de direção, tem uma velocidade linear inicial de 0,5 m/s. Na Figura 6.16 observa-se a variação da direção ao longo do tempo para essas condições, com os resultados de simulação e experimentais.

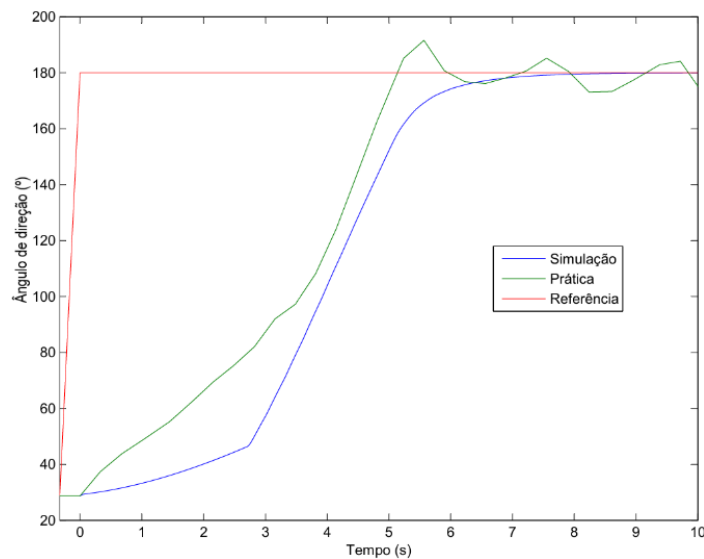


Figura 6.16 – Variação do ângulo de direção, em função do tempo, do barco autónomo com o controlo de direção.

Na reação da direção, observa-se que nos primeiros 4 segundos o barco realiza um movimento para trás, pelo facto de a velocidade de rotação do barco ser menor. Este efeito sugere que a velocidade linear inicial aumenta o tempo de rotação para trás, por causa de perturbar a velocidade de rotação do barco, como dado pelo modelo.

Nos primeiros 4 segundos, a reação da simulação não acompanha a reação prática. Uma possível explicação será a dificuldade em amostrar de

maneira precisa as constantes do movimento de rotação para trás, tendo em conta a sua ordem de grandeza. No restante tempo, a reação da simulação acompanha a reação prática, tendendo para o valor de referência.

Para observar a reação do barco somente com o movimento de rotação para a frente, testa-se a reação do barco para uma mudança de direção de 90° . Coloca-se o barco a percorrer uma trajetória de N pontos e regista-se a transição de quando o barco chega ao ponto de rotação de 90° , tendo o barco uma velocidade linear de 0,5 m/s. Na Figura 6.17 observa-se a variação da direção ao longo do tempo para essas condições, com os resultados de simulação e experimentais.

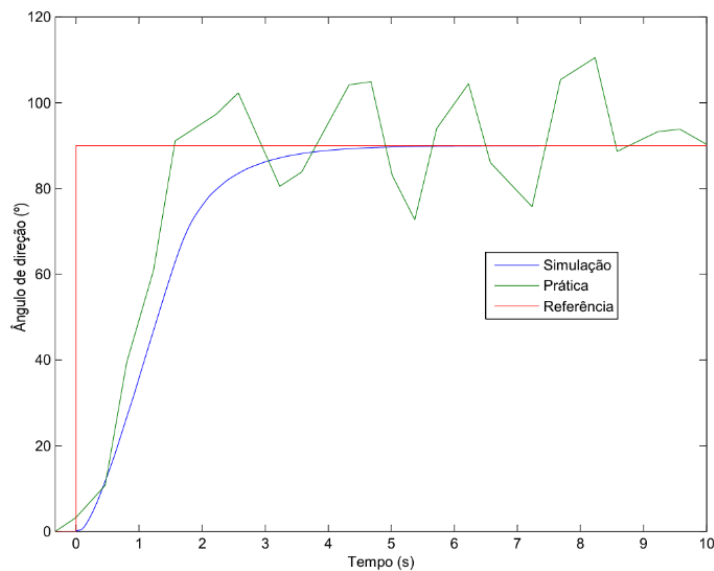


Figura 6.17 – Variação do ângulo de direção, em função do tempo, do barco autônomo com o controlo de direção.

Na reação prática do ângulo, observa-se que o tempo de subida é cerca de 1,5 segundos, dentro da ordem dos 2 segundos projetados no controlo. Quando não há movimento de rotação para trás, a reação da simulação é semelhante à reação prática.

Estes resultados validam o modelo deduzido na rotação para a frente. Sendo que as diferenças observadas entre os resultados de simulação e os resultados práticos devem-se à variação de parâmetros e às perturbações no sistema. Observa-se que em todos os casos, o sistema de controlo corrige o ângulo da direção do barco para o valor de referência.

Tendo o sistema de controlo de direção validado, procede-se para a validação do controlo de posição através do teste de seguimento da trajetória pré-definida pelo barco autónomo.

6.9 Teste de seguimento da trajetória pré-definida pelo barco autónomo

Nesta secção apresentam-se os testes para verificar o seguimento de um percurso pré-definido. Testa-se o percurso com duas coordenadas pré-definidas, num lago. Os dois pontos pré-definidos têm as seguintes coordenadas GPS: Ponto A: Latitude: 32,645512; Longitude: -16,914786. Ponto B: Latitude: 32,645189; Longitude: -16,914811. A trajetória do barco foi gravada com uma câmara cuja hiperligação do vídeo está referenciado na secção D.3 do Apêndice D. O registo dos pontos GPS, percorridos pelo barco autónomo, permitiu representar o percurso realizado com os locais apresentados na Figura 6.18. Sendo os pontos vermelhos os pontos pré-definidos da trajetória, os pontos azuis o registo da posição atual do barco em cada instante de tempo e a linha preta a trajetória de referência. O percurso tem uma distância entre pontos de aproximadamente 20 metros.



Figura 6.18 – Definição e trajeto de um percurso do barco autónomo entre dois pontos de referência.

Observa-se que o barco avança para a próxima a posição de referência antes de chegar ao destino, pelo facto de não ter pontos azuis em cima dos

pontos vermelhos, isto é devido ao facto de se adicionar uma margem de 10 metros de distância, considerando que se chegou ao local quando se vai abaixo dessa margem. Essa condição de paragem é descrita pela equação (2.9) do Capítulo 2, na secção 2.9, e é apresentada no código do Apêndice C, na secção C.3 (Figura C.15). Visualiza-se que o barco tem um desvio lateral de cerca de 3 metros, no máximo, dentro da exatidão e precisão do sensor de posição de 3 metros.

Calcula-se também a velocidade linear média do barco ao realizar o somatório da diferença de posição, distância entre pontos, a dividir pelo somatório da diferença entre tempos em N amostras. Tendo em conta a distância considera-se desprezável a diminuição da velocidade média nas transições utilizando, assim, todas as amostras. Para tal utiliza-se a equação (6.4), que faz o somatório do cálculo da distância entre dois pontos seguidos e divide pelo seu intervalo de tempo, obtendo uma velocidade média de 0,7 m/s [36]. Sendo j o índice do ponto, φ_j a latitude do local atual, φ_{j+1} a latitude do local seguinte, $\Delta\varphi$ a diferença entre as latitudes do local atual e do local seguinte, $\Delta\lambda$ a diferença entre as longitudes do local atual e do local seguinte, r o raio da Terra, Δt o intervalo de tempo e $\sim v$ a velocidade média.

$$\sim v = \frac{\sum_j^N 2r \sin^{-1} \left(\sqrt{\frac{1 - \cos(\Delta\varphi) + \cos(\varphi_j) \cos(\varphi_{j+1}) (1 - \cos(\Delta\lambda))}{2}} \right)}{\sum_j^N \Delta t} = \frac{93,5}{129,3} \cong 0,7 \text{ m/s} \quad (6.4)$$

Através do comando do barco autónomo foram enviados os pontos de uma nova trajetória ao barco. A trajetória com os dois pontos de referência, assinalados na Figura 6.18, foram substituídos por novos pontos com coordenadas GPS: Ponto A: Latitude: 32,645189; Longitude: -16,914811. Ponto B: Latitude: 32,645269; Longitude: -16,914545. O vídeo com a hiperligação indicada na secção D.3 do Apêndice D mostra o percurso do barco autónomo para esses dois novos pontos que definem o novo percurso. O registo dos pontos GPS, percorridos pelo barco autónomo, permitiu representar o percurso realizado com os locais apresentados na Figura 6.19. Sendo os pontos vermelhos os pontos pré-definidos da trajetória, os pontos azuis o registo da posição atual do barco em cada instante de tempo e a linha preta a trajetória de

referência. Observa-se que o barco muda de trajetória sem problemas, tendo uma distância entre pontos de cerca de 15 metros e começando a navegar em direção do ponto B.



Figura 6.19 – Definição e trajeto de um percurso novo do barco autônomo pela transmissão do comando de nova trajetória com dois pontos de referência.

Para testar o barco autônomo no mar foi definida uma trajetória pré-definida com cinco pontos programada no microcontrolador do barco. O barco foi programado com os 5 pontos com as coordenadas GPS: Latitude1=32,644942; Longitude1=-16,912536; Latitude2=32,644905; Longitude2=-16,912660; Latitude3=32,644684; Longitude3=-16,912638; Latitude4=32,644612; Longitude4=-16,912885; Latitude5=32,644402 e Longitude5=-16,912980. A Figura 6.20 mostra a localização dos pontos e o percurso pré-definido.

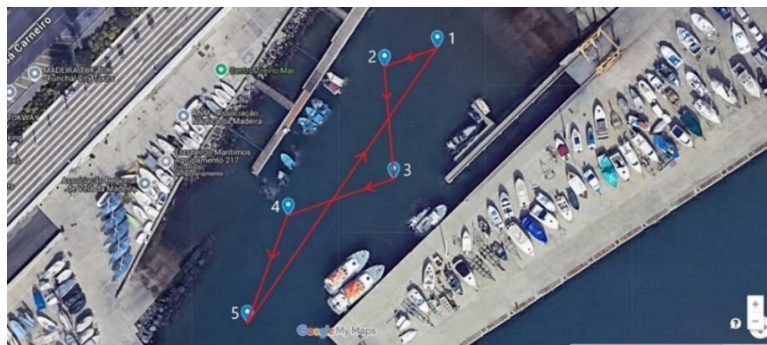


Figura 6.20 – Percurso pré-definido para o barco autônomo com cinco pontos.

Os resultados experimentais mostram que o barco segue a trajetória programada observando as mudanças de direção, sempre que o barco

Capítulo 6 – Implementação e testes do barco autónomo

autónomo chega a um ponto de referência muda para o seguinte ponto de referência como mostra a Figura 6.21 e o vídeo com a hiperligação indicada na secção D.4 do Apêndice D. Sendo os pontos vermelhos os pontos pré-definidos da trajetória, os pontos azuis o registo da posição atual do barco em cada instante de tempo e a linha preta a trajetória de referência.

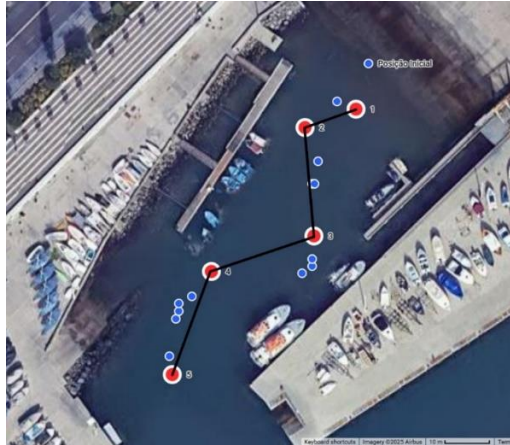


Figura 6.21 – Registo do barco autónomo ao passar pelo percurso pré-definido com 5 pontos.

Observa-se que o desvio lateral aumenta para cerca de 5 metros. Este aumento provém da adição de perturbações causadas no barco devido às ondas do mar.

Tendo o sistema de controlo de posição validado através do teste de seguimento de trajetória, procede-se para o teste de alcance das comunicações do barco.

6.10 Teste de alcance

Nesta secção são feitos testes de alcance máximo, em linha de vista, da comunicação entre o telecomando e o barco com o rádio nRF905. Na realização deste teste variou-se a posição do telecomando e do barco até perder a comunicação. É, também, de notar que quando o barco perde a ligação ele para o processamento de seguimento de trajetória até retomar a comunicação.

Primeiramente, colocou-se o barco na posição de coordenadas GPS: Latitude: 32,64422; Longitude: -16,91308. De seguida, deslocou-se o telecomando, enviado um comando de reação do motor e verificando se o barco reagia. A posição do comando foi-se afastando sem perder a linha de vista. Verificou-se que na posição mais afastada, antes de perder a linha de vista, nas

Capítulo 6 – Implementação e testes do barco autónomo

coordenadas GPS: Latitude: 32,64583; Longitude: -16,91193; continuava a comunicar, como mostra a Figura 6.22 com a localização desses pontos, obtendo uma distância de cerca de 200 metros. Não conseguindo afastar mais o telecomando sem perder a linha de vista, mudou-se de localização.



Figura 6.22 – Representação geográfica da localização do telecomando e do barco autónomo com comunicação em linha de vista.

Para descobrir o alcance máximo, foi continuamente aumentada a distância entre o barco autónomo e a posição do telecomando. Verifica-se que o telecomando na posição com as coordenadas GPS: Latitude: 32,64198; Longitude: -16,91608; e o barco autónomo na posição com as coordenadas GPS: Latitude: 32,64353; Longitude: -16,91371; corresponde à distância máxima de cerca de 300 metros da qual é possível estabelecer comunicação em linha de vista entre o comando e o barco autónomo. A Figura 6.23 mostra a localização dos pontos que define a distância de maior alcance de comunicação.

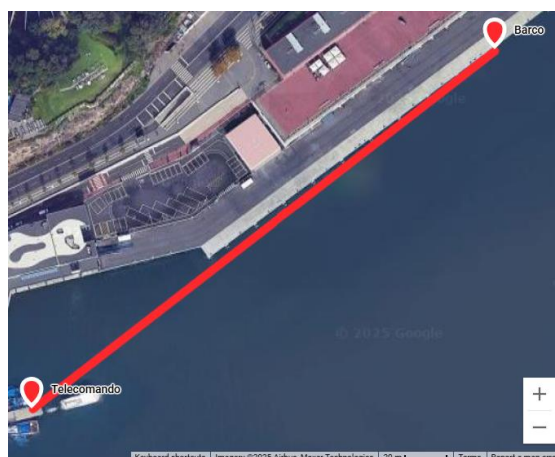


Figura 6.23 – Representação geográfica com a localização do barco autónomo e do telecomando até onde é possível estabelecer comunicação em linha de vista.

Os resultados experimentais mostram que para uma distância inferior de 300 metros, em linha de vista, o telecomando consegue estabelecer a comunicação com o barco, e para uma distância superior a 300 metros, em linha de vista, o telecomando pode não conseguir estabelecer a comunicação com o barco.

Os testes experimentais também indicam que caso se perca a linha de vista, para uma distância superior a 20 metros, perde-se a comunicação.

6.11 Considerações finais sobre a implementação e testes do barco autônomo

Em resumo, para implementar o sistema do barco autônomo realizou-se a escolha dos componentes do barco autônomo: a estrutura do barco com os respetivos motores e *drivers* [38] foi escolhida com o objetivo de reutilizar o sistema [35] já montado; os sensores para obtenção da posição e direção do barco, escolhendo o sensor GPS [39] e o sensor 9DOF [40], respetivamente, por terem baixo custo e exatidão e precisão de 3 metros e 10°; o microcontrolador e rádio, escolhendo o Raspberry Pi Pico [43] e o nRF905 [22], respetivamente, por ter bibliotecas disponíveis e interação em linguagem de alto nível e utilizando um rádio de rádio frequência a 433 MHz.

Implementou-se o sistema através da construção de placas de circuito impresso para interligar os diversos componentes com a interação de cada componente com o microcontrolador, utilizando os protocolos de comunicação SPI [45], UART [45] e I2C [47]. Realizaram-se testes de comunicação, testes de mudança de direção e testes de seguimento da trajetória, validando com esses testes o funcionamento do sistema e obtendo resultados similares à simulação. A reação da direção do barco para uma diferença de 180° tem um tempo de subida, de 10% a 90%, de cerca de 3 segundos e para uma diferença de 90° tem um tempo de subida de cerca de 1,5 segundos. Observou-se que o desvio lateral do barco aumentou para 5 metros, devido à precisão do GPS e perturbação das ondas no barco. Tendo o sistema implementado e testado, procede-se com o realçar das principais conclusões deste trabalho.

Capítulo 7

CONCLUSÕES

Neste capítulo apresenta-se um sumário do trabalho realizado, realçando as conclusões mais importantes com base nos resultados dos diversos testes e indicam-se possíveis melhorias e trabalhos de investigação futuros sobre o tema do barco autónomo.

7.1 Conclusões gerais

Nesta dissertação de mestrado criou-se um sistema de navegação autónoma de um barco.

Com vista a atingir os objetivos propostos, começou-se por realizar uma revisão da bibliografia sobre os diversos componentes do barco autónomo. Observou-se que os barcos mais adequados para a implementação deste tipo de sistema são os barcos propulsionados por motores elétricos, devido ao facto de propulsionarem o barco diretamente através de um sistema de acionamento em comunicação com o processador, que realiza o cálculo.

As principais técnicas de navegação para o barco autónomo requerem a amostragem das grandezas relacionadas com a direção e a posição do barco. Através de um sistema de controlo em malha fechada, é possível fazer com que o barco siga o trajeto desejado. No âmbito deste trabalho, a técnica de navegação autónoma escolhida consistiu no cálculo da direção pretendida, com base na comparação entre a posição atual e a posição desejada e na utilização de um sistema de controlo linear para orientar o barco na trajetória correta. Escolheu-se esta abordagem pela sua maior precisão em relação aos sistemas de controlo baseados em histerese.

Os sensores utilizados para a implementação de um sistema de navegação autónomo foram o sensor 9DOF, para a medição da direção, e o sensor GPS, para a determinação da posição. Estes sensores destacam-se pela sua elevada disponibilidade, precisão e exatidão na obtenção de dados GPS em ambiente marítimo, através da comunicação com satélites, bem como pela alta precisão e fiabilidade resultante da combinação dos componentes do sensor 9DOF, nomeadamente o magnetómetro, o acelerómetro e o giroscópio.

Os motores utilizados para este tipo de trabalho, em pequenas embarcações, são motores DC em conjunto com um sistema de acionamento em ponte-H. Esta solução foi escolhida pelo facto do conjunto motor DC e ponte-H requerer um número reduzido de componentes.

Verificou-se que os sistemas de comunicação mais adequados para este tipo de trabalho são os de baixa radiofrequência, uma vez que permitem alcançar a maior distância possível, mesmo com uma baixa taxa de transmissão de

Capítulo 7 – Conclusões

dados. Adicionalmente, identificaram-se variáveis relevantes para a monitorização no ambiente marítimo, como a temperatura, a condutividade, o valor de pH e os níveis de oxigenação, que possibilitam a análise da qualidade da água.

Neste projeto definiu-se um sistema de barco autónomo, utilizando um telecomando para comunicar com o barco, transmitindo as funcionalidades do barco que foram especificadas: controlo manual do barco, controlo autónomo do barco e transmissão de uma nova trajetória. Para a implementação do sistema de atuação, foram utilizados uma bateria, um microcontrolador, os motores DC e a ponte-H, sendo o controlo realizado através de um sinal PWM gerado pelo microcontrolador.

No telecomando é utilizado como fonte de energia o computador e um microcontrolador interpreta os periféricos de entrada e envia o comando corresponde pelo rádio, para estabelecer a comunicação entre o utilizador e barco.

No barco é utilizada como fonte de energia uma bateria e é usado um microcontrolador que interpreta as mensagens recebidas do rádio, realiza a leitura dos sensores de direção e posição, e envia um sinal para o sistema de acionamento atuar nos motores do barco.

Esse microcontrolador é responsável por implementar o sistema de controlo da posição do barco autónomo, determinando a direção desejada com base na amostragem da posição atual. A direção desejada é comparada com a direção atual através de um sistema de controlo, que calcula a velocidade de rotação de referência do barco. Em seguida, a velocidade de rotação de referência é comparada com a velocidade de rotação atual, obtendo-se um sinal diferencial entre os motores. Com base na amostragem do erro de direção, é calculado um sinal comum para ambos os motores. O sinal diferencial é então somado ao sinal comum no motor direito e subtraído no motor esquerdo, resultando na obtenção do *duty-cycle* do sinal PWM a ser aplicado a cada motor.

No controlo manual realiza-se a leitura dos periféricos de entrada correspondentes à velocidade de cada motor no telecomando e transmite-se a mensagem com esses valores. No barco é interpretada a mensagem e é enviado

Capítulo 7 – Conclusões

o sinal PWM aos motores correspondentes. No controlo autónomo realiza-se a leitura do periférico de entrada correspondente no telecomando e transmite-se a mensagem para o barco, que ao interpretar a mensagem executa o código do sistema de controlo da posição.

Na transmissão da nova trajetória, o telecomando converte o conjunto de coordenadas dos locais em *bytes* para a mensagem e o barco converte o conjunto de *bytes* da mensagem nas coordenadas dos locais.

O modelo do barco autónomo deduzido partiu da junção de 4 submodelos: o modelo do motor DC, o modelo de Yoerger, o equivalente das forças e o modelo hidrodinâmico do barco.

O modelo do motor DC descreve como é que a partir de uma tensão equivalente aplicada aos terminais de um motor DC, se gera um binário motor, em cada motor. O modelo de Yoerger descreve como é que a partir de um binário motor se gera uma força equivalente na turbina, em cada turbina. O equivalente das forças descreve como a partir de duas forças geradas pelas turbinas se obtém uma força equivalente diferencial, perpendicular à direção do barco e localizada no centro de massa, e uma força equivalente comum, direcionada na direção do barco e localizada no centro de massa.

O modelo hidrodinâmico do barco descreve como a partir de uma força equivalente diferencial se gera a velocidade de rotação do barco, que ao integrar se obtém a direção, e como a partir de uma força equivalente comum se gera a velocidade linear do barco, que ao integrar, tendo em conta o ângulo, se obtém a direção. Dê referir que o modelo resultante é não linear.

Simplificou-se o modelo para um sistema de primeira ordem, no ponto de funcionamento onde a velocidade inicial é nula, para facilitar o projeto dos controladores. Projetam-se o controlador da velocidade de rotação do barco, com um compensador proporcional-integral, para a velocidade de rotação do barco acompanhar a velocidade de rotação de referência com um tempo de resposta adequado às características do modelo do barco. Realizou-se o cancelamento polo-zero para obter uma das equações dos parâmetros desse controlador e para obter a outra projetou-se um tempo de subida de 0,6 segundos.

Capítulo 7 – Conclusões

Projetou-se o controlador de direção do barco, com um compensador proporcional-derivativo, de modo a que o barco acompanhe a direção de referência para seguir a trajetória pré-definida. Realizou-se o cancelamento polo-zero para obter uma equação dos parâmetros desse controlador e para obter a outra equação projetou-se um tempo de subida de 2 segundos.

A velocidade linear do barco foi ajustada em função do erro de direção através de um sistema de comando. Para o barco autónomo percorrer uma trajetória, amostrou-se a posição atual e comparou-se com a posição pretendida para calcular o ângulo da direção de referência do barco.

Utilizando o programa SIMULINK/MATLAB construiu-se o diagrama de blocos equivalente juntando os controladores projetados com o modelo físico deduzido. Adicionaram-se os limites físicos e atrasos no sistema de controlo e para simular utilizaram-se os parâmetros que caracterizam o barco e os controladores (Como a massa do barco, a constante de atrito de velocidade de rotação e a velocidade linear do barco, as constantes do equivalente das forças, o ganho da velocidade de rotação do motor, a constante de atrito aquático, o ganho do binário motor, o ganho da corrente, a resistência e a indutância equivalente do motor, a constante da força eletromotriz dos motores do barco e as constantes dos controladores).

Simulou-se a variação do ângulo ao longo do tempo, a variação da velocidade linear ao longo do tempo, a variação da velocidade de rotação e a sua referência ao longo do tempo para uma direção de referência de 180° , com uma direção inicial a 0° . Observou-se que o sistema tendia para o valor pretendido tendo um tempo de subida de 10% a 90% do valor do ângulo de 3 segundos, por ser limitado fisicamente, e um tempo de subida na velocidade de rotação de 0,6 segundos.

Simulou-se o sistema também para variações do ângulo e da velocidade linear inicial, observando que para uma direção de referência de 90° o tempo de subida era de cerca de 2 segundos. Simulou-se a variação da posição ao longo do tempo, verificando que a regra implementada no comando de velocidade linear, de escalar em função do erro de direção, diminui o desvio lateral do barco, ao realizar uma rotação de 180° , o barco afasta-se cerca de 1 metro perpendicularmente da trajetória pretendida. Observa-se que o sistema para

Capítulo 7 – Conclusões

uma velocidade linear inicial de 0,5 m/s demora mais tempo, cerca de 3 segundos em comparação a 1 segundo com velocidade linear inicial a 0 m/s, a realizar a rotação para trás. Isto é devido ao efeito da perturbação da velocidade linear na reação da velocidade de rotação do barco observada no modelo do barco.

Também se simulou o efeito das ondas e do ruído dos sensores no sistema ao adicionar blocos que geram ruído branco gaussiano. Adicionaram-se esses blocos com a velocidade de rotação do barco, para simular a perturbação gerada por uma onda no barco, e somou-se o ruído com a amostra do ângulo de direção do barco, para simular o ruído do sensor. Observou-se que mesmo com essas perturbações, o sistema reagiu e tendeu para o valor pretendido, com variações de cerca de 10° no estado estacionário.

Na implementação prática do sistema utilizou-se um microcontrolador, programado em *Python*, para realizar o processamento dos comandos e o controlo, demorando cerca de 0,3 segundos para completar um ciclo do controlo da direção. Utilizou-se uma frequência de comutação de 1 kHz no sinal PWM. Utilizou-se um sensor de posição GPS com uma precisão de 3 metros e um sensor de direção 9DOF com uma precisão de 10°. Utilizou-se um rádio com um alcance testado de cerca de 300 metros, que permite a comunicação entre o telecomando e o barco. Juntando estes componentes com as suas diversas características realizaram-se testes para obter os parâmetros do sistema utilizados para definir o modelo do barco.

O sistema do barco foi construído a partir da junção da bateria com os sensores, com o microcontrolador, com o rádio e com o circuito de acionamento dos motores. Com o sistema do barco a funcionar, observou-se que ele reagia ao comando manual transmitido pelo telecomando, reagia a uma mudança de direção no controlo autónomo, realizava o seguimento da trajetória e mudava de percurso ao receber o comando de nova trajetória.

O tempo de subida da reação a uma mudança de direção de 180° foi de cerca de 3 segundos, igual ao obtido na simulação. Para uma mudança de direção de 150° com velocidade linear inicial de 0,5 m/s, observou-se que a rotação para trás demorava mais tempo, cerca de 3 segundos. Contudo, a reação da simulação em comparação com a prática é distinta na rotação para trás,

Capítulo 7 – Conclusões

apresentado uma derivada diferente. Realizou-se um teste de mudança de direção de 90° com velocidade linear inicial de 0,5 m/s, visualizando um tempo de subida da reação de cerca de 1,5 segundos, perto do obtido pela simulação, podendo a diferença ser explicada por perturbações no sistema. Estes testes de reação do sistema validam o modelo deduzido para a rotação dos motores para a frente.

Observou-se que a velocidade média do barco era de 0,7 m/s, perto do valor pretendido e simulado, o que também valida o modelo deduzido. No seguimento da trajetória, verificou-se que o barco acompanhava a trajetória, para N pontos, com um desvio lateral de cerca de 5 metros, devido às perturbações e precisão do GPS. Esta implementação prática permitiu a validação do sistema como uma solução para o problema de navegação do barco autónomo.

Conclui-se que todos os objetivos projetados para esta dissertação de mestrado foram cumpridos. A revisão bibliográfica permitiu reunir a informação necessária para o desenvolvimento do sistema. Com base nas especificações, foi possível direcionar o problema para uma proposta de solução. A criação do modelo possibilitou o projeto dos controladores, que, por sua vez, permitiram a realização da simulação e a posterior implementação prática do sistema. Através da simulação, foram obtidas métricas quantitativas que possibilitaram a comparação com os resultados práticos e a formulação de conclusões sobre o desempenho do sistema. Através da implementação e dos testes práticos, foi desenvolvida uma solução para o problema de navegação do barco autónomo, validando o modelo na rotação dos motores para a frente e obtendo resultados consistentes com os da simulação. Na rotação dos motores para trás, obteve-se resultados diferentes possivelmente do facto da precisão dos dinamómetros ser muito pequena para a ordem de grandeza medida, o que forneceu parâmetros errados na rotação para trás.

7.2 Propostas para trabalhos futuros

Ao longo do desenvolvimento desta dissertação foram surgindo novos temas sobre o funcionamento do sistema de navegação autónomo, que podem ser investigados para melhorar o sistema desenvolvidos neste trabalho. Os tópicos que se propõe serem desenvolvidos são:

- 1) Simplificação do modelo deduzido do barco. O modelo, também, pode ser generalizado para N motores em diversas direções;
- 2) Desenhar um algoritmo de arranque e atracagem do barco de maneira autónoma. Tal implica um controlo do barco mais preciso, sendo necessário adicionar novos sensores para dar maior precisão à posição do barco;
- 3) Desenvolver um sistema de deteção e desvio de obstáculos para quando o barco estiver a fazer um percurso pré-definido. Podem ser usados sensores de imagens dos arredores de maneira a detetar obstáculos que se possam encontrar ao percorrer a trajetória e atuar no sistema de controlo. Para processamento de imagem pode-se utilizar inteligência artificial;
- 4) Aumentar a autonomia energética do sistema para trajetórias de longo alcance. Ao adicionar um sistema de captação de energia no barco, como por exemplo um painel fotovoltaico, consegue-se captar energia solar para carregar a bateria do barco autónomo;
- 5) Reforçar o sistema de comunicação para permitir comunicações a maior distância. A utilização de um protocolo de comunicação a longa distância, como por exemplo o LoRa, pode melhorar o alcance das comunicações. A possibilidade de adicionar comunicação por satélite, pode permitir a comunicação com o barco a longas distâncias;
- 6) Implementação de uma rede de sensores marítima. Ao ter diversas embarcações a recolher e transmitir dados será possível cobrir uma maior área de monitorização.

REFERÊNCIAS

Referências

- [1] S. McGrail, *Early ships and seafaring : European water transport*, Pen & Sword Archaeology, 2014.
- [2] European Commission, “Recreational Craft,” [Online]. Available: https://single-market-economy.ec.europa.eu/single-market/european-standards/harmonised-standards/recreational-craft_en. [Acedido em 28 2 2025].
- [3] A. A. d. M. Pereira, “Navigation and guidance of an autonomous surface vehicle,” University of Southern California, 2007.
- [4] J. M. G. B. d. Sousa, “Barco Autónomo para Monitorização Ambiental,” Universidade de Coimbra, 2012.
- [5] “Yamaha Motor M12,” Yamaha, 2020. [Online]. Available: <https://www.yamaha-motor.eu/ie/en/marine-engines/electric-drive/pdp/m12-2020/>. [Acedido em 24 05 2025].
- [6] “YAM 240T,” Yamaha, 2025. [Online]. Available: <https://www.yamaha-motor.eu/pt/pt/boats/inflatables/pdp/yam-240t-2020/#2020-YAM240T-NA>. [Acedido em 09 06 2025].
- [7] “PC60,” Horizon, 2025. [Online]. Available: <https://www.horizonyachtusa.com/models/pc/pc60>. [Acedido em 09 06 2025].
- [8] A. Reichelt-Brushett, *Marine Pollution – Monitoring, Management and Mitigation*, Springer-Verlag, 2023.
- [9] C. D. Ahrens e R. Henson, *Essentials of Meteorology An Invitation to the Atmosphere*, Cengage Learning, 2018.
- [10] P. Yang, L. Dong e W. Xu, “Small Maritime Target Detection Using Gradient Vector Field Characterization of Infrared Image,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 1827 - 1841, 2023.
- [11] J. Fraden, *Handbook of Modern Sensors, Physics, Designs, and Applications*, Springer-Verlag, 2004.
- [12] A. El-Rabbany, *Introduction to GPS: The Global Positioning System*, Artech House Publishers, 2006.
- [13] J. Stevens, J. M. Smith e R. A. Bianchetti, “Global Positioning Systems,” Pennsylvania State University, 2012. [Online]. Available: <https://www.e-education.psu.edu/geog160/node/1923>. [Acedido em 23 05 2025].
- [14] J. Nakamura, *Image sensors and signal processing for digital still cameras*, Taylor & Francis, 2006.
- [15] H. Taub e D. Schilling, *Principles of Communication Systems*, McGraw-Hill, 1986.
- [16] D. Evans e G. Jessop, *VHF-UHF Manual*, Radio Society of Great Britain, 1979.
- [17] G. Maral, M. Bousquet e Z. Sun, *Satellite Communications Systems: Systems, Techniques and Technology*, Wiley, 2009.
- [18] International Telecommunication Union, “Maritime related recommendations,” 2025. [Online]. Available: <https://www.itu.int/en/ITU-R/terrestrial/mars/Pages/References.aspx>. [Acedido em 06 06 2025].

Referências

- [19] “Xbee Sub-1 GHz RF Modules,” Digi, 2025. [Online]. Available: <https://www.digi.com/products/embedded-systems/digi-xbee/rf-modules/sub-1-ghz-rf-modules/>. [Acedido em 28 05 2025].
- [20] “Lora SX1276,” Semtech, 2025. [Online]. Available: <https://www.semtech.com/products/wireless-rf/lora-connect/sx1276>. [Acedido em 28 05 2025].
- [21] “Adafruit RFM95W LoRa Radio,” Adafruit, 2025. [Online]. Available: <https://www.adafruit.com/product/3072>. [Acedido em 06 06 2025].
- [22] “nRF905,” Nordic Semiconductor, 2008. [Online]. Available: https://docs.nordicsemi.com/bundle/nRF9-Series/resource/nRF905_PS_v1.5.pdf. [Acedido em 28 05 2025].
- [23] HiLetgo, “NRF905 Wireless Transceiver Module,” Amazon, 2025. [Online]. Available: <https://www.amazon.com/HiLetgo-Wireless-Transceiver-PTR8000-Antenna/dp/B00RJISY5Y>. [Acedido em 06 06 2025].
- [24] McGraw-Hill, “Engine,” em *Concise Encyclopedia of Science and Technology, Fifth Edition*, McGraw-Hill, 1987, p. 822.
- [25] P. A. Tipler, Física para Cientistas e Engenheiros: Mecânica, Oscilações e Ondas, Termodinâmica, LTC-Livros Técnicos e Científicos, 2002.
- [26] A. E. Fitzgerald, C. Kingsley e S. D. Umans, Electrical Machinery, McGraw Hill, 2002.
- [27] D. Yoerger, J. Cooke e J.-J. Slotine, “The influence of thruster dynamics on underwater vehicle behavior and their incorporation into control system design,” *IEEE Journal of Oceanic Engineering*, vol. 15, pp. 167-178, 1990.
- [28] L. Guzzella e C. H. Onder, Introduction to Modeling and Control of Internal Combustion Engine Systems, Springer-Verlag, 2004.
- [29] R. K. Wangsness, Electromagnetic Fields, Wiley, 1986.
- [30] R. Krishnan, Electric Motor Drives: Modeling, Analysis, and Control, Prentice Hall, 2001.
- [31] “Marine Honda BF2.3,” Honda, 2025. [Online]. Available: https://marine.honda.com/outboards/models/portable/BF2_3. [Acedido em 24 05 2025].
- [32] K. Ogata, Modern Control Engineering, Prentice Hall, 2002.
- [33] C. Ye, “USV Charging Based on WPT System,” Instituto Universitário de Lisboa, 2020.
- [34] D. S. d. Santos, C. L. Nascimento e W. C. Cunha, “Autonomous Navigation of a Small Boat Using IMU/GPS/Digital Compass Integration,” em *IEEE International Systems Conference (SysCon)*, Orlando, 2013.
- [35] J. P. Snyder, Map Projections - A Working Manual, U.S. Government Printing Office, 1987.
- [36] G. v. Brummelen, Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry, Princeton University Press, 2013.
- [37] “Amazon,” Bilbear, 24 03 2022. [Online]. Available: <https://www.amazon.de/-/en/Bilbear-Feeding-Fishing-Replacement-Batteries/dp/B09WDNL618/>. [Acedido em 11 06 2023].

Referências

- [38] “Datasheet4U,” Wuxi Microelectronics, [Online]. Available: <https://datasheet4u.com/pdf-down/R/Z/7/RZ7886-WuxiMicroelectronics.pdf>. [Acedido em 05 07 2025].
- [39] “GY-GPSV3 NEO-M8N,” Pixhawk, 28 5 2025. [Online]. Available: <https://www.amazon.com/GY-GPSV3-NEO-M8N-Controller-Pixhawk-1-65-3-6V/dp/B0FB477HQ8/>. [Acedido em 11 8 2025].
- [40] “Adafruit 9-DOF - LSM9DS1,” Adafruit, 25 3 2024. [Online]. Available: <https://www.adafruit.com/product/3387>. [Acedido em 11 08 2025].
- [41] J. Farrell, Aided Navigation: GPS with High Rate Sensors, McGraw-Hill, 2008.
- [42] P. Stoffregen, “MotionCal,” PJRC, 2016. [Online]. Available: <https://github.com/PaulStoffregen/MotionCal>. [Acedido em 31 07 2025].
- [43] “Raspberry Pi Pico,” Raspberry Pi, [Online]. Available: <https://www.raspberrypi.com/products/raspberry-pi-pico/>. [Acedido em 14 07 2025].
- [44] “Taoglas FW.43.B.SMA.M,” Mouser Electronics, [Online]. Available: <https://eu.mouser.com/ProductDetail/Taoglas/FW.43.B.SMA.M?qs=GmNq1OH0n5Qj1euafqqx0Q%3D%3D>. [Acedido em 14 07 2025].
- [45] D. S. Dawoud e P. Dawoud, Serial Communication Protocols and Standards, River Publishers, 2020.
- [46] Digi Key, “R-783.3-0.5,” Recom Power, [Online]. Available: <https://www.digikey.com/en/products/detail/recom-power/r-783-3-0-5/2256215>. [Acedido em 12 08 2025].
- [47] R. Hyde, The Book of I2C: A Guide for Adventurers, No Starch Press, 2022.
- [48] “3B Scientific Precision Dynamometer/Force Gauge - 5 N, Color Coded,” 3B Scientific Store, 11 3 2020. [Online]. Available: <https://www.amazon.es/dp/B007V60G7S/>. [Acedido em 26 10 2025].
- [49] “Tacómetro digital 0..99999RPM - Uni-T UT373,” Mauser, [Online]. Available: https://mauser.pt/catalog/product_info.php?products_id=096-6379. [Acedido em 26 10 2025].
- [50] “INA219 I2C Bi-Directional DC Current Power Supply Sensor Breakout Module,” HiLetgo, 24 7 2019. [Online]. Available: <https://www.amazon.com/HiLetgo-INA219-Bi-Directional-Current-Breakout/dp/B07VL8NY32/>. [Acedido em 26 10 2025].
- [51] Adafruit, “CircuitPython,” Adafruit, [Online]. Available: <https://docs.circuitpython.org/en/latest/README.html>. [Acedido em 16 10 2025].
- [52] A. Barela, “adafruit_gps,” Adafruit, [Online]. Available: https://github.com/adafruit/Adafruit_CircuitPython_GPS. [Acedido em 16 10 2025].
- [53] CircuitPython, “adafruit_lsm9ds1,” Adafruit, [Online]. Available: https://github.com/adafruit/Adafruit_CircuitPython_LSM9DS1. [Acedido em 16 10 2025].

Apêndice A

DEDUÇÃO E SIMPLIFICAÇÃO DAS EQUAÇÕES DO MODELO DO BARCO AUTÓNOMO

Neste apêndice são realizados os cálculos auxiliares para a dedução do modelo do barco autónomo e projeto dos controladores do barco autónomo.

A.1 Dedução das equações do modelo do barco

Começa-se com a dedução para obtenção da equação (4.12), para tal realizou-se a seguinte dedução.

$$T_{dif} - C_{\omega}\omega|\omega| = mv\omega \Rightarrow$$

Ignora-se por enquanto o sinal.

$$\Rightarrow T_{dif} - C_{\omega}\omega^2 = mv\omega \Leftrightarrow C_{\omega}\omega^2 + mv\omega - T_{dif} = 0 \Leftrightarrow$$

$$\Leftrightarrow \omega = \frac{-mv \pm \sqrt{(mv)^2 + 4T_{dif}C_{\omega}}}{2C_{\omega}} \Rightarrow$$

Quando $T_{dif} = 0 \Rightarrow \omega = 0$, ou seja, o termo dentro da raiz para $T_{dif} = 0$ tem de anular o termo de fora o que implica sinal positivo.

$$\Rightarrow \omega = \frac{-mv + \sqrt{(mv)^2 + 4T_{dif}C_{\omega}}}{2C_{\omega}} \Rightarrow$$

Tendo em conta agora a direção, a entrada T_{dif} toma o valor absoluto para não dar raízes de valor negativo e é adicionado o termo que indica a direção ao ler o sinal de T_{dif} .

$$\Rightarrow \omega(t) = \frac{-T_{dif}(t)}{|T_{dif}(t)|} * \frac{mv(t) - \sqrt{(mv(t))^2 + 4|T_{dif}(t)|C_{\omega}}}{2C_{\omega}}$$

Procede-se com a obtenção das funções de transferência do modelo simplificado de primeira ordem. Para obter a equação (4.15), realizou-se a seguinte dedução.

$$\left(\omega_{ref}(s) - \omega(s)\right) \frac{k_w}{s} \left(s + \frac{k_{wi}}{k_w}\right) * \frac{K}{1 + sT_{ds}} = \omega(s) \Leftrightarrow$$

$$\Leftrightarrow s\omega(s) + k_w \left(s + \frac{k_{wi}}{k_w}\right) * \frac{K}{1+sT_{ds}} \omega(s) = k_w \left(s + \frac{k_{wi}}{k_w}\right) \frac{K}{1+sT_{ds}} \omega_{ref}(s) \Leftrightarrow$$

$$\Leftrightarrow \omega(s) \left(s + k_w \left(s + \frac{k_{wi}}{k_w}\right) * \frac{K}{1+sT_{ds}}\right) = k_w \left(s + \frac{k_{wi}}{k_w}\right) \frac{K}{1+sT_{ds}} \omega_{ref}(s) \Leftrightarrow$$

Apêndice A - Dedução e simplificação das equações do modelo do barco autónomo

$$\Leftrightarrow \omega(s) = \frac{k_w \left(s + \frac{k_{wi}}{k_w} \right) \frac{K}{1 + sT_{ds}}}{s + k_w \left(s + \frac{k_{wi}}{k_w} \right) * \frac{K}{1 + sT_{ds}}} \omega_{ref}(s) \Leftrightarrow$$

$$\Leftrightarrow \omega(s) = \frac{k_w \left(s + \frac{k_{wi}}{k_w} \right) * \frac{\frac{K}{T_{ds}}}{s + \frac{1}{T_{ds}}}}{\frac{K}{s + k_w \left(s + \frac{k_{wi}}{k_w} \right) * \frac{T_{ds}}{s + \frac{1}{T_{ds}}}}} \omega_{ref}(s)$$

Para a obtenção da equação (4.21), realizou-se a seguinte dedução.

$$\theta(s) = \frac{1}{s} * \frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} \left(\theta_{ref}(s) - \theta(s) \right) * k_d \left(s + \frac{k_p}{k_d} \right) \Leftrightarrow$$

$$\Leftrightarrow s\theta(s) + \frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} k_d \left(s + \frac{k_p}{k_d} \right) \theta(s) = \frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} k_d \left(s + \frac{k_p}{k_d} \right) \theta_{ref}(s) \Leftrightarrow$$

$$\Leftrightarrow \theta(s) \left(s + \frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} k_d \left(s + \frac{k_p}{k_d} \right) \right) = \frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} k_d \left(s + \frac{k_p}{k_d} \right) \theta_{ref}(s) \Leftrightarrow$$

$$\Leftrightarrow \theta(s) = \frac{\frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} * k_d \left(s + \frac{k_p}{k_d} \right)}{s + \frac{k_w * \frac{K}{T_{ds}}}{s + k_w * \frac{K}{T_{ds}}} * k_d \left(s + \frac{k_p}{k_d} \right)} \theta_{ref}(s)$$

A.2 Determinação dos parâmetros dos controladores

Para calcular os parâmetros dos controladores utilizaram-se as equações (4.18) (4.19) (4.24) (4.25) com os valores de $K = 0,9$; $T_{ds} = 0,12$; t_r do controlador de velocidade de rotação do barco = 0,6 segundos; e t_r do controlador de direção do barco de 2 segundos. Para calcular os parâmetros dos controladores realizaram-se os seguintes cálculos.

$$k_w = \frac{2,2}{t_r * \frac{K}{T_{ds}}} = \frac{2,2}{0,6 * \frac{0,9}{0,12}} = \frac{2,2}{3,6} \cong 0,5 \text{ (rad/s)}$$

$$k_{wi} = \frac{k_w}{T_{ds}} = \frac{0,5}{0,12} \cong 4 \text{ (rad/s)}$$

$$k_d = \frac{2,2}{t_r * k_w * \frac{K}{T_{ds}}} = \frac{2,2}{2 * 0,5 * \frac{0,9}{0,12}} \cong 0,29 \text{ (rad)} = 0,29 * \frac{\pi}{180} \text{ (}^\circ\text{)} = 0,005 \text{ (}^\circ\text{)}$$

$$k_p = k_d * k_w * \frac{K}{T_{ds}} = 0,005 * 0,5 * \frac{0,9}{0,12} \cong 0,02 \text{ (}^\circ\text{)}$$

Apêndice B

TESTES PARA DETERMINAÇÃO DOS PARÂMETROS DO BARCO AUTÓNOMO

Neste apêndice são determinados os diversos parâmetros que caracterizam o modelo físico do barco, através da realização de testes de cada submodelo que compõe o modelo do barco. São realizados testes aos motores e testes da hidrodinâmica do barco.

B.1 Teste dos motores elétricos do barco autónomo

Primeiramente, mediu-se os parâmetros dos motores com um multímetro e obteve-se uma resistência da armadura de $R_1 = R_2 = 1,5$ ohm e uma indutância do enrolamento da armadura de $L_1 = L_2 = 480$ μ H.

Seguidamente, conectou-se o motor 1 a uma fonte com a tensão de 7,4 V e observou-se a reação do motor com um osciloscópio, como mostra a Figura B.1. Ligou-se o motor e observou-se a tensão ao longo do tempo, que depende da velocidade de rotação, como mostra a Figura B.2.

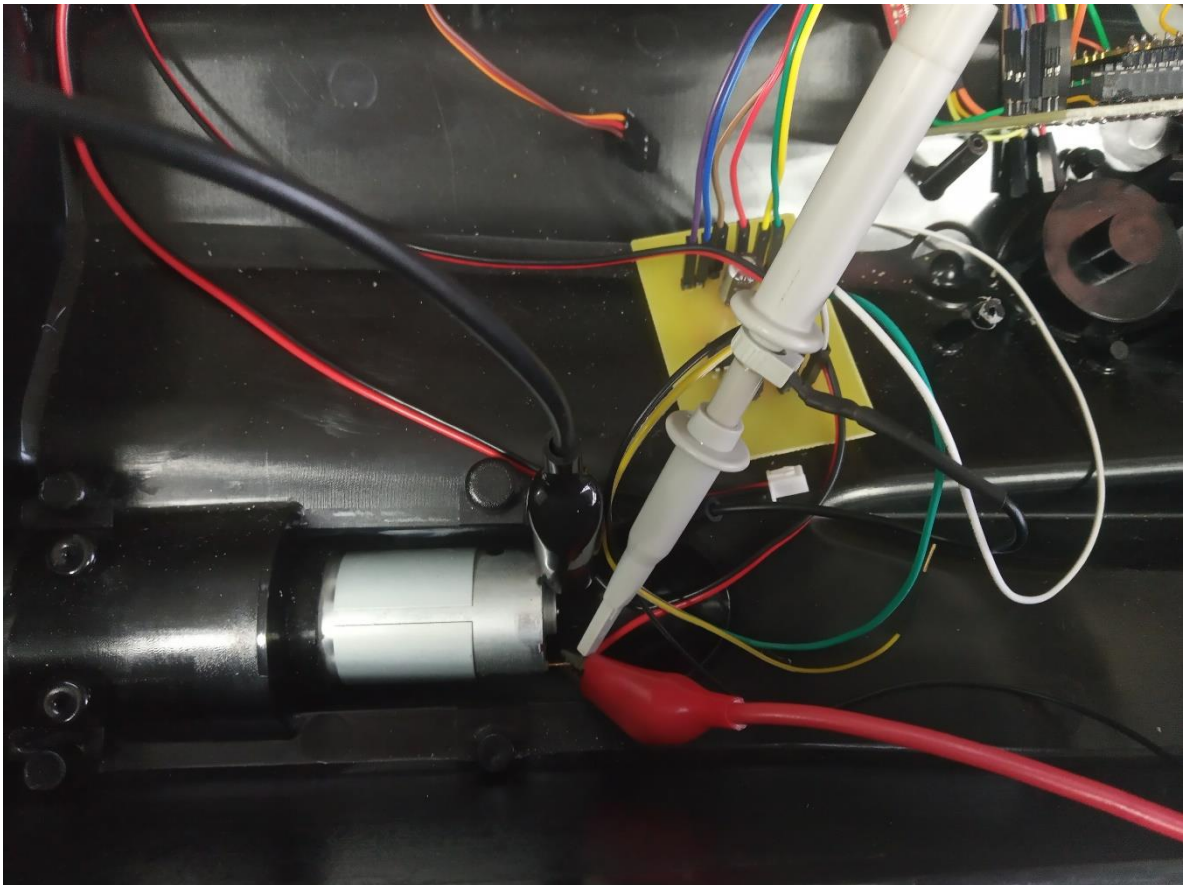


Figura B.1 – Montagem do circuito para testar o motor.

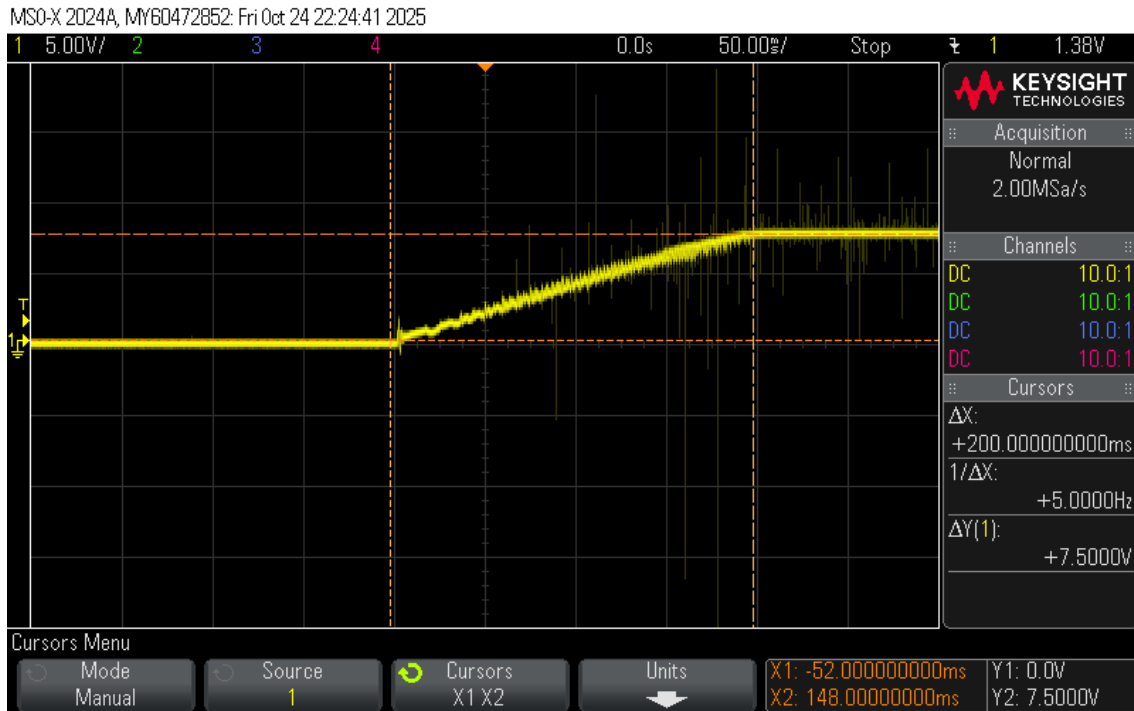


Figura B.2 – Variação da tensão ao longo do tempo ao ligar a fonte

Observa-se que o motor demora cerca de 200 milissegundos a reagir entre a velocidade de rotação máxima e a paragem. Na secção B.3, utiliza-se este valor para a determinação da constante de tempo no modelo de Yoerger.

B.2 Testes da hidrodinâmica do barco

Começou-se por medir a força equivalente máxima que cada motor aplica para propulsionar o barco através de um dinamómetro [48], com o objetivo de obter o equivalente das forças.

Ao colocar o dinamómetro diretamente em cima do motor, toda a força aplicada é contrariada pelo instrumento de medição, obtendo uma força total $T_2 = 2,1$ N com uma tensão de 7,4 V aplicada no motor direito como mostra a Figura B.3.



Figura B.3 – Teste da força equivalente gerada pelo motor.

Ao colocar o dinamômetro a meio do barco, toda a força em modo comum, T_{com} , aplicada pelo motor é contrariada pelo instrumento de medição, e ao

Apêndice B – Testes para determinação dos parâmetros do barco autónomo

bloquear a rotação do barco, obtêm-se a força comum aplicada pelo motor, $T_{com} = 1,6 \text{ N}$ quando aplicada uma tensão de 7,4 V, como mostra a Figura B.4.



Figura B.4 – Teste da força equivalente em modo comum gerada pelo motor direito.

Também se testou a força comum para trás causada pelos dois motores, que resultou em 0,2 N para -7,4 V aplicados a cada motor, ao colocar o dinamómetro na frente do barco, como mostra a Figura B.5.

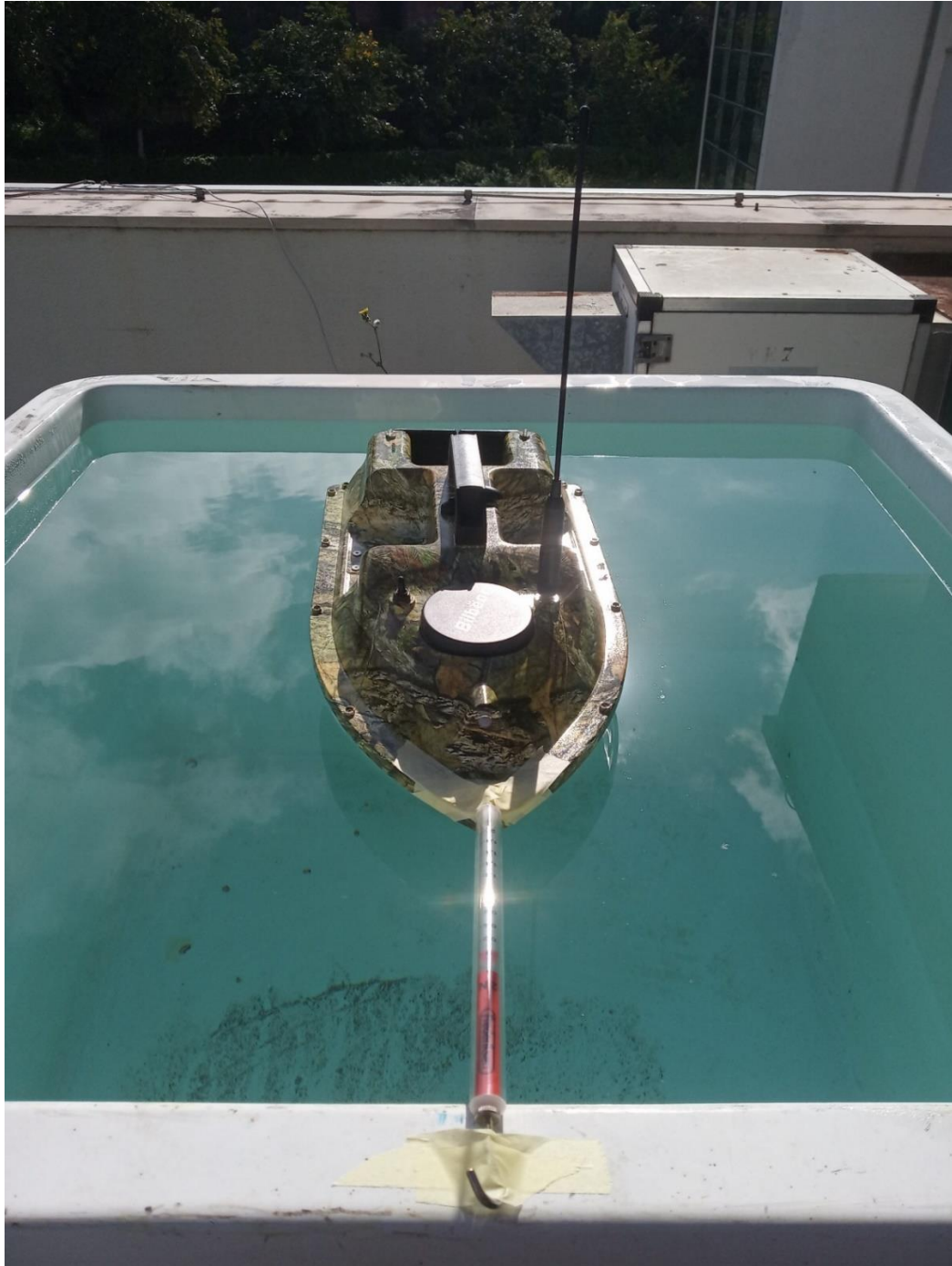


Figura B.5 – Teste da força equivalente para trás, gerada pelos motores.

A partir dos resultados estimou-se a força diferencial, T_{dif} , através do teorema de Pitágoras, tendo em conta que a força aplicada pelo motor direito nos testes realizados é decomposta numa força diferencial e numa força comum, como mostra a equação (B.1). De seguida, determinaram-se as seguintes variáveis do modelo apresentadas na Tabela B.1, através das equações (4.6) e (4.7) que calcularam o valor de C_1 e C_2 pelas equações (B.2) e (B.3), respetivamente.

Apêndice B – Testes para determinação dos parâmetros do barco autónomo

$$T_2^2 = T_{dif}^2 + T_{com}^2 \quad (B.1)$$

$$C_1 = \frac{T_{dif}}{T_1 - T_2} \quad (B.2)$$

$$C_2 = \frac{T_{com}}{T_1 + T_2} \quad (B.3)$$

Tabela B.1 – Variáveis obtidas do modelo para as equações das forças do barco autónomo.

Descrição da constante	Nomenclatura da constante	Valor da constante
Força do motor esquerdo para a tensão máxima	T_1	1,4 (N)
Força do motor direito para a tensão máxima	T_2	2,1 (N)
Força em modo comum do motor direito para a tensão máxima	T_{com}	1,6 (N)
Força em modo comum dos motores para trás para a tensão máxima	T_{com}	0,2 (N)
Força em modo diferencial do motor direito para a tensão máxima	T_{dif}	1,36 (N)
Constante equivalente da força diferencial	C_1	0,65
Constante equivalente da força comum	C_2	0,76

Conhecendo o valor das forças do barco autónomo, amostrou-se a velocidade de rotação e a velocidade linear. Deixando o barco a navegar com o motor direito, para a frente e para trás, amostrou-se o tempo, a distância percorrida e a variação do ângulo. Estas variáveis permitem calcular a velocidade de rotação do barco e a velocidade linear e com isso estimam-se os valores das constantes C_ω e C_v através das equações (4.10) e (4.11), com o cálculo apresentado na secção B.4. Na Figura B.6 representa-se o diagrama dos testes realizados com os valores obtidos apresentados na Tabela B.2.

Apêndice B – Testes para determinação dos parâmetros do barco autônomo

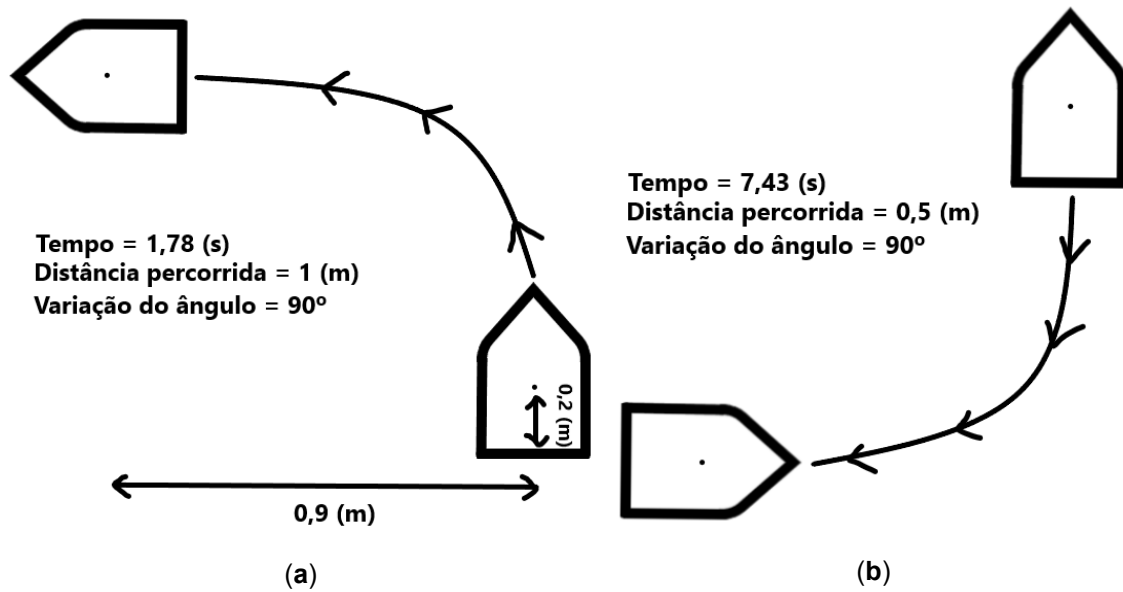


Figura B.6 – Diagrama do teste realizado de propulsão do barco com o motor direito: (a) Motor a rodar para a frente; (b) Motor a rodar para trás.

Tabela B.2 - Variáveis obtidas do modelo hidrodinâmico do barco.

Descrição da constante	Nomenclatura da constante	Valor da constante
Massa do barco	m	1,8 (kg)
Velocidade linear para a frente	v	0,55 (m/s)
Velocidade de rotação para a frente	ω	0,88 (rad/s)
Velocidade linear para trás	v	0,068 (m/s)
Velocidade de rotação para trás	ω	0,21 (rad/s)
Constante de atrito da velocidade de rotação	C_{ω}	0,63
Constante de atrito da velocidade linear	C_v	5,3

B.3 Testes para determinar os parâmetros dos motores

Utilizou-se um tacômetro [49] para medir a velocidade de rotação dos motores, juntamente com um sensor de corrente [50] colocado na entrada do

Apêndice B – Testes para determinação dos parâmetros do barco autónomo

motor. Obtém-se a constante k_b através da equação (4.1), com os motores do barco dentro da água e cálculo apresentado na secção B.4. No motor DC a constante k_t é igual à constante k_b [30]. As variáveis obtidas são apresentadas na Tabela B.3.

Tabela B.3 - Variáveis obtidas para obter os parâmetros modelo do motor DC.

Descrição da constante	Nomenclatura da constante	Valor da constante
Velocidade de rotação máxima do motor esquerdo	Ω_1	256 (rad/s)
Velocidade de rotação máxima do motor direito	Ω_2	256 (rad/s)
Corrente máxima da armadura do motor esquerdo	i_1	1,6 (A)
Corrente máxima da armadura do motor direito	i_2	1,6 (A)
Tensão da bateria aplicada ao motor esquerdo	V_1	7,5 (V)
Tensão da bateria aplicada ao motor direito	V_2	7,5 (V)
Constante da força eletromotriz induzida do motor esquerdo e do motor direito	k_b	0,02
Ganho da corrente do motor esquerdo e do motor direito	k_t	0,02

Sabendo que a reação do motor demora cerca de 200 milissegundos a transitar entre a velocidade máxima e a velocidade nula, e que a equação (4.4) indica que α é a constante de tempo dessa reação, calcula-se α .

Tendo a corrente consegue-se calcular o binário motor, pela equação (4.2). Com a velocidade de rotação do motor consegue-se calcular, através da equação (2.3), a constante β .

O cálculo dessa variável é apresentado na secção B.4. Calcula-se a constante C_t para cada turbina e direção de rotação, relacionando a velocidade de rotação do motor com a força gerada, dada pela equação (4.5), com o cálculo

Apêndice B – Testes para determinação dos parâmetros do barco autónomo

apresentado na secção B.4. A Tabela B.4 tem os restantes valores dos parâmetros do motor e da turbina.

Tabela B.4 - Variáveis obtidas para o modelo de Yoerger [27].

Descrição da constante	Nomenclatura da constante	Valor da constante
Binário motor gerado do motor esquerdo	τ_{g_1}	0,032 (Nm)
Binário motor gerado do motor direito	τ_{g_2}	0,032 (Nm)
Constante de atrito aquático do motor esquerdo	α_1	0,037
Constante de atrito aquático do motor direito	α_2	0,037
Ganho do binário motor do motor esquerdo	β_1	75000
Ganho do binário motor do motor direito	β_2	75000
Ganho da velocidade de rotação do motor esquerdo ao rodar para a frente	C_{t11}	0,00002
Ganho da velocidade de rotação do motor esquerdo ao rodar para trás	C_{t12}	0,000001
Ganho da velocidade de rotação do motor direito ao rodar para a frente	C_{t21}	0,00003
Ganho da velocidade de rotação do motor direito ao rodar para a trás	C_{t22}	0,000001

B.4 Determinação dos parâmetros apresentados no Apêndice B

Para determinar a força diferencial, a constante equivalente da força diferencial e a constante equivalente da força comum, utiliza-se a equação (B.1), (B.2) e (B.3) substituindo pelos valores obtidos nos testes da secção B.2 no Apêndice B, realizou-se a seguinte dedução:

$$T_2^2 = T_{dif}^2 + T_{com}^2 \Leftrightarrow T_{dif}^2 = 2,1^2 - 1,6^2 \Leftrightarrow T_{dif} = \pm\sqrt{2,1^2 - 1,6^2}$$

$$\Leftrightarrow T_{dif} = \pm\sqrt{1,85} \cong \pm 1,36 (N)$$

$$C_1 = \frac{T_{dif}}{T_1 - T_2} = \frac{-1,36}{0 - 2,1} \cong 0,65$$

$$C_2 = \frac{T_{com}}{T_1 + T_2} = \frac{1,6}{0 + 2,1} \cong 0,76$$

Para determinar as constantes de atrito da velocidade linear e velocidade de rotação, C_v e C_ω , utilizam-se as equações (4.10) e (4.11) com os valores amostrados na secção B.2 do Apêndice B, e considerando o estado estacionário, realizou-se a seguinte dedução:

$$T_{com} - C_v v|v| = m \frac{dv}{dt} \Rightarrow$$

Em estado estacionário:

$$\Rightarrow T_{com} = C_v v|v| \Leftrightarrow C_v = \frac{T_{com}}{v|v|} = \frac{1,6}{0,55^2} \cong 5,3$$

$$T_{dif} - C_\omega \omega|\omega| = mv\omega \Leftrightarrow C_\omega \omega|\omega| = T_{dif} - mv\omega \Leftrightarrow$$

$$\Leftrightarrow C_\omega = \frac{T_{dif} - mv\omega}{\omega|\omega|} = \frac{1,36 - 1,8 * 0,55 * 0,88}{0,88^2} \cong 0,63$$

Para determinar a constante da força eletromotriz induzida, k_b , para cada motor, utiliza-se a equação (4.1) com os valores amostrados na secção B.3 do Apêndice B e considerando o estado estacionário, realizou-se a seguinte dedução:

$$L_j \frac{di_j}{dt} = V_j - k_b \Omega_j - R_j i_j \Leftrightarrow$$

Tendo em conta que os valores dos parâmetros dos motores e que se amostra no estado estacionário:

$$\Leftrightarrow k_b = \frac{V_j - R_j i_j}{\Omega_j} = \frac{7,5 - 1,5 * 1,6}{256} \cong 0,02$$

Apêndice B – Testes para determinação dos parâmetros do barco autónomo

Para determinar o binário de motor, τ_g , para cada motor, utiliza-se a equação (4.2) com os valores amostrados na secção B.3 do Apêndice B, como mostra a seguinte dedução:

$$\tau_{g_j} = k_t \dot{i}_j \Leftrightarrow \tau_{g_j} \cong 0,02 * 1,6 \cong 0,032 \text{ (Nm)}$$

Para determinar o ganho de binário motor, β , para cada turbina, utiliza-se a equação (2.3) com os valores amostrados na secção B.3 do Apêndice B e considerando o estado estacionário, como mostra a seguinte dedução:

$$\frac{d\Omega}{dt} = \beta\tau - \alpha\Omega|\Omega| \Leftrightarrow$$

Em estado estacionário:

$$\Leftrightarrow \beta\tau = \alpha\Omega|\Omega| \Leftrightarrow \beta = \frac{\alpha\Omega|\Omega|}{\tau} \cong \frac{0,037 * 256^2}{0,032} \cong 75000$$

Para calcular o ganho da velocidade de rotação, C_t , para cada turbina e direção de rotação, utiliza-se a equação (4.5) com os valores amostrados na secção B.3 do Apêndice B, como mostra o seguinte cálculo:

$$T_j = C_t \Omega_j |\Omega_j| \Leftrightarrow C_t = \frac{T_j}{\Omega_j |\Omega_j|} \Rightarrow$$

$$\Rightarrow C_{t11} = \frac{1,4}{256 * 256} \cong 0,00002$$

$$\Rightarrow C_{t12} = \frac{0,1}{256 * 256} \cong 0,000001$$

$$\Rightarrow C_{t21} = \frac{2,1}{256 * 256} \cong 0,00003$$

$$\Rightarrow C_{t22} = \frac{0,1}{256 * 256} \cong 0,000001$$

Apêndice C

CONFIGURAÇÃO DO SISTEMA DO BARCO AUTÓNOMO

Neste apêndice são apresentados os códigos utilizados nesta dissertação, a realização da calibração do sensor de direção, o código que realiza o processamento do barco, o código do telecomando e o código da simulação.

C.1 Amostragem dos sensores do barco autónomo e calibração do sensor de direção

Para amostrar o GPS e o sensor 9DOF utilizam-se as bibliotecas desenvolvidas pela *Adafruit* em relação aos componentes utilizados, com a linguagem do *CircuitPython* [51]-[53].

O código apresentado na Figura C.1 mostra a maneira utilizada para amostrar os dados do GPS.

```
import board
import adafruit_gps
import busio

gps_serial = busio.UART(board.GP4, board.GP5,
baudrate=9600) #Configuração do protocolo de comunicação
UART

my_gps = adafruit_gps.GPS(gps_serial) #Associação do
dispositivo GPS à linha de comunicação UART

GPS=[0.0,0.0]

def SampleGPS():
    my_gps.update() #Atualiza o estado do GPS
    global GPS
    #Atualiza o GPS se receber novos dados
    if my_gps.latitude != None:
        GPS = [my_gps.latitude, my_gps.longitude]
```

Figura C.1 – Código de amostragem dos dados GPS.

Com o GPS calcula-se o ângulo de direção de referência através da equação (2.8), apresentada na secção 2.9 do Capítulo 2, como mostra o código apresentado na Figura C.2.

```
from ulab import numpy as np
#Cálculo do ângulo de direção entre duas coordenadas GPS
def CourseTo(lat1, long1, lat2, long2):
    #Referenciado com Norte a 0° e Oeste a -90°
    dlon = np.radians(long2-long1) #Converte em radianos
    lat1 = np.radians(lat1)
```

Apêndice C – Configuração do sistema do barco autónomo

```
lat2 = np.radians(lat2)
#Algoritmo de cálculo do ângulo
a1 = np.sin(dlon) * np.cos(lat2)
a2 = np.sin(lat1) * np.cos(lat2) * np.cos(dlon)
a2 = np.cos(lat1) * np.sin(lat2) - a2
a2 = np.arctan2(a1, a2) #Ângulo de direção
return np.degrees(a2) #Converte em graus
```

Figura C.2 – Código do cálculo da direção pretendida através das coordenadas GPS.

Para amostrar o sensor 9DOF faz-se em duas partes, numa primeira amostra-se a aceleração e o giroscópio, amostragem do 9DOF, e na segunda juntam-se esses dados ao magnetómetro e amostra-se a direção, obtendo o código apresentado na Figura C.3.

```
import board
import busio
import adafruit_lsm9ds1

i2c = busio.I2C(board.GP3, board.GP2)
sensor9DOF = adafruit_lsm9ds1.LSM9DS1_I2C(i2c)
NineDOF = [0,0,0,0,0,0,0,0,0,0]
GyroBias = 0
Gyro = [0,0,0]
acl = [0,0,0]
#Amostragem da velocidade de rotação + acelerómetro
def Sample6DOF():
    global Gyro
    global acl
    gyro[0],gyro[1],gyro[2] = sensor9DOF.gyro
    acl[0],acl[1],acl[2] = sensor9DOF.acceleration
    Gyro = gyro[2] - GyroBias #velocidade de rotação no
    ângulo de direção

#Amostragem dos dados do 9DOF
```

Apêndice C – Configuração do sistema do barco autónomo

```
def Sample9DOF(GyroMed, AclMed): #Tem como entrada a
média do giroscópio e acelerómetro

    global NineDOF

    mag[0],mag[1],mag[2] = sensor9DOF.magnetic

    NineDOF = [AclMed[0], AclMed[1], AclMed[2], 0, 0,
GyroMed, mag[0]*100, mag[1]*100, mag[2]*100] #Só
interessa a rotação do plano da Terra as outras igualam a
zero
```

Figura C.3 – Código de amostragem dos dados 9DOF.

Para obter a direção implementam-se em código as operações matemáticas descritas na secção 6.2 do Capítulo 6, utilizando as equações (6.1), (6.2) e (6.3), como mostra o código apresentado na Figura C.4, utilizando a biblioteca *numpy* [51] para realizar as operações matemáticas.

```
from ulab import numpy as np
#Estimacao do valor da direção atual
def EstYaw(Accel, Mag):

    #Para estimar o angulo da direção do barco
    (compasso) utiliza-se o magnetómetro e os outros ângulos
    de rotação

    #Assume-se que a densidade de fluxo magnético única
    a ser aplicada no magnetómetro foi o fluxo magnético
    terrestre e que

    #o eixo dos Xs está alinhado na direção do barco e o
    eixo dos Zs está alinhado com a gravidade da terra

    #Para estimar os outros ângulos de rotação utiliza-
    se o acelerómetro

    #Assume-se que a única aceleração foi gravítica e
    que

    #o eixo dos Zs do acelerómetro para dar 0° de
    rotação nos eixos, tem de estar alinhado com a gravidade
    da terra

    #Esta estimação tem erros devido a aplicação de
    outras forças no barco, ruído e calibração

    global roll

    global pitch
```

Apêndice C – Configuração do sistema do barco autônomo

```
roll = np.arctan2(Accel[0], Accel[2]) #Cálculo do
ângulo de enrolamento

pitch = np.arctan2(Accel[1],
np.sqrt(Accel[0]*Accel[0] + Accel[2]*Accel[2])) #Cálculo
do ângulo de inclinação

measuredYaw = -np.degrees(np.arctan2(np.sin(roll)*
Mag[2] + np.cos(roll)*Mag[0], np.cos(pitch)*Mag[1] +
np.sin(roll)*np.sin(pitch)*Mag[0] - np.cos(roll)*
np.sin(pitch)*Mag[2])) #Cálculo do ângulo de direção

return measuredYaw
```

Figura C.4 – Código de estimação da direção do barco.

A calibração do dispositivo 9DOF é necessária para corrigir as distorções causadas pelas interferências eletromagnéticas por todo o próprio circuito do barco. Idealmente, as referências amostradas do campo magnético e da aceleração gravítica não podem variar a intensidade em todas as direções, por isso, mapeiam-se os valores obtidos em X_s , Y_s e Z_s para uma superfície esférica centrada em (0,0,0), sendo assim o valor médio na direção dos X_s , Y_s e Z_s igual a zero, e o raio dessa esfera é o módulo do campo magnético terrestre.

Para calibrar o magnetómetro é preciso multiplicar os valores X_s , Y_s e Z_s por uma matriz que contraria o escalar da intensidade nas direções e é preciso somar os desvios para anular as constantes de cada direção. Utiliza-se para tal calibração o programa *MotionCal* [42], que faz a leitura da rotação do objeto ao longo do tempo para amostrar diversos pontos. O programa realiza a média desses pontos de maneira a encontrar os desvios, centralizando a superfície esférica em (0,0,0), e remapeando os pontos para formar a superfície esférica. Com isso é construída a matriz que ao multiplicar pelos dados amostrados e ao eliminar os desvios, calibra os dados amostrados [42].

Na Figura C.5 é apresentada a formatação dos dados para serem encaminhados para o programa *MotionCal* [42].

```
import board
import time
import busio
import adafruit_lsm9ds1
```

Apêndice C – Configuração do sistema do barco autônomo

```
#Config 9DOF
i2c = busio.I2C(board.GP3, board.GP2) #Configuração do
protocolo de comunicação I2C
sensor9DOF = adafruit_lsm9ds1.LSM9DS1_I2C(i2c)
#Associação do dispositivo 9DOF à linha de comunicação
I2C
mag = [0,0,0]
acl = [0,0,0]
gyro = [0,0,0]
while True:
    #Comando de leitura de cada variável do 9DOF
    mag[0],mag[1],mag[2] = sensor9DOF.magnetic
    acl[0],acl[1],acl[2] = sensor9DOF.acceleration
    gyro[0],gyro[1],gyro[2] = sensor9DOF.gyro
    #Envio dos dados para a linha de comunicação COM
    print('Raw:'+ str(int(acl[0]*8192/9.8)) + ',' +
str(int(acl[1]*8192/9.8)) + ',' +
str(int(acl[2]*8192/9.8)) + ',' +
str(int(57.29578*gyro[0]*16)) + ',' +
str(int(57.29578*gyro[1]*16)) + ',' +
str(int(57.29578*gyro[2]*16)) + ',' +
str(int(mag[0]*1000)) + ',' + str(int(mag[1]*1000)) + ',' +
+ str(int(mag[2]*1000)))
    time.sleep(0.2)
```

Figura C.5 – Código de formatação dos dados do sensor 9DOF, para os encaminhar para o programa *MotionCal* [42].

Após a calibração do magnetômetro, obtém-se a matriz de correção para os valores do campo magnético nos três eixos, dada pelo programa *MotionCal* [42], com os valores e a correção apresentada na equação (C.1), sendo Mag_{Xs} , Mag_{Ys} , Mag_{Zs} os valores do campo magnético no eixo dos Xs , Ys e Zs e Mag_{Xs}' , Mag_{Ys}' , Mag_{Zs}' os valores do campo magnético no eixo dos Xs , Ys e Zs calibrados, respetivamente.

$$\begin{bmatrix} Mag_{Xs}' \\ Mag_{Ys}' \\ Mag_{Zs}' \end{bmatrix} = \begin{bmatrix} 1,021 & 0,072 & 0,007 \\ 0,072 & 0,988 & 0,005 \\ 0,007 & 0,005 & 0,996 \end{bmatrix} * \begin{bmatrix} Mag_{Xs} \\ Mag_{Ys} \\ Mag_{Zs} \end{bmatrix} + \begin{bmatrix} 10,9 \\ 11,17 \\ 1,43 \end{bmatrix} \quad (C.1)$$

Apêndice C – Configuração do sistema do barco autônomo

Para calibrar o acelerômetro e o giroscópio basta remover os desvios, ou seja, é somente necessário fazer a média dos dados com o barco parado. É necessário ter em atenção que no acelerômetro a aceleração gravítica perturba a média, ou seja, remove-se o valor da referência do eixo dos Zs.

Na Figura C.6 é apresentado o código para a obtenção dos dados de calibração do acelerômetro e na Figura C.7 é apresentado o código para a obtenção dos dados de calibração do giroscópio.

```
import board
import time
import busio
import adafruit_lsm9ds1

#Config 9DOF
i2c = busio.I2C(board.GP3, board.GP2) #Configuração do
protocolo de comunicação I2C
sensor9DOF = adafruit_lsm9ds1.LSM9DS1_I2C(i2c)
#Associação do dispositivo 9DOF à linha de comunicação
I2C
acl = [0.0, 0.0, 0.0]
bias = [0.0, 0.0, 0.0]
i = 1
while True:
    #Amostragem da aceleração
    acl[0],acl[1],acl[2] = sensor9DOF.acceleration
    #Cálculo do desvio atual
    bias = [bias[0]*(i-1)/i + acl[0]/i, bias[1]*(i-1)/i +
acl[1]/i, bias[2]*(i-1)/i + acl[2]/i] #Adiciona o valor
anterior à média
    i = i + 1
    print(bias)
    time.sleep(0.2)
```

Figura C.6 – Código de calibração do acelerômetro.

```
import board
import time
```

Apêndice C – Configuração do sistema do barco autónomo

```
import busio
import adafruit_lsm9ds1
#Config 9DOF
i2c = busio.I2C(board.GP3, board.GP2) #Configuração do
protocolo de comunicação I2C
sensor9DOF = adafruit_lsm9ds1.LSM9DS1_I2C(i2c)
#Associação do dispositivo 9DOF à linha de comunicação
I2C
gyro = [0.0, 0.0, 0.0]
bias = [0.0, 0.0, 0.0]
i = 1
while True:
    #Amostragem da velocidade de rotação
    gyro[0],gyro[1],gyro[2] = sensor9DOF.gyro
    #Cálculo do desvio atual
    bias = [bias[0]*(i-1)/i + gyro[0]/i, bias[1]*(i-1)/i
+ gyro[1]/i, bias[2]*(i-1)/i + gyro[2]/i] #Adiciona o
valor anterior à média
    i = i + 1
    print(bias)
    time.sleep(0.2)
```

Figura C.7 – Código de calibração do giroscópio.

Após a calibração do acelerómetro e do giroscópio, obtém-se a correção dos valores da aceleração nos três eixos e da velocidade de rotação nos três eixos. Com os valores e correções obtidas, através da execução do código apresentado na Figura C.6 e na Figura C.7, apresentam-se as equações (C.2) e (C.3) que realizam a calibração do acelerómetro e do giroscópio, respetivamente [42]. Sendo Acl_{Xs} , Acl_{Ys} e Acl_{Zs} os valores de aceleração no eixo dos Xs , Ys e Zs , Acl_{Xs}' , Acl_{Ys}' e Acl_{Zs}' os valores de aceleração no eixo dos Xs , Ys e Zs calibrados, e $Gyro_{Xs}$, $Gyro_{Ys}$ e $Gyro_{Zs}$ os valores de velocidade de rotação no plano perpendicular ao eixo dos Xs , Ys e Zs e $Gyro_{Xs}'$, $Gyro_{Ys}'$ e $Gyro_{Zs}'$ os valores de velocidade de rotação no plano perpendicular ao eixo dos Xs , Ys e Zs calibrados, respetivamente.

Apêndice C – Configuração do sistema do barco autónomo

$$\begin{bmatrix} Acl_{Xs}' \\ Acl_{Ys}' \\ Acl_{Zs}' \end{bmatrix} = \begin{bmatrix} Acl_{Xs} \\ Acl_{Ys} \\ Acl_{Zs} \end{bmatrix} + \begin{bmatrix} 0,04 \\ 0,6 \\ 0,2 \end{bmatrix} \quad (\text{B.2})$$

$$\begin{bmatrix} Gyro_{Xs}' \\ Gyro_{Ys}' \\ Gyro_{Zs}' \end{bmatrix} = \begin{bmatrix} Gyro_{Xs} \\ Gyro_{Ys} \\ Gyro_{Zs} \end{bmatrix} + \begin{bmatrix} -0,0047 \\ 0,008 \\ 0,025 \end{bmatrix} \quad (\text{B.3})$$

Através das equações (C.1), (C.2) e (C.3), implementa-se a correção dos valores amostrados, subtraindo elemento a elemento os vetores do magnetómetro, acelerómetro e giroscópio dos seus desvios e multiplicando o vetor do magnetómetro pela matriz de correção [42]. Na Figura C.8 é apresentado o código de calibração dos valores amostrados.

```
import board
import busio
import adafruit_lsm9ds1
from ulab import numpy as np
#Config 9DOF
i2c = busio.I2C(board.GP3, board.GP2) #Configuração do
protocolo de comunicação I2C
sensor9DOF = adafruit_lsm9ds1.LSM9DS1_I2C(i2c)
#Associação do dispositivo 9DOF à linha de comunicação
I2C
GyroMed = 0
GyroBias = 0.0247
AclMed = [0,0,0]
NineDOF = np.array([0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0])
NineDOFCal = np.array([0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0])
for k in range(50): #50 Amostras do 6DOF
    Sample6DOF()
    GyroMed = GyroMed + Gyro/50
    AclMed[0] = AclMed[0] + acl[0]/50
    AclMed[1] = AclMed[1] + acl[1]/50
```

Apêndice C – Configuração do sistema do barco autónomo

```
AclMed[2] = AclMed[2] + acl[2]/50
Sample9DOF(GyroMed, AclMed) #Amostragem do 9DOF
NineDOFBias = [0.04, 0.6, 0.2, -0.0047, 0.008, 0.025,
10.9, 11.17, 1.43] #Desvios em relação ao valor
pretendido
SoftIronCal = np.array([[1.021, 0.072, 0.007],[0.072,
0.988, 0.005],[0.007, 0.005, 0.996]]) #Matriz de
calibração
#Calibração das amostras
NineDOFCal = [NineDOF[0] - NineDOFBias[0], NineDOF[1] -
NineDOFBias[1], NineDOF[2] - NineDOFBias[2], NineDOF[3] -
NineDOFBias[3], NineDOF[4] - NineDOFBias[4], GyroMed,
NineDOF[6] - NineDOFBias[6], NineDOF[7] - NineDOFBias[7],
NineDOF[8] - NineDOFBias[8]] #Remoção dos desvios
NineDOFCal[6:] = [NineDOFCal[6]*SoftIronCal[0][0] +
NineDOFCal[7]*SoftIronCal[0][1] +
NineDOFCal[8]*SoftIronCal[0][2],
NineDOFCal[6]*SoftIronCal[1][0] +
NineDOFCal[7]*SoftIronCal[1][1] +
NineDOFCal[8]*SoftIronCal[1][2],
NineDOFCal[6]*SoftIronCal[2][0] +
NineDOFCal[7]*SoftIronCal[2][1] +
NineDOFCal[8]*SoftIronCal[2][2]] #Remoção da distorção do
campo magnético
AngAtual = EstYaw(NineDOFCal[:3], NineDOFCal[6:])
#Cálculo da direção atual
```

Figura C.8 – Código de calibração dos dados do sensor 9DOF.

C.2 Comunicação com o telecomando do barco

Para interagir entre o rádio e o microcontrolador é preciso programar os códigos de linha SPI [45] com a sequência de *bytes* equivalentes para cada ação que se pretende fazer. Denominam-se, por isso, três funções principais do rádio para estabelecer a comunicação entre os dois rádios, do telecomando e do barco. Uma função é responsável pela escrita das configurações, *writeConfig()*, outra função é responsável pela transmissão da mensagem, *sendstr()*, e outra função é responsável pela escuta e receção da mensagem transmitida, *listen()*.

Apêndice C – Configuração do sistema do barco autónomo

Implementa-se a configuração do rádio com a escrita da função `writeConfig()` apresentada no excerto de código da Figura C.9.

```
import board
import busio
from digitalio import DigitalInOut, Direction, Pull

SpiRF905 = busio.SPI(board.GP10, board.GP11, board.GP12)
#Configuração do protocolo de comunicação SPI
while not SpiRF905.try_lock():
    pass
SpiRF905.configure(baudrate=106000)
CSN = DigitalInOut(board.GP13) #Chip select do rádio
CSN.direction = Direction.OUTPUT
PWR = DigitalInOut(board.GP7) #Liga o rádio
PWR.direction = Direction.OUTPUT
PWR.value = 1

def writeConfig(CH_NO, HFREQ_PLL, PA_PWR, RX_RED_PWR,
AUTO_RETRAN, TX_AFW, RX_AFW, RX_PW, TX_PW, address1,
address2, address3, address4, CRC_MODE, CRC_EN, XOF,
UP_CLK_EN, UP_CLK_FREQ):
    CSN.value = 0 #Habilita a escrita
    SpiRF905.write(bytes([0x00])) #comando escrever no
registo de configuração
    SpiRF905.write(bytes([(CH_NO%256)])) #frequência
central 1
    SpiRF905.write(bytes([AUTO_RETRAN*32 + RX_RED_PWR*16
+ PA_PWR*4 + HFREQ_PLL*2 + CH_NO//256]))
    SpiRF905.write(bytes([TX_AFW*16 + RX_AFW])) #tamanho
do endereço
    SpiRF905.write(bytes([RX_PW])) #tamanho payload RX
    SpiRF905.write(bytes([TX_PW])) #tamanho payload TX
    SpiRF905.write(bytes([address1, address2, address3,
address4])) #Endereço do rádio
    SpiRF905.write(bytes([CRC_MODE*128 + CRC_EN*64 +
XOF*8 + UP_CLK_EN*4 + UP_CLK_FREQ]))
    CSN.value = 1 #Desabilita a escrita
```

Apêndice C – Configuração do sistema do barco autónomo

```
writeConfig(116, 0, 3, 0, 0, 4, 4, 32, 32, 0xA2, 0xA2,  
0xA2, 0xA2, 1, 1, 3, 0, 0)
```

Figura C.9 – Código da função de configuração do rádio nRF905 [22].

A implementação da função responsável pela transmissão da mensagem, *sendstr()*, é apresentada no excerto de código da Figura C.10.

```
import board  
import busio  
import time  
from digitalio import DigitalInOut, Direction, Pull  
SpiRF905 = busio.SPI(board.GP10, board.GP11, board.GP12)  
#Configuração do protocolo de comunicação SPI  
while not SpiRF905.try_lock():  
    pass  
SpiRF905.configure(baudrate=106000)  
CSN = DigitalInOut(board.GP13)  
CSN.direction = Direction.OUTPUT  
TRX_CE = DigitalInOut(board.GP8)  
TRX_CE.direction = Direction.OUTPUT  
PWR = DigitalInOut(board.GP7)  
PWR.direction = Direction.OUTPUT  
PWR.value = 1  
TXEN = DigitalInOut(board.GP6)  
TXEN.direction = Direction.OUTPUT  
  
def sendstr(message, address1, address2, address3,  
address4):  
    TRX_CE.value = 0  
    TXEN.value = 1  
    time.sleep(0.04)  
    CSN.value = 0  
    SpiRF905.write(bytes([0x20])) #comando para escrever  
a mensagem no rádio  
    SpiRF905.write(bytes(message))
```

Apêndice C – Configuração do sistema do barco autónomo

```
CSN.value = 1
time.sleep(0.04)
CSN.value = 0

SpiRF905.write(bytes([0x22])) #comando para escrever
o endereço destino no rádio

SpiRF905.write(bytes([address1, address2, address3,
address4]))

CSN.value = 1

TRX_CE.value = 1
time.sleep(0.04)
TRX_CE.value = 0
```

Figura C.10 – Código da função de transmissão da mensagem do rádio nRF905 [22].

A função responsável pela receção da mensagem, *listen()*, é apresentada no excerto de código da Figura C.11.

```
import board
import busio
import time
from digitalio import DigitalInOut, Direction, Pull

SpiRF905 = busio.SPI(board.GP10, board.GP11, board.GP12)
while not SpiRF905.try_lock():
    pass

SpiRF905.configure(baudrate=106000)
CSN = DigitalInOut(board.GP13)
CSN.direction = Direction.OUTPUT
TRX_CE = DigitalInOut(board.GP8)
TRX_CE.direction = Direction.OUTPUT
PWR = DigitalInOut(board.GP7)
PWR.direction = Direction.OUTPUT
PWR.value = 1
TXEN = DigitalInOut(board.GP6)
TXEN.direction = Direction.OUTPUT

def listen():
```

Apêndice C – Configuração do sistema do barco autónomo

```
TXEN.value = 0
TRX_CE.value = 1
global data
if DR.value != 0:
    time.sleep(0.04)
    TRX_CE.value = 0
    CSN.value = 0 #Habilita a escrita
    SpiRF905.write(bytes([0x24])) #Comando de leitura
dos dados
    SpiRF905.readinto(data) #Leitura dos dados
    CSN.value = 1
    time.sleep(0.04)
    TRX_CE.value = 1
```

Figura C.11 – Código da função de receção da mensagem do rádio nRF905 [22].

Na comunicação por rádio utiliza-se o primeiro *byte* da mensagem para identificar o comando, sendo 0x80 o controlo manual, 0x40 o controlo autónomo e 0x20 uma nova trajetória, como apresentado no formato da trama no Capítulo 6, na secção 6.3 (Figura 6.3). O telecomando foi programado de maneira a enviar as mensagens através da interação com o utilizador, com o código apresentado na Figura C.12.

```
import board
import busio
import time
import struct
from digitalio import DigitalInOut, Direction, Pull
from analogio import AnalogIn

def writeConfig(CH_NO, HFREQ_PLL, PA_PWR, RX_RED_PWR,
AUTO_RETRAN, TX_AFW, RX_AFW, RX_PW, TX_PW, address1,
address2, address3, address4, CRC_MODE, CRC_EN, XOF,
UP_CLK_EN, UP_CLK_FREQ):
    CSN.value = 0 #Habilita a escrita
```

Apêndice C – Configuração do sistema do barco autónomo

```
SpiRF.write(bytes([0x00])) #comando escrever no
registo de configuração

SpiRF.write(bytes([(CH_NO%256)])) #frequência central
1

SpiRF.write(bytes([AUTO_RETRAN*32 + RX_RED_PWR*16 +
PA_PWR*4 + HFREQ_PLL*2 + CH_NO//256]))

SpiRF.write(bytes([TX_AFW*16 + RX_AFW])) #tamanho do
endereço

SpiRF.write(bytes([RX_PW])) #tamanho payload RX

SpiRF.write(bytes([TX_PW])) #tamanho payload TX

SpiRF.write(bytes([address1, address2, address3,
address4])) #Endereço do rádio

SpiRF.write(bytes([CRC_MODE*128 + CRC_EN*64 + XOF*8 +
UP_CLK_EN*4 + UP_CLK_FREQ]))

CSN.value = 1 #Desabilita a escrita

def sendstr(message, address1, address2, address3,
address4):

    TRX_CE.value = 0

    TXEN.value = 1

    time.sleep(0.04)

    CSN.value = 0

    SpiRF.write(bytes([0x20])) #comando para escrever a
mensagem no rádio

    SpiRF.write(bytes(message))

    CSN.value = 1

    time.sleep(0.04)

    CSN.value = 0

    SpiRF.write(bytes([0x22])) #comando para escrever o
endereço destino no rádio

    SpiRF.write(bytes([address1, address2, address3,
address4]))

    CSN.value = 1

    TRX_CE.value = 1

    time.sleep(0.04)
```

Apêndice C – Configuração do sistema do barco autônomo

```
TRX_CE.value = 0

def listen():
    TXEN.value = 0
    TRX_CE.value = 1
    global data
    if DR.value != 0:
        time.sleep(0.04)
        TRX_CE.value = 0
        CSN.value = 0 #Habilita a escrita
        SpiRF.write(bytes([0x24])) #Comando de leitura
dos dados
        SpiRF.readinto(data) #Leitura dos dados
        CSN.value = 1
        time.sleep(0.04)
        TRX_CE.value = 1

if __name__ == '__main__': #Código principal
    #Configuração da comunicação SPI
    SpiRF = busio.SPI(board.GP10, board.GP11, board.GP12)
    while not SpiRF.try_lock():
        pass
    SpiRF.configure(baudrate=106000)
    CSN = DigitalInOut(board.GP13)
    CSN.direction = Direction.OUTPUT
    CSN.value = 1
    #Configuração dos pinos do rádio
    DR = DigitalInOut(board.GP9)
    DR.direction = Direction.INPUT
    TRX_CE = DigitalInOut(board.GP8)
    TRX_CE.direction = Direction.OUTPUT
    PWR = DigitalInOut(board.GP7)
```

Apêndice C – Configuração do sistema do barco autónomo

```
PWR.direction = Direction.OUTPUT
PWR.value = 1
TXEN = DigitalInOut(board.GP6)
TXEN.direction = Direction.OUTPUT
result = 0
writeConfig(116, 0, 3, 0, 0, 4, 4, 32, 32, 0xA2,
0xA2, 0xA2, 0xA2, 1, 1, 3, 0, 0)
#Configuração das entradas do utilizador
adc1 = AnalogIn(board.GP26) #Potenciómetro 1
adc2 = AnalogIn(board.GP27) #Potenciómetro 2
ChMod = DigitalInOut(board.GP22) #Botão troca modo de
funcionamento
ChMod.direction = Direction.INPUT
ChMod.pull = Pull.UP
SendCoord = DigitalInOut(board.GP18) #Botão envia
nova trajetória
SendCoord.direction = Direction.INPUT
SendCoord.pull = Pull.UP
ControloManual = True
ConfigRota = True
ConfigRota = ~ConfigRota
message = [0x00]
while True:
    if ChMod.value == False: #Se o utilizador
pressionou o botão
        ControloManual = ~ControloManual #Troca modo
funcionamento
        time.sleep(2)
    if SendCoord.value == False: #Se o utilizador
pressionou o botão
        ConfigRota = ~ConfigRota #Envia rota
        time.sleep(2)
    if ControloManual == True: #Se estiver no
controlo manual
```

Apêndice C – Configuração do sistema do barco autónomo

```
message = [0x80]

#Constrói a mensagem com os valores dos
potenciómetros para enviar para o barco e usa-los como
indicadores de velocidade

message = message + [adc1.value//256] +
[adc1.value%256] + [adc2.value//256] + [adc2.value%256]

message = message + [0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00]

sendstr(message, 0xA2, 0xA2, 0xA2, 0xA2)
#Envia a mensagem

time.sleep(0.1)

else:

#Constrói a mensagem com a indicação que está
no modo autónomo

message = [0x40]

message = message + [0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00]

sendstr(message, 0xA2, 0xA2, 0xA2, 0xA2)
#Envia a mensagem

if ConfigRota == True: #Se for para enviar nova
trajetória

message = [0x20, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00] #Indica o comando de nova trajetória

CoordGPS = [32.645265, -16.914576, 32.645287,
-16.914864, 0.0, 0.0] #Nova trajetória programada

for i in range(6): #Faz a leitura de cada
coordenada

ba = bytearray(struct.pack("d",
CoordGPS[i])) #Separa as coordenadas em bytes

message = message + [int(ba[4])] +
[int(ba[5])] + [int(ba[6])] + [int(ba[7])] #Adiciona na
mensagem
```

Apêndice C – Configuração do sistema do barco autónomo

```
        sendstr(message, 0xA2, 0xA2, 0xA2, 0xA2)
#Envia a mensagem

        print(message) #Visualiza-se a mensagem enviada
na consola

        time.sleep(0.1)
```

Figura C.12 – Código do telecomando do barco.

Na receção, no comando de envio das coordenadas GPS pretendidas para o barco, é feito o processo inverso da decomposição das coordenadas em *bytes*. Convertendo os *bytes* para coordenadas no código do barco, através da função *struct.unpack()*, como mostra a Figura C.13.

```
import struct
from ulab import numpy as np
CoordGPS = [0,0,0,0,0,0] #Array da trajetória
data = bytearray(32)
listen() #Receção da mensagem
if data[0] == 0x20: #Se receber o comando nova trajetória
    for j in range(6): #Preenche o array coordenada a
coordenada

        CoordGPS[j] = struct.unpack('d',
bytearray([0,0,0,0] + [data[8+j*4]] + [data[9+j*4]] +
[data[10+j*4]] + [data[11+j*4]])) #Converte conjunto de
bytes para float

    CoordGPS = np.array(CoordGPS) #Transforma em um array
onde se possa realizar operações matemáticas
```

Figura C.13 – Código da receção do comando de nova trajetória no barco autónomo.

C.3 Controlo do barco autónomo

Para controlar o barco é preciso inicializar a comunicação com os *drivers* do motor, enviado o sinal PWM equivalente para controlar o motor. No controlo manual, controla-se esse sinal de PWM através dos dados de velocidade recebidos pelo rádio do barco. Na Figura C.14 é apresentado o código de implementação do controlo manual do barco.

Apêndice C – Configuração do sistema do barco autônomo

```
import board
import struct
import time
import supervisor
import pwmio
import storage
from ulab import numpy as np

#Função de cálculo da diferença entre tempos
def ticks_diff(ticks1, ticks2):
    _TICKS_PERIOD = const(1<<29)
    _TICKS_MAX = const(_TICKS_PERIOD-1)
    _TICKS_HALFPERIOD = const(_TICKS_PERIOD//2)
    #Compute the signed difference between two ticks
    values, assuming that they are within 2**28 ticks
    diff = (ticks1 - ticks2) & _TICKS_MAX
    diff = ((diff + _TICKS_HALFPERIOD) & _TICKS_MAX) -
    _TICKS_HALFPERIOD
    return diff

listen() #Receção da mensagem

#Inicialização dos sinais PWM
pwm11 = pwmio.PWMOut(board.GP14, duty_cycle = 0,
frequency = 1000)
pwm12 = pwmio.PWMOut(board.GP15, duty_cycle = 0,
frequency = 1000)
pwm21 = pwmio.PWMOut(board.GP20, duty_cycle = 0,
frequency = 1000)
pwm22 = pwmio.PWMOut(board.GP17, duty_cycle = 0,
frequency = 1000)
data = bytearray(32)

#Amostragem de tempo, inicializa a contagem
Tanterior = supervisor.ticks_ms()
Tatual = supervisor.ticks_ms()
while data[0] == 0x80:
```

Apêndice C – Configuração do sistema do barco autónomo

```
ControloAutonomo = False
Tatual = supervisor.ticks_ms()
if ticks_diff(Tatual, Tanterior) > 5000: #De 5 em
5 segundos limpa o registo do rádio
    Tanterior = Tatual
    data[1] = 0x80
    data[2] = 0x80
    data[3] = 0x80
    data[4] = 0x80
    #Desabilita a escrita no microcontrolador
    storage.remount("/", readonly = True)
    #Limpa o registo do rádio
    PWR.value = 0
    time.sleep(0.1)
    PWR.value = 1
    time.sleep(0.1)
    #Através dos valores da mensagem cria o sinal PWM
    enviado para o driver do motor
    pwm11.duty_cycle = ((data[3]-128)*510 +
data[4])*(data[3]>170)
    pwm12.duty_cycle = ((128-data[3])*510 + (255-
data[4]))*(data[3]<80)
    pwm21.duty_cycle = ((data[1]-128)*510 +
data[2])*(data[1]>170)
    pwm22.duty_cycle = ((128-data[1])*510 + (255-
data[2]))*(data[1]<80)
    listen()
    if data[0] != 0x80: #Desliga os motores caso
acabe a comunicação do controlo manual
        PWR.value = 0
        time.sleep(0.1)
        PWR.value = 1
        time.sleep(0.1)
        pwm11.duty_cycle = 0
```

Apêndice C – Configuração do sistema do barco autónomo

```
pwm12.duty_cycle = 0
pwm21.duty_cycle = 0
pwm22.duty_cycle = 0
```

Figura C.14 – Código do controlo manual do barco.

Para realizar o controlo autónomo do barco, começa-se por definir a entrada no fluxo de código de execução do controlo autónomo, com uma variável booleana, que é ativada através do comando.

O sistema de controlo do barco autónomo tem duas malhas de realimentação. Na malha de realimentação externa realiza-se o controlo de direção, amostram-se os dados do sensor GPS para calcular a posição atual. A posição atual é utilizada para calcular a direção pretendida, θ_{ref} , através da implementação da equação (2.8) em código. Amostram-se os dados do sensor 9DOF para obter a direção atual, θ , e realiza-se a subtração da direção pretendida com a atual para se obter o erro de direção. Para obter a velocidade de rotação de referência, ω_{ref} , multiplica-se o erro de direção por k_p , o termo proporcional, e subtrai-se à média de velocidade de rotação, $\sim\omega$, multiplicada por k_d , o termo derivativo. Na equação (C.4) apresenta-se o cálculo da velocidade de rotação de referência na transformada de *Laplace* e na equação (C.5) apresenta-se o cálculo da velocidade de rotação de referência no domínio do tempo discreto.

$$(sk_d + k_p)(\theta_{ref}(s) - \theta(s)) = \omega_{ref}(s) \quad (C.4)$$

$$-k_d * \sim\omega(n) + k_p(\theta_{ref}(n) - \theta(n)) = \omega_{ref}(n) \quad (C.5)$$

Na malha de realimentação interior realiza-se o controlo da velocidade de rotação do barco. É necessário amostrar a velocidade de rotação do barco, ω , com o giroscópio e a diferença de tempo, Δt , por causa do termo integral. O termo integral é calculado através de um erro acumulado, somando ao erro acumulado anterior o valor do erro atual multiplicado pela diferença de tempo [32]. Para projetar a variável intermédia μ_{dif} , soma-se o erro acumulado multiplicado por k_{wi} , o termo integral, com o erro de velocidade de rotação multiplicado por k_w , o termo proporcional. Na equação (C.6) apresenta-se o

Apêndice C – Configuração do sistema do barco autónomo

cálculo de μ_{dif} na transformada de Laplace e na equação (C.7) apresenta-se o cálculo de μ_{dif} no domínio do tempo discreto.

$$\left(\frac{k_{wi}}{s} + k_w\right) (\omega_{ref}(s) - \omega(s)) = \mu_{dif}(s) \quad (C.6)$$

$$k_{wi} * \sum_1^n (\omega_{ref}(n) - \omega(n)) * \Delta t + k_w (\omega_{ref}(n) - \omega(n)) = \mu_{dif}(n) \quad (C.7)$$

Calcula-se também a variável intermédia μ_{com} através da multiplicação da velocidade linear de referência, v_{ref} , por uma constante k_v , cuja velocidade de referência é determinada em função do erro do ângulo, como é especificada na secção 3.4 do Capítulo 3. Na equação (C.8) apresenta-se o cálculo de μ_{com} , no domínio do tempo discreto.

$$k_v * v_{ref}(n) = \mu_{com}(n) \quad (C.8)$$

Somam-se essas variáveis intermédias para determinar o valor do ciclo PWM do motor direito e subtraem-se essas variáveis para obter o valor do ciclo PWM do motor esquerdo. Também é de notar que, no código, o valor das constantes k_v , k_w e k_{wi} é multiplicado por 65535, devido ao facto do *duty-cycle* variar entre 0 e 65535, em vez de variar entre 0 a 1.

Calcula-se a distância entre coordenadas GPS a partir da equação (2.9), comparando esse resultado com uma margem de 10 metros para determinar quando se passa para o próximo local, verificando se o barco autónomo está próximo do ponto de referência. Verifica-se, também, se o telecomando recebeu alguma mensagem com a indicação para terminar o controlo autónomo.

O código da Figura C.15 apresenta a implementação do controlo autónomo no barco.

```
import board
import struct
import time
import supervisor
import pwmio
import storage
import adafruit_lsm9ds1
```

Apêndice C – Configuração do sistema do barco autônomo

```
import adafruit_gps
from digitalio import DigitalInOut, Direction
from ulab import numpy as np

#Função de cálculo da diferença entre tempos
def ticks_diff(ticks1, ticks2):
    _TICKS_PERIOD = const(1<<29)
    _TICKS_MAX = const(_TICKS_PERIOD-1)
    _TICKS_HALFPERIOD = const(_TICKS_PERIOD//2)
    #Compute the signed difference between two ticks
    values, assuming that they are within 2**28 ticks
    diff = (ticks1 - ticks2) & _TICKS_MAX
    diff = ((diff + _TICKS_HALFPERIOD) & _TICKS_MAX) -
    _TICKS_HALFPERIOD
    return diff

#Cálculo da distância entre duas coordenadas GPS
def DistanceBetween(lat1, long1, lat2, long2):
    delta = np.radians(long1-long2)
    sdlong = np.sin(delta)
    cdlong = np.cos(delta)
    lat1 = np.radians(lat1)
    lat2 = np.radians(lat2)
    slat1 = np.sin(lat1)
    clat1 = np.cos(lat1)
    slat2 = np.sin(lat2)
    clat2 = np.cos(lat2)
    delta = (clat1 * slat2) - (slat1 * clat2 * cdlong)
    delta = delta**2
    delta += (clat2 * sdlong)**2
    delta = np.sqrt(delta)
    denom = (slat1 * slat2) + (clat1 * clat2 * cdlong)
    delta = np.arctan2(delta, denom)
```

Apêndice C – Configuração do sistema do barco autônomo

```
    return delta * 6371009 #distância = ângulo em
radianos vezes o raio da terra

#Inicialização dos sinais PWM
pwm11 = pwmio.PWMOut(board.GP14, duty_cycle = 0,
frequency = 1000)
pwm12 = pwmio.PWMOut(board.GP15, duty_cycle = 0,
frequency = 1000)
pwm21 = pwmio.PWMOut(board.GP20, duty_cycle = 0,
frequency = 1000)
pwm22 = pwmio.PWMOut(board.GP17, duty_cycle = 0,
frequency = 1000)

#Inicialização do GPS
gps_serial = busio.UART(board.GP4, board.GP5,
baudrate=9600)
my_gps = adafruit_gps.GPS(gps_serial)

#Inicialização do 9DOF
i2c = busio.I2C(board.GP3, board.GP2)
sensor9DOF = adafruit_lsm9ds1.LSM9DS1_I2C(i2c)

#Inicialização do nRF905
SpiRF905 = busio.SPI(board.GP10, board.GP11, board.GP12)
while not SpiRF905.try_lock():
    pass
SpiRF905.configure(baudrate=106000)
CSN = DigitalInOut(board.GP13)
CSN.direction = Direction.OUTPUT
CSN.value = 1
DR = DigitalInOut(board.GP9)
DR.direction = Direction.INPUT
TRX_CE = DigitalInOut(board.GP8)
TRX_CE.direction = Direction.OUTPUT
PWR = DigitalInOut(board.GP7)
PWR.direction = Direction.OUTPUT
```

Apêndice C – Configuração do sistema do barco autónomo

```
PWR.value = 1
TXEN = DigitalInOut(board.GP6)
TXEN.direction = Direction.OUTPUT
TXEN.value = 1
#Escrita da configuração no rádio
writeConfig(116, 0, 3, 0, 0, 4, 4, 32, 32, 0xA2, 0xA2,
0xA2, 0xA2, 1, 1, 3, 0, 0)
data = bytearray(32) #Dados da mensagem recebida
ControloAutonomo = True
#Inicialização de variáveis
ErrAccW = 0.0
Gyro = 0.0
GyroMed = 0.0
AclMed = [0.0, 0.0, 9.8]
GPS = [0.0, 0.0]
CoordGPS = [32.645189,-16.914811,32.645269,-
16.914545,0,0]
dist=10
i=0
#Matriz de calibração + correção
SoftIronCal = np.array([[1.021, 0.072, 0.007],[0.072,
0.988, 0.005],[0.007, 0.005, 0.996]])
NineDOFBias = [0.04, 0.6, 0.2, -0.0047 ,0.008 ,0.025
,10.9 , 11.17, 1.43]
GyroBias = 0.0247
#Variáveis de controlo
Kp = -0.02
Kd = -0.005*180/3.1416
Kw = 26214
Kwi = 262140
#Inicialização dos temporizadores
Tanterior = supervisor.ticks_ms()
Tanterior1 = Tanterior
```

Apêndice C – Configuração do sistema do barco autónomo

```
Tatual = T anterior
#Inicialização das variáveis 9DOF
mag = [0,0,0]
acl = [0,0,0]
gyro = [0,0,0]
roll = 0
pitch = 0
#Receção da mensagem de controlo autónomo
if data[0] == 0x40 and ControloAutonomo == False:
    ControloAutonomo = True
    #Inicialização dos temporizadores
    T anterior = supervisor.ticks_ms()
    T anterior1 = T anterior
    Tatual = T anterior
    #Habilita a escrita no microcontrolador
    storage.remount("/", readonly = False)
#Loop do controlo autónomo
while ControloAutonomo == True:
    SampleGPS() #Amostra o GPS
    Sample9DOF(GyroMed, AclMed) #Amostra o 9DOF
    #Realiza a calibração dos dados do 9DOF
    NineDOFCal = [NineDOF[0] - NineDOFBias[0], NineDOF[1]
- NineDOFBias[1], NineDOF[2] - NineDOFBias[2], NineDOF[3]
- NineDOFBias[3], NineDOF[4] - NineDOFBias[4], GyroMed,
NineDOF[6] - NineDOFBias[6], NineDOF[7] - NineDOFBias[7],
NineDOF[8] - NineDOFBias[8]]
    NineDOFCal[6:] = [NineDOFCal[6]*SoftIronCal[0][0] +
NineDOFCal[7]*SoftIronCal[0][1] +
NineDOFCal[8]*SoftIronCal[0][2],
NineDOFCal[6]*SoftIronCal[1][0] +
NineDOFCal[7]*SoftIronCal[1][1] +
NineDOFCal[8]*SoftIronCal[1][2],
NineDOFCal[6]*SoftIronCal[2][0] +
NineDOFCal[7]*SoftIronCal[2][1] +
NineDOFCal[8]*SoftIronCal[2][2]]
```

Apêndice C – Configuração do sistema do barco autónomo

```
#Cálculo da direção de referência
Angref = CourseTo(GPS[0], GPS[1], CoordGPS[i],
CoordGPS[i+1])

#Cálculo da direção atual
AngAtual = EstYaw(NineDOFCal[:3], NineDOFCal[6:])

#Cálculo da diferença entre os ângulos
ErrAng = Angref - AngAtual
if ErrAng > 180:
    ErrAng = ErrAng - 360
if ErrAng < -180:
    ErrAng = ErrAng + 360

#Escrita dos dados amostrados para análise
try:
    with open("dados.txt", "a") as f:
        f.write(str('%0.6f' % CoordGPS[i]) + ',' +
str('%0.6f' % CoordGPS[i+1]) + ',' + str('%0.6f' % GPS[0])
+ ',' + str('%0.6f' % GPS[1]) + ',' + str(Angref) + ',' +
str(AngAtual) + ',' + str(Tatual) + '\n')
except:
    pass

#Cálculo da velocidade de rotação de referência
Wref = ErrAng*Kp - Kd*GyroMed

#Limita a referência à sua máxima velocidade de
rotação
Wref = np.clip(Wref, -0.8, 0.8)

#Cálculo da velocidade de referência
vref = 0.5*(1 - abs(ErrAng)/90)

#Cálculo do sinal intermédio comum
Ucom = vref*75000
Ucom = np.clip(Ucom, -50000, 50000) #Limita o sinal

#Reset dos valores médios
GyroMed = 0.0
AclMed = [0.0, 0.0, 0.0]
```

Apêndice C – Configuração do sistema do barco autónomo

```
for k in range(50): #Loop interior do controlo
autónomo

    #Cálculo da diferença de tempo no loop interno
    Tactual = supervisor.ticks_ms()
    dt = float(ticks_diff(Tactual, Tanterior))/1000
    Tanterior = Tactual

    #Amostragem do giroscópio + acelerómetro
    Sample6DOF()

    #Cálculo do erro da velocidade de rotação
    ErrW = Wref - Gyro

    #Cálculo do erro acumulado da velocidade de
rotação

    ErrAccW = ErrAccW + ErrW*dt

    #Limita o erro acumulado ao máximo DutyCycle
    ErrAccW = np.clip(ErrAccW, -0.2, 0.2)

    #Cálculo do sinal intermédio diferencial
    Udif = ErrAccW*Kwi + ErrW*Kw

    Udif = np.clip(Udif, -65535, 65535) #Limita o
sinal

    #Cálculo dos DutyCycles
    DutyCycle1 = int(Ucom - Udif)
    DutyCycle2 = int(Ucom + Udif)

    #Limita os valores ao máximo
    DutyCycle1 = np.clip(DutyCycle1, -65535, 65535)
    DutyCycle2 = np.clip(DutyCycle2, -65535, 65535)

    #Cria o sinal PWM enviado para o driver de motor
    pwm11.duty_cycle = DutyCycle1*(DutyCycle1>30000)
    pwm12.duty_cycle = -DutyCycle1*(DutyCycle1<=
30000)

    pwm21.duty_cycle = DutyCycle2*(DutyCycle2>30000)
    pwm22.duty_cycle = -DutyCycle2*(DutyCycle2<=
30000)

    #Cálculo da média dos valores amostrados
```

Apêndice C – Configuração do sistema do barco autónomo

```
GyroMed = GyroMed + Gyro/50
AclMed[0] = AclMed[0] + acl[0]/50
AclMed[1] = AclMed[1] + acl[1]/50
AclMed[2] = AclMed[2] + acl[2]/50

#Periodicamente é realizada a leitura do rádio
if ticks_diff(Tatual, Tanterior1) > 1000:
    Tanterior1 = Tatual
    listen()
    if data[0] == 0x80:
        ControloAutonomo = False
        if data[0] != 0x40: #Caso seja uma mensagem
diferente, saí do controlo autónomo
            break
    #Verifica se chegou ao local
    if DistanceBetween(GPS[0], GPS[1], CoordGPS[i],
CoordGPS[i+1]) < dist:
        #Atualiza as coordenadas para o próximo local
        i = (i+2)%6
        if CoordGPS[i] == 0.0:
            i=0
        #Desliga os motores
        pwm11.duty_cycle = 0
        pwm12.duty_cycle = 0
        pwm21.duty_cycle = 0
        pwm22.duty_cycle = 0
        time.sleep(1)
```

Figura C.15 – Código do controlo autónomo do barco.

C.4 Programa final do barco autónomo

Ao reunir todas as partes anteriormente descritas, criou-se o programa do barco na sua totalidade, como mostra o código da Figura C.16.

```
import board
```

Apêndice C – Configuração do sistema do barco autônomo

```
import busio
import time
import supervisor
import pwmio
import microcontroller
import adafruit_gps
import adafruit_lsm9ds1
import storage
import struct
from digitalio import DigitalInOut, Direction
import adafruit_ina219
from ulab import numpy as np

def ticks_diff(ticks1, ticks2):
    _TICKS_PERIOD = const(1<<29)
    _TICKS_MAX = const(_TICKS_PERIOD-1)
    _TICKS_HALFPERIOD = const(_TICKS_PERIOD//2)
    #Compute the signed difference between two ticks
    values, assuming that they are within 2**28 ticks
    diff = (ticks1 - ticks2) & _TICKS_MAX
    diff = ((diff + _TICKS_HALFPERIOD) & _TICKS_MAX) -
    _TICKS_HALFPERIOD
    return diff

#Amostragem dos dados do GPS
def SampleGPS():
    my_gps.update()
    global GPS
    if my_gps.latitude != None:
        GPS = [my_gps.latitude, my_gps.longitude]

#Amostragem dos dados do 9DOF
```

Apêndice C – Configuração do sistema do barco autônomo

```
def Sample9DOF(GyroMed, AclMed):
    global NineDOF
    mag[0],mag[1],mag[2] = sensor9DOF.magnetic
    NineDOF = [AclMed[0], AclMed[1], AclMed[2], 0, 0,
GyroMed, mag[0]*100, mag[1]*100, mag[2]*100]

#Amostragem da velocidade de rotação + aceleração
def Sample6DOF():
    global Gyro
    global acl
    gyro[0],gyro[1],gyro[2] = sensor9DOF.gyro
    acl[0],acl[1],acl[2] = sensor9DOF.acceleration
    Gyro = gyro[2] - GyroBias

#Cálculo da distância entre duas coordenadas GPS
def DistanceBetween(lat1, long1, lat2, long2):
    delta = np.radians(long1-long2)
    sdlong = np.sin(delta)
    cdlong = np.cos(delta)
    lat1 = np.radians(lat1)
    lat2 = np.radians(lat2)
    slat1 = np.sin(lat1)
    clat1 = np.cos(lat1)
    slat2 = np.sin(lat2)
    clat2 = np.cos(lat2)
    delta = (clat1 * slat2) - (slat1 * clat2 * cdlong)
    delta = delta**2
    delta += (clat2 * sdlong)**2
    delta = np.sqrt(delta)
    denom = (slat1 * slat2) + (clat1 * clat2 * cdlong)
    delta = np.arctan2(delta, denom)
```

Apêndice C – Configuração do sistema do barco autônomo

```
    return delta * 6371009 #distância = ângulo em
radianos vezes o raio da terra

#Cálculo do ângulo de direção entre duas coordenadas GPS
def CourseTo(lat1, long1, lat2, long2):
    #Referenciado com Norte a 0° e Oeste a -90°
    dlon = np.radians(long2-long1)
    lat1 = np.radians(lat1)
    lat2 = np.radians(lat2)
    a1 = np.sin(dlon) * np.cos(lat2)
    a2 = np.sin(lat1) * np.cos(lat2) * np.cos(dlon)
    a2 = np.cos(lat1) * np.sin(lat2) - a2
    a2 = np.arctan2(a1, a2)
    return np.degrees(a2)

#Estimação do valor da direção atual
def EstYaw(Accel, Mag):
    #Para estimar o ângulo da direção do barco
    (compasso) utiliza-se o magnetômetro e os outros ângulos
    de rotação

    #Assume-se que a densidade de fluxo magnético única
    a ser aplicada no magnetômetro foi a terrestre e que

    #o eixo dos Xs esta alinhado na direção do barco e o
    eixo dos Zs esta alinhado com a gravidade da terra

    #Para estimar os outros ângulos de rotação utiliza-
    se o acelerômetro

    #Assume-se que a única aceleração foi gravítica e
    que

    #o eixo dos Zs do acelerômetro para dar 0° de
    rotação nos eixos, tem de estar alinhado com a gravidade
    da terra

    #Esta estimação tem erros devido a aplicação de
    outras forças no barco, ruído e calibração

    global roll
    global pitch
```

Apêndice C – Configuração do sistema do barco autónomo

```
roll = np.arctan2(Accel[0], Accel[2])

pitch = np.arctan2(Accel[1],
np.sqrt(Accel[0]*Accel[0] + Accel[2]*Accel[2]))

measuredYaw = -np.degrees(np.arctan2(np.sin(roll)
*Mag[2] + np.cos(roll)*Mag[0], np.cos(pitch)*Mag[1] +
np.sin(roll)*np.sin(pitch)*Mag[0] - np.cos(roll)*
np.sin(pitch)*Mag[2]))

return measuredYaw

#Escrita da configuração no rádio nRF905
def writeConfig(CH_NO, HFREQ_PLL, PA_PWR, RX_RED_PWR,
AUTO_RETRAN, TX_AFW, RX_AFW, RX_PW, TX_PW, address1,
address2, address3, address4, CRC_MODE, CRC_EN, XOF,
UP_CLK_EN, UP_CLK_FREQ):

    CSN.value = 0

    SpiRF905.write(bytes([0x00])) #comando escrever no
registo de configuração

    SpiRF905.write(bytes([(CH_NO%256)])) #frequência
central 1

    SpiRF905.write(bytes([AUTO_RETRAN*32 + RX_RED_PWR*16
+ PA_PWR*4 + HFREQ_PLL*2 + CH_NO//256]))

    SpiRF905.write(bytes([TX_AFW*16 + RX_AFW])) #tamanho
do endereço

    SpiRF905.write(bytes([RX_PW])) #tamanho payload RX
    SpiRF905.write(bytes([TX_PW])) #tamanho payload TX
    SpiRF905.write(bytes([address1, address2, address3,
address4]))

    SpiRF905.write(bytes([CRC_MODE*128 + CRC_EN*64 +
XOF*8 + UP_CLK_EN*4 + UP_CLK_FREQ]))

    CSN.value = 1

#Envio de mensagem no rádio nRF905
def sendstr(message, address1, address2, address3,
address4):

    TRX_CE.value = 0

    TXEN.value = 1
```

Apêndice C – Configuração do sistema do barco autónomo

```
time.sleep(0.04)
CSN.value = 0
SpiRF905.write(bytes([0x20])) #comando para escrever
mensagem no rádio
SpiRF905.write(bytes(message))
CSN.value = 1
time.sleep(0.04)
CSN.value = 0
SpiRF905.write(bytes([0x22])) #comando para escrever
endereço no rádio
SpiRF905.write(bytes([address1, address2, address3,
address4]))
CSN.value = 1
TRX_CE.value = 1
time.sleep(0.04)
TRX_CE.value = 0

#Receção da mensagem no rádio nRF905
def listen():
    TXEN.value = 0
    TRX_CE.value = 1
    global data
    if DR.value != 0:
        time.sleep(0.04)
        TRX_CE.value = 0
        CSN.value = 0
        SpiRF905.write(bytes([0x24]))
        SpiRF905.readinto(data)
        CSN.value = 1
        time.sleep(0.04)
        TRX_CE.value = 1

if __name__ == '__main__': #Código principal
```

Apêndice C – Configuração do sistema do barco autônomo

```
#Inicialização dos sinais PWM
pwm11 = pwmio.PWMOut(board.GP14, duty_cycle = 0,
frequency = 1000)
pwm12 = pwmio.PWMOut(board.GP15, duty_cycle = 0,
frequency = 1000)
pwm21 = pwmio.PWMOut(board.GP20, duty_cycle = 0,
frequency = 1000)
pwm22 = pwmio.PWMOut(board.GP17, duty_cycle = 0,
frequency = 1000)

#Inicialização do GPS
gps_serial = busio.UART(board.GP4, board.GP5,
baudrate=9600)

my_gps = adafruit_gps.GPS(gps_serial)

#Inicialização do 9DOF
i2c = busio.I2C(board.GP3, board.GP2)
sensor9DOF = adafruit_lsm9ds1.LSM9DS1_I2C(i2c)

#Sensor de corrente para análise dos parâmetros
ina219a = adafruit_ina219.INA219(i2c, 0x40)
ina219b = adafruit_ina219.INA219(i2c, 0x41)

#Inicialização do rádio
SpiRF905 = busio.SPI(board.GP10, board.GP11,
board.GP12)

while not SpiRF905.try_lock():
    pass

SpiRF905.configure(baudrate=106000)
CSN = DigitalInOut(board.GP13)
CSN.direction = Direction.OUTPUT
CSN.value = 1

DR = DigitalInOut(board.GP9)
DR.direction = Direction.INPUT

TRX_CE = DigitalInOut(board.GP8)
TRX_CE.direction = Direction.OUTPUT

PWR = DigitalInOut(board.GP7)
```

Apêndice C – Configuração do sistema do barco autónomo

```
PWR.direction = Direction.OUTPUT
PWR.value = 1
TXEN = DigitalInOut(board.GP6)
TXEN.direction = Direction.OUTPUT
TXEN.value = 1
writeConfig(116, 0, 3, 0, 0, 4, 4, 32, 32, 0xA2,
0xA2, 0xA2, 0xA2, 1, 1, 3, 0, 0)
time.sleep(0.1)
data = bytearray(32) #Memória da mensagem recebida
#Inicialização das variáveis
ControloAutonomo = False
GPS = [0.0, 0.0]
NineDOF = np.array([0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0])
NineDOFCal = np.array([0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0])
dist=10
CoordGPS = np.array([32.644942, -16.912536,
32.644905, -16.912660, 32.644684, -16.912638, 32.644612,
-16.912885, 32.644402, -16.912980])
i=0
#Matriz correção + calibração
SoftIronCal = np.array([[1.021, 0.072, 0.007],[0.072,
0.988, 0.005],[0.007, 0.005, 0.996]])
NineDOFBias = [0.04, 0.6, 0.2, -0.0047, 0.008, 0.025,
10.9, 11.17,1.43]
GyroBias = 0.0247
#Inicialização das variáveis intermédia de controlo
ErrAccW = 0.0
v=0.0
Gyro = 0.0
GyroMed = 0.0
AclMed = [0.0, 0.0, 9.8]
#Espera até receber sinal GPS para continuar o código
```

Apêndice C – Configuração do sistema do barco autónomo

```
while GPS == [0.0, 0.0]:
    SampleGPS()
    time.sleep(0.1)

#Váriaveis de controlo
Kp = -0.02
Kd = -0.005*180/3.1416
Kw = 26214
Kwi = 262140

#Inicialização dos temporizadores
Tanterior = supervisor.ticks_ms()
Tanterior1 = Tanterior
Tatual = Tanterior

#Inicialização das variáveis do 9DOF
mag = [0,0,0]
acl = [0,0,0]
gyro = [0,0,0]
roll = 0
pitch = 0

while True:
    listen() #Escuta o rádio
    Tatual = supervisor.ticks_ms() #Amostra tempo
    #Receção de mensagem de controlo manual
    while data[0] == 0x80:
        ControloAutonomo = False
        Tatual = supervisor.ticks_ms() #Amostra tempo
        if ticks_diff(Tatual, Tanterior) > 5000: #De
5 em 5 segundos limpa o registo do rádio
            Tanterior = Tatual
            data[1] = 0x80
            data[2] = 0x80
            data[3] = 0x80
            data[4] = 0x80
```

Apêndice C – Configuração do sistema do barco autónomo

```
#Desabilita a escrita no microcontrolador
storage.remount("/", readonly = True)
#limpa registo
PWR.value = 0
time.sleep(0.1)
PWR.value = 1
time.sleep(0.1)

#Através dos valores da mensagem cria o sinal
PWM enviado para o driver do motor
    pwm11.duty_cycle = ((data[3]-128)*510 +
data[4])*(data[3]>170)
    pwm12.duty_cycle = ((128-data[3])*510 + (255-
data[4]))*(data[3]<80)
    pwm21.duty_cycle = ((data[1]-128)*510 +
data[2])*(data[1]>170)
    pwm22.duty_cycle = ((128-data[1])*510 + (255-
data[2]))*(data[1]<80)

listen() #Escuta o rádio
if data[0] != 0x80:
    #limpa registo e desliga motores
    PWR.value = 0
    time.sleep(0.1)
    PWR.value = 1
    time.sleep(0.1)
    pwm11.duty_cycle = 0
    pwm12.duty_cycle = 0
    pwm21.duty_cycle = 0
    pwm22.duty_cycle = 0

#Receção da mensagem de controlo autónomo
if data[0] == 0x40 and ControloAutonomo == False:
    ControloAutonomo = True
    #Inicialização dos temporizadores
    T anterior = supervisor.ticks_ms()
```

Apêndice C – Configuração do sistema do barco autônomo

```
Tanterior1 = Tanterior
Tatual = Tanterior

#Habilita a escrita no microcontrolador
storage.remount("/", readonly = False)

#Receção da mensagem de nova trajetória
if data[0] == 0x20:
    for j in range(6): #Preenche o array
        coordenada a coordenada
            CoordGPS[j] = struct.unpack('d',
bytearray([0,0,0,0] + [data[8+j*4]] + [data[9+j*4]] +
[data[10+j*4]] + [data[11+j*4]])) #Converte conjunto de
bytes para float

            CoordGPS = np.array(CoordGPS) #Transforma em
um array onde se possa realizar operações matemáticas

            while ControlAuto == True: #Loop do controle
autônomo

                SampleGPS() #Amostra GPS

                Sample9DOF(GyroMed, AclMed) #Amostra 9DOF

                #Realiza a calibração dos dados do 9DOF

                NineDOFCal = [NineDOF[0] - NineDOFBias[0],
NineDOF[1] - NineDOFBias[1], NineDOF[2] - NineDOFBias[2],
NineDOF[3] - NineDOFBias[3], NineDOF[4] - NineDOFBias[4],
GyroMed, NineDOF[6] - NineDOFBias[6], NineDOF[7] -
NineDOFBias[7], NineDOF[8] - NineDOFBias[8]]

                NineDOFCal[6:] =
[NineDOFCal[6]*SoftIronCal[0][0] +
NineDOFCal[7]*SoftIronCal[0][1] +
NineDOFCal[8]*SoftIronCal[0][2],
NineDOFCal[6]*SoftIronCal[1][0] +
NineDOFCal[7]*SoftIronCal[1][1] +
NineDOFCal[8]*SoftIronCal[1][2],
NineDOFCal[6]*SoftIronCal[2][0] +
NineDOFCal[7]*SoftIronCal[2][1] +
NineDOFCal[8]*SoftIronCal[2][2]]

                Angref = CourseTo(GPS[0], GPS[1],
CoordGPS[i], CoordGPS[i+1]) #Cálculo da direção de
referência
```

Apêndice C – Configuração do sistema do barco autónomo

```
AngAtual = EstYaw(NineDOFCal[:3],
NineDOFCal[6:]) #Cálculo da direção atual
#Cálculo da diferença entre os ângulos
ErrAng = Angref - AngAtual
if ErrAng > 180:
    ErrAng = ErrAng - 360
if ErrAng < -180:
    ErrAng = ErrAng + 360
try:
    with open("dados.txt", "a") as f:
        #Escrita dos dados amostrados para
análise
        f.write(str('%0.6f' % CoordGPS[i]) +
', ' + str('%0.6f' % CoordGPS[i+1]) + ', ' + str('%0.6f' %
GPS[0]) + ', ' + str('%0.6f' % GPS[1]) + ', ' + str(Angref)
+ ', ' + str(AngAtual) + ', ' + str(Tatual) + '\n')
    except:
        pass
Wref = ErrAng*Kp - Kd*GyroMed #Cálculo da
velocidade de rotação de referência
Wref = np.clip(Wref, -0.8, 0.8) #Limita a
referência à sua máxima velocidade de rotação
vref = 0.5*(1 - abs(ErrAng)/90) #Cálculo da
velocidade de referência
Ucom = vref*75000 #Cálculo do sinal
intermédio comum
Ucom = np.clip(Ucom, -50000, 50000) #Limita o
sinal
#Reset dos valores médios
GyroMed = 0.0
AclMed = [0.0, 0.0, 0.0]
for k in range(50): #Loop interior do
controlo autónomo
    #Cálculo da diferença de tempo no loop
interno
```

Apêndice C – Configuração do sistema do barco autónomo

```
Tatual = supervisor.ticks_ms() #amostra
tempo

dt = float(ticks_diff(Tatual,
Tanterior))/1000

Tanterior = Tatual
Sample6DOF() #Amostragem do giroscópio +
acelerómetro

#Cálculo do erro da velocidade de rotação
ErrW = Wref - Gyro

#Cálculo do erro acumulado da velocidade
de rotação

ErrAccW = ErrAccW + ErrW*dt

#Limita o erro acumulado ao máximo
DutyCycle

ErrAccW = np.clip(ErrAccW, -0.2, 0.2)
#Cálculo do sinal intermédio diferencial
Udif = ErrAccW*Kwi + ErrW*Kw
Udif = np.clip(Udif, -65535, 65535)
#Limita o sinal

#Cálculo dos DutyCycles
DutyCycle1 = int(Ucom - Udif)
DutyCycle2 = int(Ucom + Udif)
#Limita os valores ao máximo
DutyCycle1 = np.clip(DutyCycle1, -65535,
65535)
DutyCycle2 = np.clip(DutyCycle2, -65535,
65535)

#Cria o sinal PWM enviado para o driver
de motor

pwm11.duty_cycle =
DutyCycle1*(DutyCycle1>30000)
pwm12.duty_cycle = -
DutyCycle1*(DutyCycle1<-30000)
pwm21.duty_cycle =
DutyCycle2*(DutyCycle2>30000)
```

Apêndice C – Configuração do sistema do barco autónomo

```
        pwm22.duty_cycle = -
DutyCycle2*(DutyCycle2<-30000)

        #Cálculo da média dos valores amostrados
        GyroMed = GyroMed + Gyro/50
        AclMed[0] = AclMed[0] + acl[0]/50
        AclMed[1] = AclMed[1] + acl[1]/50
        AclMed[2] = AclMed[2] + acl[2]/50

        #Periodicamente é realizada a leitura do
rádio

        if ticks_diff(Tatual, Tanterior1) > 1000:
            Tanterior1 = Tatual
            listen()
            if data[0] == 0x80:
                ControloAutonomo = False
                if data[0] != 0x40: #Caso seja uma
mensagem diferente, sai do controlo autónomo
                    break

            if DistanceBetween(GPS[0], GPS[1],
CoordGPS[i], CoordGPS[i+1]) < dist: #Verifica se chegou
ao local

                #Atualiza as coordenadas para o próximo
local

                i = (i+2)%6
                if CoordGPS[i] == 0.0:
                    i=0

                #Desliga os motores
                pwm11.duty_cycle = 0
                pwm12.duty_cycle = 0
                pwm21.duty_cycle = 0
                pwm22.duty_cycle = 0
                time.sleep(1)

                #Após 3 segundos sem receber mensagem nova limpa
o registo

                if ticks_diff(Tatual, Tanterior) > 3000:
```

Apêndice C – Configuração do sistema do barco autónomo

```
Tanterior = Tactual
PWR.value = 0
time.sleep(0.1)
PWR.value = 1
time.sleep(0.1)
time.sleep(0.1)
```

Figura C.16 – Código global do barco autónomo.

Também se apresenta o circuito global do barco na Figura C.17, para completar a configuração prática.

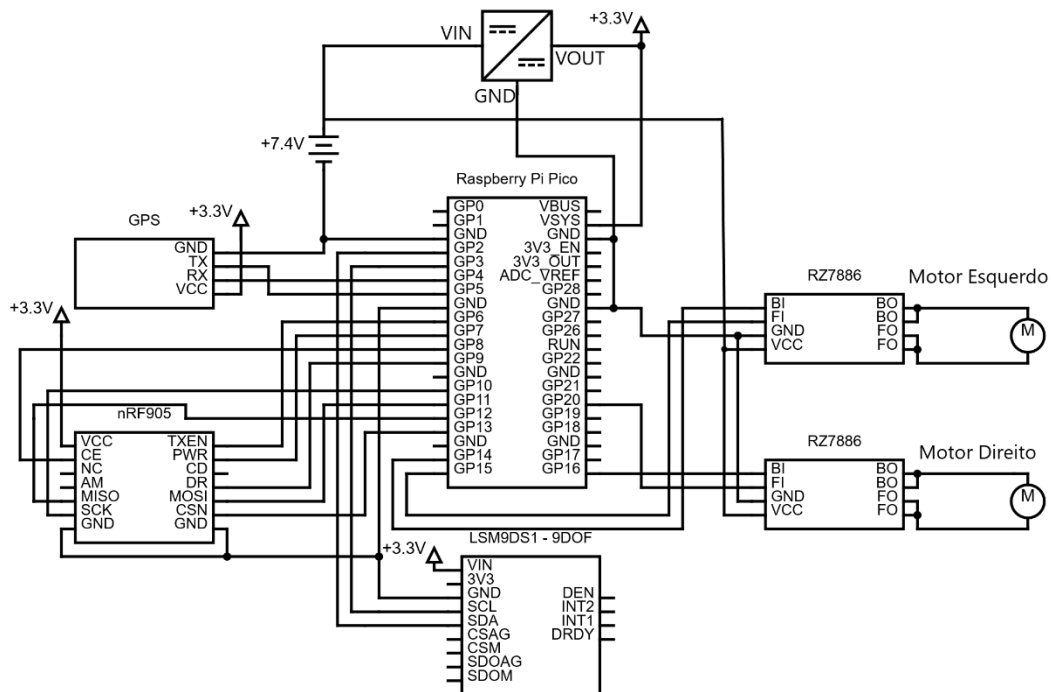


Figura C.17 – Circuito global equivalente do barco autónomo.

Tendo apresentado o código do barco autónomo, procede-se para a apresentação do código das variáveis de simulação.

C.5 Simulação do barco autónomo no MATLAB/SIMULINK

Para realizar a simulação do barco autónomo no MATLAB/SIMULINK, atribuíram-se os valores das constantes dos controladores apresentados na Tabela 4.1. Também se atribuíram as constantes calculadas no Apêndice B, ao

Apêndice C – Configuração do sistema do barco autônomo

programa de simulação apresentado, através do código apresentado na Figura C.18 em MATLAB.

```
%Parâmetros de entrada
AngRef=180;
vRef=0.5;
Vbat=7.4;

%Constantes dos Controladores
Kp=0.02; %(em °)
Kd=0.005*180/pi; %(em rads)
Kw=0.5; %(em rads)
Kwi=4; %(em rads)

%Constantes do modelo Motor DC
Kb1=0.02;
Kb2=0.02;
La1=0.000480;
La2=0.000480;
Ra1=1.5;
Ra2=1.5;
Kt1=0.02;
Kt2=0.02;

%Constantes do modelo Yoerger
B1=75000;
B2=75000;
a1=0.04;
a2=0.04;
Ct11=2e-5;
Ct12=1e-6;
Ct21=3e-5;
Ct22=1e-6;

%Constantes da conversão das forcas nos motores para
força aplicada em comum e aplicada perpendicularmente
C1=0.65;
```

Apêndice C – Configuração do sistema do barco autônomo

```

C2=0.76;

%Constantes da conversão das forças para velocidade e
rotação

m=1.8;
Cv=5.3;
Cw=0.63;
    
```

Figura C.18 – Código de simulação do barco autônomo.

Também se mostra o diagrama de blocos construído no SIMULINK/MATLAB para completar o programa de simulação do barco autônomo, como mostra a Figura C.19.

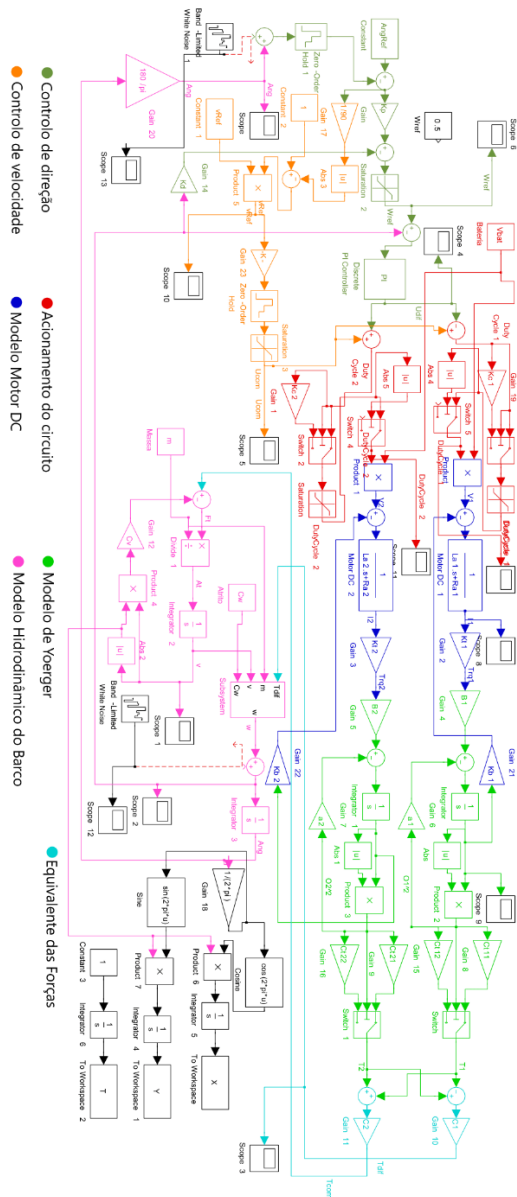


Figura C.19 – Diagrama de blocos para simulação do barco autônomo.

Apêndice D

VISUALIZAÇÃO DA REAÇÃO

DO BARCO

Neste apêndice apresentam-se os diversos vídeos realizados para mostrar o funcionamento do barco autônomo.

D.1 Vídeos do controlo manual do barco

Testa-se cada motor por si, transmitindo um sinal de comando de propulsão para a frente e para trás, como mostra o vídeo na seguinte hiperligação: https://youtube.com/shorts/a_Fim2D8Kq

Testam-se ambos os motores ao mesmo tempo, como mostra a seguinte hiperligação: <https://youtube.com/shorts/01Slw37xNWQ>

D.2 Vídeos de reação do sistema de controlo

Neste vídeo visualiza-se a reação do sistema de controlo do barco a funcionar com μ_{com} igual a zero. O vídeo está na seguinte hiperligação: <https://youtu.be/xXYiaE4-GAkh>

Neste teste regista-se a reação do barco quando está preso, existindo uma perturbação constante no sistema, como mostra a seguinte hiperligação: <https://youtu.be/brjdgXIK3cc>

Neste teste criam-se perturbações no barco e verifica-se como ele reage para corrigir as perturbações, como mostra a seguinte hiperligação: <https://youtube.com/shorts/R9TfwocVam0>

D.3 Vídeos da trajetória pré-definida do barco autónomo num lago

Neste vídeo observa-se a realização da trajetória programada num lago através da seguinte hiperligação: <https://youtube.com/shorts/mVmNunZjRxo>

Neste teste transmite-se uma nova trajetória para o barco enquanto ele percorre a trajetória anterior, como mostra a seguinte hiperligação: <https://youtube.com/shorts/78u8SRYsOR0>

D.4 Vídeo da trajetória pré-definida do barco autónomo no mar

Neste vídeo observa-se uma trajetória pré-definida no mar com perturbações, ruído nos sensores, ondas e múltiplos pontos. O vídeo mostra o barco a percorrer a trajetória como exhibe a seguinte hiperligação: <https://youtu.be/cGuw3h7CUIY>