

Modelação, Controlo e Simulação de uma Estação Marítima Autónoma

JOSÉ PEDRO DE FIGUEIREDO TAVARES

outubro de 2023

POLITÉCNICO DO PORTO
INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

Modelação, Controlo e Simulação de uma Estação Marítima Autónoma

José Pedro de Figueiredo Tavares

Mestrado em Engenharia Electrotécnica e de Computadores
Área de Especialização em Sistemas Autónomos

ISEP INSTITUTO SUPERIOR
DE ENGENHARIA DO PORTO

DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA
Instituto Superior de Engenharia do Porto

Outubro, 2023

*Esta dissertação satisfaz, parcialmente, os requisitos que constam da Ficha de
Unidade Curricular de Tese/Dissertação, do 2.º ano, do Mestrado em
Engenharia Electrotécnica e de Computadores, Área de Especialização em
Sistemas Autónomos.*

Candidato: José Pedro de Figueiredo Tavares, N.º 1160624,
1160624@isep.ipp.pt

Orientação Científica: Maria Benedita Campos Neves Malheiro,
mbm@isep.ipp.pt

Coorientação Científica: Pedro Barbosa Guedes, pbg@isep.ipp.pt

ISEP INSTITUTO SUPERIOR
DE ENGENHARIA DO PORTO

DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA
Instituto Superior de Engenharia do Porto
Rua Dr. António Bernardino de Almeida, 431, 4200-072 Porto

Outubro, 2023

Aos meus avós...

Agradecimentos

A todos que até hoje, em algum momento, passaram pelo meu percurso, que me fizeram aprender, crescer e tornar-me na pessoa que sou hoje, quero desde já expressar o meu agradecimento.

Em especial, aos meus orientadores Benedita Malheiro e Pedro Guedes, pela oportunidade de desenvolver esta dissertação, por todo o apoio, paciência e disponibilidade que tiveram comigo.

A todos os professores e colegas do LSA e ISEP que ao longo destes anos tanto me ensinaram.

Aos meus pais, irmã, família e amigos que sempre acreditaram em mim, me motivaram, me apoiaram e tornaram este objetivo possível.

Obrigado a todos!

Por fim, dedico este trabalho aos meus avós que tanto orgulho têm em mim, especialmente à minha avó Armanda que desde sempre me acompanhou no meu percurso escolar e infelizmente não teve a oportunidade de ver este trabalho concluído.

Resumo

A exploração dos oceanos através de plataformas do tipo *Autonomous Underwater Vehicle* (AUV) e *Autonomous Surface Vehicle* (ASV) constitui uma forma segura e eficiente de caracterizar, mapear, monitorizar e, consequentemente, proteger estas vastas áreas remotas do planeta.

O objetivo desta dissertação, realizada no Laboratório de Sistemas Autónomos (LSA) do Instituto Superior de Engenharia do Porto (ISEP), foi conceber e avaliar, recorrendo a simulação, uma estação autónoma marítima para a realização de missões de duração variável em áreas restritas. A estação autónoma deve ser capaz de permanecer num local de interesse por um período de tempo configurável sem estar ancorada (*station keeping*) e ser energeticamente eficiente.

Em primeiro lugar, efetuou-se um levantamento exaustivo dos *Autonomous Surface Vehicles* com diferentes tipos de propulsão existentes, seguido de um estudo dos princípios físicos da navegação à vela. Decidiu-se então conceber um ASV propulsionado pelo vento, equipado com um conjunto vela-cauda de asas rígidas. Esta solução, que permite controlar o ângulo de ataque da vela asa com o vento através da cauda, tem um baixo consumo energético.

A plataforma idealizada foi modelada no SolidWorks e, posteriormente, controlada e simulada através do *Robot Operating System* (ROS) e do ambiente de simulação Gazebo. A correta simulação do comportamento do ASV no Gazebo envolveu a reutilização, alteração e criação de *plugins*. O sistema de controlo do ângulo de ataque da vela asa e da direção do veleiro recorreu a dois controladores do tipo Proporcional Derivativo (PD), um para atuar na cauda e outro no leme. O ASV simulado realiza missões definidas através de *waypoints*, incluindo *station keeping*.

Por fim, foram realizados testes para caracterizar e validar o comportamento físico e o controlo do veleiro. Os resultados obtidos são promissores: a estação marítima autónoma é capaz de navegar à vela até aos *waypoints* definidos e permanecer na área de interesse estabelecida.

Palavras-Chave: Estação marítima autónoma, vento, vela asa rígida, controlo, simulação

Abstract

The exploration of the oceans with *Autonomous Underwater Vehicle* (AUV) and *Autonomous Surface Vehicle* (ASV) platforms is a safe and efficient way of characterising, mapping, monitoring and, consequently, protecting these vast remote areas of the planet.

The aim of this dissertation, carried out at the Laboratório de Sistemas Autónomos (LSA) of the Instituto Superior de Engenharia do Porto (ISEP), was to design and evaluate, using simulation, an autonomous maritime station for carrying out missions of varying duration in restricted areas. The autonomous station must be able to remain in a place of interest for a configurable period of time without being anchored (station keeping) and be energy efficient.

Firstly, an exhaustive survey was carried out of ASV with different types of propulsion, followed by a study of the physical principles of sailing. Subsequently, the decision was to design a wind-propelled ASV, equipped with a rigid wing sail-tail assembly. This solution, which makes it possible to control the angle of attack of the wing sail with the wind through the tail, has low energy consumption.

The idealised platform was modelled in SolidWorks and then controlled and simulated using the *Robot Operating System* (ROS) and the Gazebo environment. In order to correctly simulate the behaviour of the ASV in the Gazebo, it was necessary to reuse, modify and create plugins. The system for controlling the angle of attack of the wing sail and the sailboat's direction used two Proportional Derivative (PD) controllers, one for the tail and the other for the rudder. The simulated ASV carries out missions defined through waypoints, including station keeping.

Finally, tests were carried out to characterise and validate the physical behaviour and control of the sailboat. The results obtained are promising: the autonomous maritime station is able to sail towards the defined waypoints and remain within the given area of interest.

Keywords: Autonomous maritime station, wind, rigid wingsail, control, simulation

Índice

Lista de Equações	xiii
Glossário	xv
Lista de Acrónimos	xix
Lista de Símbolos	xxv
1 Introdução	1
1.1 Contextualização	1
1.2 Motivação	1
1.3 Objetivos	2
1.4 Organização da Dissertação	2
2 <i>Autonomous Surface Vehicles</i>	3
2.1 Plataformas Propulsionadas por Vento	3
2.1.1 Atlantis	3
2.1.2 FASt	4
2.1.3 SailBuoy	6
2.1.4 Saildrone Explorer	6
2.1.5 SKB	8
2.1.6 Unmanned Sailboat Platform	9
2.2 Plataformas Propulsionadas por Motor	10
2.2.1 C-Enduro	10
2.2.2 FLEXUS	13
2.2.3 OASIS	14
2.2.4 ROAZ	15
2.2.5 Underactuated ASV	17
2.2.6 WAM-V USV16	18
2.2.7 Wivenhoe ASV	20
2.3 Plataformas Propulsionadas por Ondas	21
2.3.1 Wave Glider	21
2.3.2 Boia Robótica	22

2.4	Análise	23
2.5	Sumário	28
3	Navegação à Vela	29
3.1	Navegação	30
3.1.1	Popa	31
3.1.2	Bolina	33
3.2	Vela Asa	38
3.2.1	Vela Asa com Cauda	39
3.2.2	Vela Asa com <i>Canard</i>	40
3.3	Cascos	42
3.4	Sumário	42
4	Estação Marítima Autónoma	43
4.1	Enquadramento e Problema	43
4.2	Objetivos	44
4.3	Análise de Requisitos	44
4.4	Proposta de Solução	45
4.4.1	Casco	45
4.4.2	Vela Asa com Cauda	46
4.4.3	Ficha Técnica	50
4.5	Sumário	50
5	Simulação	53
5.1	Packages	53
5.2	Modelo do Veleiro	54
5.3	Controlo	54
5.3.1	Controlo da Vela Asa	54
5.3.2	Controlo do Leme	56
5.4	Algoritmo de Station Keeping	59
5.5	Sumário	61
6	Testes e Resultados	63
6.1	Alinhamento da Vela Asa ao Vento	63
6.2	Controlo da Vela Asa	65
6.3	Diagrama Polar	67
6.4	Controlo de Direção	67
6.5	<i>Station Keeping</i>	68
6.6	Sumário	70
7	Conclusão	73
7.1	Discussão de Resultados	73

7.2	Dificuldades	74
7.3	Trabalho Futuro	74
7.4	Sumário	74
Referências		76
Anexo A Mareações e Manobras		89
A.1	Mareações	89
A.2	Manobras	90
Anexo B Ambiente de Desenvolvimento e Simulação		93
B.1	ROS	93
B.2	Simuladores	94
B.2.1	Webots	94
B.2.2	Gazebo	94
B.2.3	Unreal Engine	94
B.2.4	Unity	95
B.2.5	Análise e Comparação	95
B.3	Ambiente de Simulação	96
B.3.1	Virtual RobotX	96
B.3.2	ASV Simulator	96
B.3.3	USVSim	96
B.3.4	Análise e Comparação	97
B.4	Plugins	97
B.4.1	LiftDragPlugin	97
B.4.2	Plugin flap_control	100
B.4.3	Plugin usv_control	100
Anexo C Código		101
C.1	Ficheiro URDF	101

Lista de Figuras

2.1	Atlantis <i>Autonomous Surface Vehicle</i> (ASV) [16]	4
2.2	FEUP <i>Autonomous Sailboat</i> (FASt) [19]	5
2.3	Manobra de <i>station keeping</i> implementada no FEUP <i>Autonomous Sailboat</i> (FASt) à volta dos <i>waypoints</i> definidos (pontos vermelhos) [19]	6
2.4	Sailbuoy [20]	6
2.5	Saildrone Explorer [24]	7
2.6	Diferentes configurações do Saildrone adaptado às características de cada missão	8
2.7	Protótipos desenvolvidos do <i>Station Keeping Buoy</i> (SKB) [30]	9
2.8	Unmanned Sailboat Plataform [31]	10
2.9	Arquitetura do Sistema de Controlo [31]	11
2.10	C-Enduro ASV [33]	11
2.11	FLEXUS ASV [35]	13
2.12	<i>Ocean Atmosphere Sensor Integration System</i> (OASIS) ASV num atrelado [37]	14
2.13	OASIS 2 e 3 durante os testes na Chincoteague Bay [38]	15
2.14	ROAZ [40]	16
2.15	ROAZ II [42]	17
2.16	Underactuated ASV em modo <i>station keeping</i> [12]	18
2.17	<i>Wave Adaptive Modular-Vessel</i> (WAM-V) <i>Unmanned Surface Vehicle</i> (USV)16 durante os testes de <i>station keeping</i> [45]	19
2.18	Wivenhoe ASV [50]	20
2.19	Wave Glider composto por flutuador (superfície) e <i>glider</i> (submerso) [51]	21
2.20	Wave Glider na missão do <i>Northwest Fisheries Science Center</i> [57]	22
2.21	Boia robótica composta por flutuador (superfície), <i>wave turbine</i> (submersa a meio) e o <i>wave glider</i> (submerso no fundo) [59]	23
3.1	Principais componentes de um barco à vela	29
3.2	Diferentes zonas do casco e os seus elementos [62]	30
3.3	Seis graus de liberdade de um barco [13]	30
3.4	Veleiro a navegar à popa [67]	32

3.5	Força de sustentação ($\vec{L}$), arrasto ($\vec{D}$) e resultante ($\vec{F}$) numa vela asa rígida ou patilhão	33
3.6	Representação das forças exercidas na vela e da força de propulsão e inclinação [8]	34
3.7	Representação da força de inclinação gerada na vela e de sustentação gerada no patilhão [72]	35
3.8	Força de sustentação e força de arrasto em função do ângulo de ataque [67]	35
3.9	Forças exercidas pelo vento e pela água num barco que navega com estabilidade	36
3.10	Estabilidade metacêntrica transversal [66]	37
3.11	Relação entre as variáveis que influenciam na estabilidade metacêntrica de um barco [66]	37
3.12	Forças presentes na vela asa rígida [11]	38
3.13	Perfis alares	39
3.14	Esboço de um conjunto vela asa rígida com cauda [61]	39
3.15	Forças e binários presentes no controlo do conjunto vela asa com cauda: cauda não atuada (superior), atuação da cauda (centro), conjunto vela asa e cauda em equilíbrio (inferior) [8]	40
3.16	Esboço de um conjunto vela asa rígida com <i>canard</i> [61]	41
3.17	Forças e binários presentes no conjunto vela asa com <i>canard</i> [76]	41
4.1	Estação marítima autónoma proposta	46
4.2	Dimensões do casco projetado	46
4.3	Perfil alar Eppler E473 [90]	47
4.4	Principais características aerodinâmicas do perfil alar Eppler E473 [90]	48
4.5	Dimensões do conjunto vela asa e cauda projetado	49
5.1	Diagrama de blocos da simulação	54
5.2	Diagrama do modelo do <i>robot</i> implementado	55
5.3	Estação Marítima Autónoma no Gazebo	55
5.4	Controlador PID da cauda	56
5.5	Fluxograma da função <code>MainSail_Control()</code>	57
5.6	Controlador PD do leme	58
5.7	Fluxograma da sub-rotina <code>Turning()</code>	58
5.8	Fluxograma da função <code>Navigation_Control()</code>	60
5.9	Esboço da manobra de <i>station keeping</i>	61
6.1	Alinhamento da vela asa ao vento sem controlo	64
6.2	Resposta temporal do alinhamento da vela asa para diferentes velocidades do vento	65

6.3	Resposta da vela asa para diferentes ângulos da cauda	66
6.4	Resposta temporal para diferentes ganhos do controlo implementado para o ângulo de ataque da vela asa	66
6.5	Resposta temporal para diferentes velocidades do vento do controlo implementado para o ângulo de ataque da vela asa	67
6.6	Diagrama polar da estação marítima modelada	68
6.7	Seguimento de <i>waypoints</i> no modo normal de navegação	69
6.8	Seguimento de <i>waypoints</i> no modo <i>turning</i> de navegação	69
6.9	Seguimento de <i>waypoints</i> utilizando os dois modos de navegação . .	70
6.10	Manobra de <i>station keeping</i> através do seguimento de <i>waypoints</i> . . .	71
A.1	Diferentes tipos de mareações [91]	90
A.2	Manobra viragem por davante [72]	90
A.3	Manobra viragem em roda [17]	90
B.1	Funcionamento do <code>LiftDragPlugin</code> para a vela	98
B.2	Linearização da $C_l(\alpha)$ [113].	99
B.3	Diagrama de comunicação do <i>plugin flap_control</i>	100
B.4	Diagrama de comunicação do <i>plugin usv_control</i>	100

Lista de Tabelas

2.1	Algoritmos de <i>station keeping</i> testados [32]	12
2.2	Média dos erros da posição e orientação do WAM-V USV16, com e sem controlador <i>feedforward</i> de vento [47]	19
2.3	Características relevantes dos ASV analisados	25
2.3	Características relevantes dos ASV analisados	26
2.4	Algoritmos implementados nos ASV analisados	27
4.1	Ficha técnica da estação marítima modelada	49
B.1	Características dos diferentes simuladores	95
B.2	Características dos diferentes ambientes de simulação	97

Lista de Equações

3.1	Força da Flutuação	31
3.2	Força do Peso	31
3.3	Força de arrasto na vela	32
3.4	Vento aparente	32
3.5	Força de arrasto no casco	32
3.6	Área do casco	33
3.7	Vento real	34
3.8	Estabilidade	36
3.9	Estabilidade metacêntrica	36
4.1	Número de Reynolds	47
4.2	Relação entre o perfil alar escolhido e uma vela tradicional	48
4.3	Força de <i>Payload</i>	50
4.4	Volume submerso	50
5.1	Cálculo do ângulo de ataque	56
5.2	Orientação desejada	57
5.3	Nova orientação desejada	59
5.4	Distância percorrida no bordo corrente	59
5.5	Distância percorrida	59
5.6	Cálculo dos <i>waypoints</i> para <i>station keeping</i>	59

Glossário

anemómetro

Instrumento que tem a função de avaliar a velocidade e a direção do vento

Automatic Identification System (AIS)

Sistema de rastreamento utilizado pelos barcos e navios [1]

BeiDou

BeiDou Navigation Satellite System é um sistema de *Global Positioning System* (GPS) chinês [2]

bordo de ataque

Parte do perfil alar que tem o primeiro contacto com o fluxo de ar [3]

bordo de fuga

Parte do perfil alar em que o fluxo de ar passa por último [3]

catamaran

Multi-casco composto por dois cascos

Communicating Sequential Process (CSP)

Linguagem utilizada para a descrição de padrões de interação [4]

controlador *backsteeping*

Um controlador *backstepping*, de uma forma simplificada, pretende controlar um sistema não linear utilizando vários subsistemas de controlo de forma recursiva [5]

controlador *feedforward*

O conceito básico de um controlador *feedforward* é medir as variáveis importantes que irão perturbar o sistema e tomar uma ação corretiva antes que estas atuem no sistema [6]

controlador *sliding mode*

É uma técnica de controlo adequada para sistemas com perturbações e incertezas. Este método permite uma abordagem robusta para alcançar a *performance* desejada do sistema [7]

corda média

Designado em inglês por *chord line*, é a linha imaginária entre o bordo de ataque e o bordo de fuga de um perfil alar [8]

glider

Expressão utilizada para algo que plana

gybing

Virada em roda ou *gybing*, diz respeito à manobra executada por um veleiro quando navega em "zig zag" com o vento a entrar pela popa ("a favor do vento")

Iridium

Comunicação satélite capaz de oferecer serviço de voz e dados em qualquer parte da superfície do globo. É operada pela Iridium Communications. [9]

length overall

Comprimento máximo do casco

mareações

Orientar a vela de acordo com a direção do vento

Microtransat Challenge

O *Microtransat Challenge* é uma corrida transatlântica para barcos autónomos [10]

middleware

Software que permite a comunicação entre diferentes sistemas

perfil alar

Designado em inglês por *airfoil*, é a forma da secção transversal obtida pela interseção de uma vela asa rígida com um plano perpendicular [11]

stall

Condição aerodinâmica em que o ângulo de ataque ideal do *airfoil* com o fluído é excedido, não permitindo que o fluído flua suavemente pelo *airfoil*, deixando de produzir força *lift* [3]

station keeping

Pela definição náutica é quando um barco mantém a sua posição relativa a outro barco ou quando este mantém-se fixo em relação a um objeto ou local [12]

tacking

Virada por davante ou *tacking*, diz respeito à manobra executada por um veleiro quando navega em "zig zag" com o vento a entrar pela proa ("contra o vento")

trimaran

Multi-casco composto por um casco principal no centro e dois cascos secundários dos lados

vento aparente

Velocidade e direção do vento medido a partir de um objeto em movimento, por exemplo, o vento sentido por uma pessoa parada na proa de um navio em movimento [13]

vento real

Velocidade e direção do vento medido numa plataforma fixa ao chão, por exemplo, o vento sentido por uma pessoa imóvel [13]

Lista de Acrónimos

ADCP	<i>Acoustic Doppler Current Profiler</i>
AIS	<i>Automatic Identification System</i>
ASV	<i>Autonomous Surface Vehicle</i>
AUV	<i>Autonomous Underwater Vehicle</i>
C&DH	<i>Command and Data Handling</i>
CFD	<i>Computational Fluid Dynamics</i>
CIT	<i>Center for Innovative Technology</i>
CPU	<i>Central Processing Unit</i>
CSP	<i>Communicating Sequential Process</i>
CTU	<i>Control Terminal Unit</i>
DARPA	<i>Defense Advanced Research Projects Agency</i>
DART	<i>Dynamic Animation and Robotics Toolkit</i>
DP	<i>Dynamic Positioning</i>
EKF	<i>Extended Kalman Filter</i>
FASt	<i>FEUP Autonomous Sailboat</i>
FEUP	<i>Faculdade de Engenharia da Universidade do Porto</i>
FPGA	<i>Field-Programmable Gate Array</i>
GN&C	<i>Guidance Navigation and Control</i>
GPS	<i>Global Positioning System</i>
GPU	<i>Graphics Processing Unit</i>
GSM	<i>Global System for Mobile communication</i>
HAB	<i>Harmful Algal Bloom</i>

HD	<i>High-Definition</i>
I&D	Investigação e Desenvolvimento
IMU	<i>Inertial Measurement Unit</i>
INESC TEC	Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência
INS	<i>Inertial Navigation System</i>
ISEP	Instituto Superior de Engenharia do Porto
JHU/APL	<i>Johns Hopkins University Applied Physics Laboratory</i>
L&R	<i>Launch and Recovery</i>
LOS	<i>Line Of Sight</i>
LSA	Laboratório de Sistemas Autónomos
LVFG	<i>Lyapunov Vector Field Guidance</i>
MCU	<i>Main Control Unit</i>
MEEC	Mestrado em Engenharia Electrotécnica e Computadores
MSS	<i>Marine System Simulator</i>
NATO	<i>North Atlantic Treaty Organization</i>
NDBC	<i>National Data Buoy Center</i>
NOAA	<i>National Oceanic and Atmospheric Administration</i>
OASIS	<i>Ocean Atmosphere Sensor Integration System</i>
ODE	<i>Open Dynamics Engine</i>
OGRE	<i>Object-Oriented Graphics Rendering Engine</i>
PD	Proporcional Derivativo
PI	Proporcional Integral
PID	Proporcional Integral Derivativo
PMEL	<i>Pacific Marine Environmental Laboratory</i>
POS	<i>Persistent Ocean Surveillance</i>
RAWFIE	<i>Road-, Air-, and Water- based Future Internet Experimentation</i>

RF	<i>Radio Frequency</i>
ROS	<i>Robot Operating System</i>
RTOS	<i>Real-Time Operating System</i>
SBC	<i>Single Board Computer</i>
SKB	<i>Station Keeping Buoy</i>
USV	<i>Unmanned Surface Vehicle</i>
VRX	<i>VirtualRobotX</i>
WAM-V	<i>Wave Adaptive Modular-Vessel</i>

Lista de Símbolos

Símbolo	Descrição	Unidades
$\vec{g}$	aceleração gravítica	m/s ²
h	altura do casco	m ³
$\overline{GM}$	altura metacêntrica	m
α	ângulo de ataque	°
A_{sail}	área da vela	m ²
A_{hull}	área do casco	m ²
B	centro de flutuação	m
G	centro gravítico	m
C_D	coeficiente de <i>drag</i>	-
C_L	coeficiente de <i>lift</i>	-
L_c	comprimento característica	m
l	comprimento do casco	m
B_x	coordenada ' x ' do barco	m
W_x	coordenada ' x ' do waypoint	m
B_y	coordenada ' y ' do barco	m
W_y	coordenada ' y ' do waypoint	m
$\vec{v}_c$	corrente marítima	m/s
$\vec{v}_{ac}$	corrente marítima aparente	m/s
ρ_{water}	densidade da água	kg/m ³
ρ_{air}	densidade do ar	kg/m ³
$\vec{F}$	força	N
$\vec{D}$	força de arrasto	N
$\vec{D}_{sail}$	força de arrasto na vela	N
$\vec{D}_{hull}$	força de arrasto no casco	N
$\vec{F}_B$	força de flutuação	N
$\vec{H}$	força de inclinação	N
$\vec{T}$	força de propulsão	N
$\vec{L}$	força de sustentação	N
$\vec{F}_W$	força peso	N
$\vec{F}_{wind}$	força vento	N
$\vec{F}_{water}$	força água	N

Símbolo	Descrição	Unidades
w	largura do casco	m
LOA	<i>length overall</i>	m
m	massa	kg
m_{boat}	massa barco	kg
M	metacentro	m
Re	número de Reynolds	-
ψ	orientação atual do barco	rad
$\vec{v}_b$	velocidade barco	m/s
$\vec{v}$	velocidade da vela	m/s
$\vec{v}_f$	velocidade do fluído	m/s
$\vec{v}_{aw}$	velocidade vento aparente	m/s
$\vec{v}_{tw}$	velocidade vento real	m/s
μ	viscosidade do meio	m ² /s
V_{hull}	volume do casco	m ³
∇	volume água deslocado	m ³
A_v	área vélica	m ²

Capítulo 1

Introdução

1.1 Contextualização

Cobrindo mais de metade da superfície do planeta Terra, o oceano constitui uma imensa fonte de recursos e desempenha um importante papel em diversas áreas. Por esta razão, ao longo dos anos, o interesse da comunidade científica, governos e indústria pelo mar tem vindo a crescer. Contudo, a sua exploração segura e eficiente ainda não é uma realidade. Neste contexto, os veículos autónomos para ambientes aquáticos, os *Autonomous Underwater Vehicle* (AUV) e os *Autonomous Surface Vehicle* (ASV), desempenham um papel crucial.

1.2 Motivação

Em contexto marítimo, a realização de missões de longa duração com veículos autónomos apresenta dois grandes desafios: a comunicação sem fios com o exterior e a autonomia energética. Sendo esta dissertação desenvolvida no Laboratório de Sistemas Autónomos (LSA), um centro de Investigação e Desenvolvimento (I&D) do Instituto Superior de Engenharia do Porto (ISEP) dedicado à investigação e desenvolvimento de sistemas autónomos com grande foco no ambiente aquático, surgiu a necessidade de se conceber e avaliar uma solução para realizar missões de duração variável em locais distintos no mar.

Os oceanos, devido à sua vastidão, permanecem em grande parte desconhecidos. A exploração do mar, que sempre me cativou pessoalmente, despertou ainda

mais interesse quando entrei para o ramo de Sistemas Autónomos do Mestrado em Engenharia Electrotécnica e Computadores (MEEC). Este projeto, realizado no Laboratório de Sistemas Autónomos (LSA), onde sou assistente de investigação há três anos, é pessoalmente motivador e permitirá acrescentar valor às atividades do laboratório.

1.3 Objetivos

Esta dissertação pretende avaliar a viabilidade de desenhar e controlar uma estação marítima autónoma, i.e., que permaneça temporariamente numa mesma área de interesse sem necessidade de fundear. A eficiência energética do veículo é um ponto que também deve ser considerado.

Para evitar a dependência de terceiros, optou-se por estudar a viabilidade da plataforma através de simulação. Assim, será necessário: *(i)* conceber uma plataforma com as características desejadas; *(ii)* criar o respetivo modelo através de um *software* de modelação 3D; *(iii)* desenvolver o sistema de controlo da estação marítima; e *(iv)* importar e controlar o modelo num simulador.

1.4 Organização da Dissertação

Esta dissertação está dividida em sete capítulos. O Capítulo 1 apresenta uma breve contextualização do projeto, motivação, os objetivos genéricos e a estrutura da dissertação. O Capítulo 2 resume e analisa várias plataformas marítimas com diferentes tipos de propulsão. O Capítulo 3 apresenta os componentes principais de um veleiro, os conceitos físicos associados e o seu comportamento durante a navegação. O Capítulo 4 aprofunda o problema, detalhando os objetivos e requisitos, e apresenta uma proposta de solução. O Capítulo 5 descreve o trabalho desenvolvido, desde a criação do modelo do veleiro, sistema de controlo até à simulação. O Capítulo 6 apresenta os testes realizados e os resultados obtidos. Por fim, o Capítulo 7 analisa os resultados, partilha as dificuldades, identifica as direções futuras e encerra com a conclusão final da dissertação.

Capítulo 2

Autonomous Surface Vehicles

Os *Autonomous Surface Vehicles* (ASV), também designados por *Unmanned Surface Vehicles* (USV), são embarcações não tripuladas capazes de navegar autonomamente à superfície da água [14]. Nos últimos anos, devido à grande evolução tecnológica, estes veículos tornaram-se uma solução viável para várias aplicações, principalmente para realizar missões longas, repetitivas ou perigosas para o ser humano. Dependendo do equipamento que transportam a bordo, os ASV são bastante versáteis, podendo ser utilizados em contextos científicos, comerciais ou militares.

2.1 Plataformas Propulsionadas por Vento

2.1.1 Atlantis

O Atlantis, apresentado na Figura 2.1, é um *catamaran* com 7,2 m de comprimento e 3 m de largura, baseado no Prindle-19 [15], que utiliza uma vela asa rígida em conjunto com uma *flying tail* como sistema de propulsão. Esta configuração permite à vela principal alcançar automaticamente o ângulo ideal com o vento, além de reduzir os efeitos das rajadas de vento [16]. Este ASV tem um computador para *Guidance Navigation and Control* (GN&C), encarregue das tarefas de estimação e controlo necessárias. Para estimar a sua posição e velocidade utiliza um GPS e um sistema *low-cost* com um acelerómetro e um magnetómetro de três eixos. Este veículo não tem nenhum sistema para gerar energia a bordo, o que faz com que aguente aproximadamente 15 h antes das baterias precisarem de ser carregadas [17].



Figura 2.1: Atlantis ASV [16]

Utiliza um controlador Proporcional Integral (PI) para realizar trajetórias. Este controlador usa os erros da direção e da posição para definir o ângulo do leme, de maneira a que o barco realize o percurso pretendido. O Atlantis apresenta três algoritmos de navegação por *waypoints*: mudança de *waypoint*, *tacking* e *station keeping*. O algoritmo de mudança de *waypoint* calcula uma trajetória em forma de arco com raio constante para suavizar a mudança de direção da embarcação na transição entre *waypoints* consecutivos. O algoritmo de *tacking* é automaticamente executado para atingir um *waypoint* à bolina. O algoritmo de *station keeping* utiliza o motor auxiliar para orientar o barco a favor da corrente e assim ficar alinhado com a posição (*waypoint*) de *station keeping*.

As simulações efetuadas demonstraram que, recorrendo apenas à vela, o Atlantis apresenta um erro de direção de $3,25^\circ$ e de posição de 6 cm em relação à trajetória definida. Estes resultados são bastante idênticos aos da simulação da realização do mesmo trajeto a motor.

2.1.2 FAST

Desenvolvido pela Faculdade de Engenharia da Universidade do Porto (FEUP), com o objetivo de participar no *Microtransat Challenge*, o FEUP *Autonomous Sailboat* (FAST) é um veleiro autónomo com 2,5 m de comprimento e 50 kg de massa [18]. Este ASV está equipado com: uma *Field-Programmable Gate Array* (FPGA), como sistema de computação; um microprocessador, para interface com os sensores e atuadores; um *router* Wi-Fi, para comunicação sem fios a curta distância; um *Global System for Mobile communication* (GSM) modem, para comunicações até poucos

km da costa; um *Iridium* modem, que permite a comunicação com o FAST em todo o globo; um recetor rádio, para controlo da embarcação em modo manual, caso seja necessário; um anemómetro, capaz de fornecer a direção e velocidade do vento; um recetor GPS; um inclinómetro; um sensor de luz ambiente; dois servomotores, que permitem controlar independentemente os dois lemes; e um motor DC, para controlar a vela. O veleiro, apresentado na Figura 2.2, além do equipamento anteriormente referido, possui um painel fotovoltaico 45 Wp que, aproveitando a energia solar, permite recarregar as baterias Li-Ion com capacidade de 194 Wh.



Figura 2.2: FEUP *Autonomous Sailboat* (FAST) [19]

Como a maior parte dos sistemas de navegação autónoma são desenvolvidos para ASV motorizados ou *Autonomous Underwater Vehicle* (AUV), não contemplam as restrições e manobras típicas da navegação à vela. Por esta razão, a equipa do FAST desenvolveu um sistema de programação de missões dedicado que permite definir missões através de mensagens de texto simples e intuitivas, incluindo *waypoints*, eventos ou ações [19]. Uma das ações implementadas é a de *station keeping*. Esta ação tem como parâmetros de entrada o raio da área de *station keeping*, sendo o centro desta área a posição atual do veleiro, e a condição de terminação, que pode ser um período de tempo ou um evento externo. Quando esta ação é executada, são definidos dois *waypoints* colineares com a posição atual do barco, perpendiculares à direção do vento real e separados por uma distância igual a 80% do diâmetro da área definida. A manobra de *station keeping* consiste na navegação contínua entre os *waypoints* definidos, realizando percursos em forma de oito, como se pode visualizar na Figura 2.3, até a condição de terminação ser verdadeira.

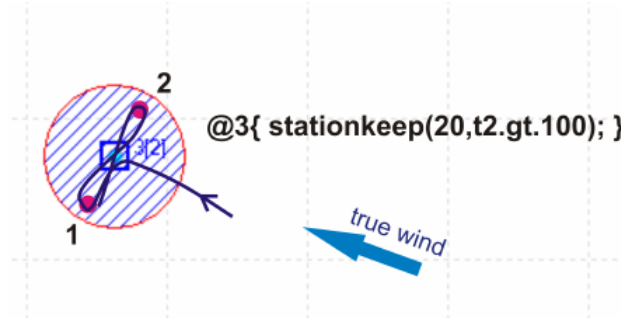


Figura 2.3: Manobra de *station keeping* implementada no FAST à volta dos *waypoints* definidos (pontos vermelhos) [19]

2.1.3 SailBuoy

Desenvolvido, produzido e vendido desde 2014 pela Offshore Sensing AS [20], o SailBuoy é um ASV criado para missões de longa duração. Capaz de carregar uma ampla variedade de sensores, de transmitir dados em tempo-real via satélite e aguentar fortes colisões e condições meteorológicas adversas (Beaufort Scale 9) [21], foi o primeiro ASV a cruzar o Atlântico. Navegando da costa este do Canadá, até à Noruega via Irlanda durante 118 dias e percorrido no total 7800 km [22]. Este veículo tem 2m de comprimento, carrega até 15 kg de *payload* e possui uma vela asa rígida, conseguindo navegar com ventos com velocidade entre 2 m/s e 20 m/s, atingindo uma velocidade entre 0,51 m/s e 1,03 m/s. O SailBuoy, apresentado na Figura 2.4, tem uma autonomia até um ano, produzindo a sua energia através de painéis fotovoltaicos colocados no casco.



Figura 2.4: Sailbuoy [20]

2.1.4 Saildrone Explorer

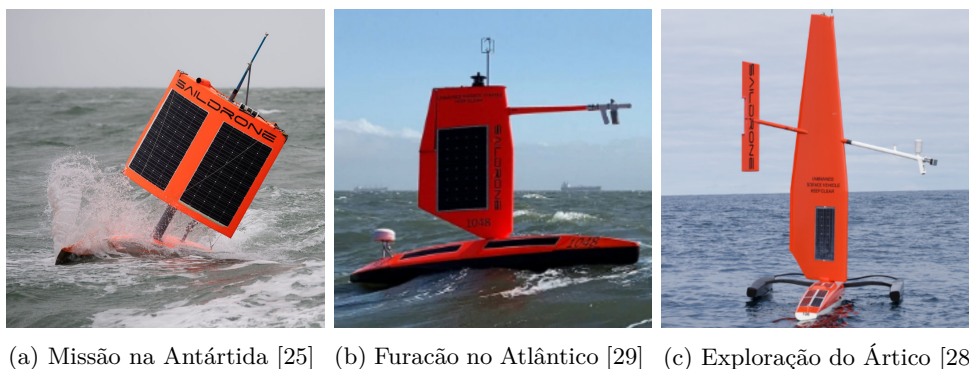
O Saildrone Explorer é um ASV equipado com uma vela asa rígida, com 5 m de altura, em conjunto com uma *tail flap*, como se poder ver na Figura 2.5. Este sistema simplifica bastante o *design* e controlo do veículo face a veleiros tradicionais [23]. O

Saildrone Explorer tem 7 m de comprimento e um patilhão de 2 m, alcançando uma velocidade média de cerca de 1,5 m/s [24]. Os painéis fotovoltaicos, colocados no casco e na vela, permitem gerar energia para toda a eletrônica do veículo [23].



Figura 2.5: Saildrone Explorer [24]

O Saildrone já participou em várias missões, tendo um dos seus maiores sucessos sido a circum-navegação à Antártida, em 2019 [25]. Nesta missão navegou durante 196 dias, tendo percorrido 22 000 km com temperaturas negativas, ondas de 15 m de comprimento, ventos com velocidade de 130 km/h e resistido a colisões com *icebergs*. Outro feito notável do Saildrone ocorreu em 2021 quando realizou uma missão de estudo dos fenómenos entre a atmosfera e o oceano, durante um furacão. O Saildrone resistiu a um furacão de categoria 4, com rajadas de vento de velocidade superior a 193 km/h e ondas com mais de 15 m, tendo enviado dados, incluindo vídeo, em tempo quase real durante toda a tempestade. No contexto de uma parceria entre a Saildrone e a *National Oceanic and Atmospheric Administration* (NOAA) [26], efetuou a exploração do ártico e do mar Bering, em 2015. Nesta missão, o ASV navegou mais de 7600 km ao longo de 97 dias, tendo atingido uma velocidade máxima de 3,7 m/s, e recolheu dados relativos à atmosfera, ao oceano e às espécies marinhas [27, 28, 23]. A Figura 2.6 apresenta as diferentes configurações, tanto da vela como do casco, utilizadas nas missões referidas. Na Figura 2.6a, obtida durante a missão na Antártida, o Saildrone tem uma vela quadrangular. Na Figura 2.6b da missão de recolha de dados durante o furacão, o ASV apresenta uma vela semelhante à normal mas cortada no extremo. Por último, na Figura 2.6c relativa à missão no ártico e no mar de Bering, o casco passou a ter dois estabilizadores.



(a) Missão na Antártida [25] (b) Furacão no Atlântico [29] (c) Exploração do Ártico [28]

Figura 2.6: Diferentes configurações do Saildrone adaptado às características de cada missão

2.1.5 SKB

O *Station Keeping Buoy* (SKB) foi desenvolvido pela *Johns Hopkins University Applied Physics Laboratory* (JHU/APL) [30] no âmbito do programa *Persistent Ocean Surveillance* (POS), patrocinado pela *Defense Advanced Research Projects Agency* (DARPA). O objetivo deste projeto era monitorizar o oceano através de um ASV, o mais autossustentável possível, que permaneça num determinado local por um longo período de tempo. Por esta razão, o SKB é um veleiro propulsionado pelo vento que utiliza painéis solares para produzir energia elétrica e, desta forma, recarregar as suas baterias. Foram construídas duas versões do SKB, ambas apresentadas na Figura 2.7, com o casco em forma de *trimaran*, uma vela, um leme, um patilhão e um motor auxiliar. A primeira, em 2005, tinha 1,93 m de comprimento, 0,94 m de largura e 86,2 kg de massa. Foi equipado com GPS, anemómetro, três acelerómetros, um para cada eixo, e bússola. O segundo protótipo, desenvolvido em 2008, possui adicionalmente um módulo de comunicação *Iridium*, um painel fotovoltaico e um sensor acústico. Em termos de dimensões, tem 1,81 m de comprimento, 1,04 m de largura e 63,5 kg de massa, sendo consideravelmente mais leve.

A arquitetura de controlo é idêntica em ambos os protótipos. Possui três camadas hierárquicas, cada uma com um *Communicating Sequential Process* (CSP) independente, que permite a cada camada avaliar e controlar o seu próprio estado. A camada superior seleciona o modo de navegação com base na direção do vento e na localização do SKB. O primeiro protótipo implementa oito modos de navegação: (i) sete para navegar à vela, considerando as diferentes mareações da vela; e (ii) um para navegar a motor, utilizando o motor auxiliar. O segundo protótipo implementa ainda o modo “*hove-to*” de *station keeping* para velocidades de vento superiores a 15 m/s. As entradas da camada intermédia de controlo incluem o modo de navegação selecionado pela camada superior, a velocidade e a direção do vento e a orientação do SKB. O objetivo desta camada é definir e controlar as configurações



(a) 1.º Protótipo



(b) 2.º Protótipo

Figura 2.7: Protótipos desenvolvidos do SKB [30]

da vela (mastro e retranca) ou do motor. Por fim, a última camada, controla o leme, tendo em consideração a rota definida e a direção do vento e da embarcação.

O primeiro protótipo do SKB foi testado em Key West na Florida em 2006, com correntes marítimas consideráveis e velocidades de vento de 4,1 m/s a 11 m/s. O ASV manteve-se dentro de um círculo de raio 250 m durante 24 h, excetuando três situações de perda do sinal GPS, em que saiu desta área. O SKB, nestes testes realizados na Florida, navegou 91 % do tempo à vela. Em 2008, os ensaios do segundo protótipo foram realizados durante sete dias na costa de Monterey, na Califórnia. Com ventos de velocidade superior a 14 m/s, o SKB manteve-se 90 % do período dentro de um círculo de raio 250 m e, durante os sete dias, manteve-se num círculo de raio 1000 m. Os resultados de *station keeping* obtidos foram consistentes com os da primeira fase: o ASV manteve-se, em modo de navegação à vela, num círculo de raio 250 m 90 % do tempo.

2.1.6 Unmanned Sailboat Platform

Zaisheng Pan et al. desenvolveram e testaram numa plataforma real, apresentada na Figura 2.8, uma arquitetura de controlo específica para barcos à vela não tripulados [31].

Esta plataforma está equipada com uma *Main Control Unit* (MCU), uma *Control Terminal Unit* (CTU), uma unidade de armazenamento de dados e módulos de comunicação 4G e satélite. A nível sensorial possui um *Inertial Measurement Unit* (IMU) de 9 eixos, um módulo GPS/*BeiDou*, uma câmara *High-Definition* (HD) multi-ângulos para detetar obstáculos, um sensor de temperatura ambiente e um sensor capaz de medir a velocidade e direção do vento. A MCU é composta por uma *Central Processing Unit* (CPU) de alto desempenho ou uma *Graphics Processing Unit* (GPU) e interfaces para os vários sensores, sendo responsável pelos algoritmos



Figura 2.8: Unmanned Sailboat Platform [31]

computacionalmente mais exigentes. Já a CTU contém um microcontrolador e interfaces para os sensores e atuadores, responsável por tarefas de baixo processamento computacional.

A arquitetura do sistema de controlo desenvolvido por Pan et al. [31] teve como requisitos: (i) a modelação e normalização dos *drivers* dos sensores e atuadores; (ii) ser *open-source*; e (iii) apresentar uma interface *online*, isto é, possibilitar a interação do utilizador final com o veículo através da *Internet*. Como se pode visualizar na Figura 2.9, a arquitetura proposta apresenta três camadas. A primeira camada, designada *Hardware Platform*, é responsável pela interface com todos os sensores e atuadores, processamento de dados e execução dos algoritmos implementados. A segunda camada, intitulada *Development Platform*, permite ao utilizador implementar os seus algoritmos em C/C++, Python ou Matlab sobre diferentes sistemas operativos e *frameworks* (Linux, *Robot Operating System* (ROS), UCOS *Real-Time Operating System* (RTOS)). A última camada, chamada *Application Platform*, disponibiliza aos utilizadores finais a informação, em tempo real, recolhida pela embarcação.

Foram realizados testes no Lago Yuehu, Ningbo, China, com vento de cerca de 5 m/s e uma corrente de aproximadamente 1 m/s. A Unmanned Sailboat Platform percorreu com sucesso o conjunto de *waypoints* definidos e permaneceu em *station keeping* numa área de 3 m de raio. Foi igualmente capaz de detetar, reconhecer e evitar obstáculos em tempo real.

2.2 Plataformas Propulsionadas por Motor

2.2.1 C-Enduro

Hyondong Oh et al. utilizam o C-Enduro, apresentado na Figura 2.10, para comparar a eficiência energética de vários algoritmos de *station keeping* num *Autonomous*

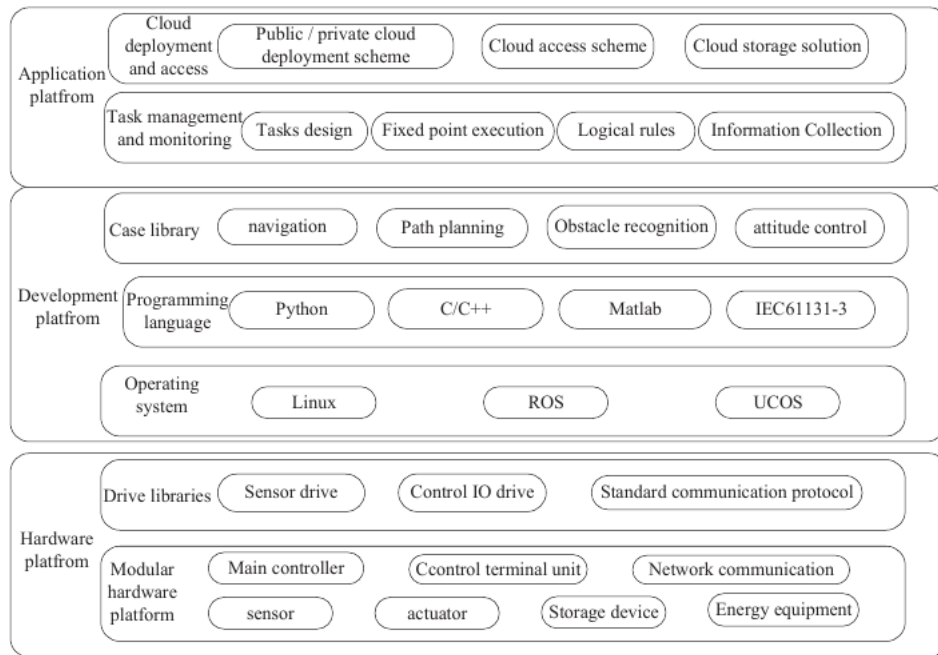


Figura 2.9: Arquitetura do Sistema de Controlo [31]

Surface Vehicle [32]. Este *catamaran* autónomo tem 4,1 m de comprimento, 2,45 m de largura e 350 kg de massa. Foi desenvolvido para permanecer longos períodos de tempo no mar e, por esta razão, utiliza três sistemas diferentes de produção de energia: (i) gerador a *diesel*; (ii) painéis fotovoltaicos; e (iii) turbina eólica.



Figura 2.10: C-Enduro ASV [33]

Foram testados cinco algoritmos de navegação:

- *Waypoints Line Of Sight* (LOS) consiste na navegação entre vários *waypoints* dentro da região de *station keeping* definida. Este conjunto de *waypoints* podem formar um retângulo ou um percurso em “zig zag”.

- *Lyapunov Vector Field Guidance* (LVFG) gera um campo vetorial circular, de raio previamente definido, em torno do *waypoint* desejado, considerando o vento e a corrente. O ASV é guiado pelo campo vetorial, navegando ao longo da circunferência centrada no *waypoint* desejado.
- *Conventional Dynamic Positioning* (DP), quando alcança o *waypoint* desejado, utiliza os motores para contrariar as forças externas e fixar a posição.
- *Modified DP* minimiza o consumo de energia dentro da região de *station keeping*, centrada no *waypoint* desejado, desligando os motores e ficando à deriva, até voltar a sair da zona de *station keeping*. Nessa altura, volta a navegar para *waypoint* desejado e desliga os motores quando regressa à região.
- *Modified DP + LVFG* combina os algoritmos LVFG e *Modified DP*, utilizando o campo vetorial gerado pelo algoritmo LVFG para prolongar o tempo em que o ASV está dentro da região de *station keeping*.

A Tabela 2.1 apresenta, resumidamente, as vantagens e desvantagens destes cinco algoritmos.

Tabela 2.1: Algoritmos de *station keeping* testados [32]

Algoritmos	Vantagens	Desvantagens
<i>Waypoints</i> LOS	Fácil de gerar	Mudança brusca de direção na transição de <i>waypoints</i>
LVFG	Mudança suave de direção Considera a velocidade do vento e da corrente	Cálculo complexo Navega numa única direção, recorrendo mais a um motor
<i>Conventional DP</i>	Bom para ambientes ideais	Ineficiente em ambientes dinâmicos
<i>Modified DP</i>	Poupa energia desligando os motores	Motores são frequentemente comutados Estima a velocidade do vento e da corrente
<i>Modified DP + LVFG</i>	Mudança suave de direção Poupa energia desligando os motores	Estrutura complicada

Os testes foram realizados em ambiente de simulação, mais em específico na *Simulink Toolbox Marine System Simulator* (MSS). Foram modeladas as forças do vento, das correntes marítimas e das ondas, com base em dados reais do mar do Norte durante o Inverno. Foi ainda acrescentado ruído gaussiano à força do vento e da corrente marítima para modelizar melhor as suas variações. A zona de *station keeping* definida foi um quadrado de 100 m de lado. Nesta simulação conclui-se que, para condições de vento com velocidade constante de 10 m/s e correntes marítimas de velocidade 0,2 m, todos os algoritmos são bastante semelhantes em termos de consumo energético. Nota-se diferença quando se acrescenta à simulação rajadas de vento com velocidades superiores e correntes marítimas de maior velocidade, neste

cenário os algoritmos *Modified DP* e *Modified DP + LVFG* são os que apresentam melhores resultados.

2.2.2 FLEXUS

O FLEXUS é um *catamaran* autónomo, desenvolvido pelo Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência (INESC TEC) no âmbito do projeto europeu *Road-, Air-, and Water- based Future Internet Experimentation* (RAWFIE) [34], com 15 kg de massa e 0,9 m de comprimento [35]. O principal objetivo foi o desenvolvimento de ferramentas para a comunicação remota entre múltiplas plataformas robóticas através da *Internet*. Dadas as suas especificações físicas, o FLEXUS é um ASV prático em termos logísticos, sendo facilmente transportado e colocado na água para missões.

Este veículo, apresentado na Figura 2.11, é constituído por um *pack* de baterias com capacidade de 240 W h, dois motores em configuração diferencial, um *Single Board Computer* (SBC), um recetor GPS, um magnetómetro, suportando 5 kg de *payload*. Possui um sistema de comunicação constituído por três módulos Wi-Fi: dois operaram nas faixas de 2,4 GHz e 5 GHz e o terceiro funciona como *access point*.



Figura 2.11: FLEXUS ASV [35]

O FLEXUS fez os seus primeiros testes perto do Porto, Portugal, mas foi em Lisboa, na base naval da marinha portuguesa, que realizou as suas primeiras missões. O principal objetivo foi testar e avaliar o desempenho do ASV e, principalmente, o sistema de comunicação. A plataforma conseguiu manter a comunicação ponto-a-ponto até 550 m e realizar autonomamente: (i) uma missão de *station keeping*, conseguindo permanecer dentro de um círculo de raio 1 m; e (ii) uma missão de varrimento, tendo percorrido uma área de 10 000 m² com um erro da trajetória

inferior a 1 m. No total, o FLEXUS navegou 23 km, conseguindo percorrer 8,3 km apenas com um *pack* de baterias.

2.2.3 OASIS

Desenvolvido no âmbito do projeto *Costal Observations* gerido pelo *Center for Innovative Technology* (CIT) e financiado pelo NOAA [36], o *Ocean Atmosphere Sensor Integration System* (OASIS) é um ASV concebido para missões de longa duração, onde o objetivo é obter dados científicos do oceano. Além destes requisitos, havia também o objetivo de tornar este tipo de missões mais acessíveis economicamente e práticas, para isso a plataforma deveria ser *low-cost*, reutilizável e de fácil logística [37].

O OASIS tem o casco em fibra de vidro, de maneira a reduzir o preço de produção da plataforma, um patilhão, um leme, atuado por um motor passo, e uma motor servo para propulsão. Além disso, através dos seis painéis fotovoltaicos instalados, utilizava a energia solar para produção de eletricidade e carregar as baterias, prolongando a duração das missões. A nível de eletrónica possui um SBC, para controlo e processamento de todos os dados a bordo, um módulo *Iridium* para comunicação, um GPS, um IMU, uma bússola e um inclinómetro. Apresenta também uma estação meteorológica localizada no cimo do mastro, esta era constituída por um anemómetro, um barómetro, um sensor para medir a humidade e outro para medir a temperatura ambiente. Como se pode visualizar na Figura 2.12, o OASIS, era possível de ser transportado por um reboque de carro, tornando-se logisticamente mais prático em missões.



Figura 2.12: OASIS ASV num atrelado [37]

Este ASV pode navegar autonomamente ou ser comandado manualmente. A arquitetura de controlo implementada está dividida em sete subsistemas: (i) *Propulsion*, (ii) *Communications*, (iii) *Command and Data Handling* (C&DH), (iv)

GN&C, (v) *Attitude Determination*, (vi) *Power* e (vii) *Core Meteorological Sensors*. Terminada a missão, o OASIS permanece em modo *station keeping*, aguardando por novas tarefas ou *waypoints*.

Em maio de 2006, foram realizados dois testes em Chesapeake Bay, Delaware, Maryland, Virginia. No primeiro, o OASIS navegou aproximadamente 7,4 km e, no segundo, percorreu 11 km. Em ambas as missões, os *waypoints* estavam espaçados entre si aproximadamente 1,85 km, correspondente a uma milha náutica.

Posteriormente, foram desenvolvidos mais dois OASIS, com aproximadamente 5 m de comprimento e 1361 kg de massa. Fisicamente, são bastante semelhantes ao primeiro [38], como se pode ver na Figura 2.13. Em termos de melhoramentos, integram um módulo Wi-Fi e de rede móvel, uma câmara para auxiliar a navegação, um *Automatic Identification System* (AIS) e sensores adicionais, como por exemplo um *Acoustic Doppler Current Profiler* (ADCP) para caracterizar as correntes marítimas.



Figura 2.13: OASIS 2 e 3 durante os testes na Chincoteague Bay [38]

O OASIS 2 foi a primeira das três plataformas a ser testada no oceano. Em 2007, navegou ao longo de 26 km e efetuou várias manobras de *station keeping*, tendo passado uma noite inteira no oceano. Já em 2008, com o objetivo de estudar o fluxo de dióxido de carbono no mar e o fenómeno do crescimento excessivo de algas nocivas, também designado por *Harmful Algal Bloom* (HAB), foi realizado um teste ao longo de 24 h, onde os OASIS percorreram 63 km da costa do oceano atlântico.

2.2.4 ROAZ

O ROAZ e o ROAZ II são dois ASV de duplo casco [39] desenvolvidos pelo Laboratório de Sistemas Autónomos (LSA) do Instituto Superior de Engenharia do Porto (ISEP) [39]. A Figura 2.14 apresenta o ROAZ, concebido para ambientes com águas

calmas e pouca ondulação, tipicamente rios e lagos, com 1,5 m de comprimento e 1 m de largura [40].



Figura 2.14: ROAZ [40]

Independentemente da sua pequena dimensão, o ROAZ consegue suportar 50 kg de massa adicional, tornando-se bastante versátil na execução de missões. Uma das suas principais aplicações é funcionar como veículo de suporte, lançando e recolhendo AUV na água, e de *relay* de comunicação entre o AUV e uma estação costeira [40]. Está equipado com *packs* de baterias NiMh, um SBC de baixo consumo, um modem *ethernet* com uma antena externa para as comunicações rádio, um modem acústico para comunicação subaquática, um *Inertial Navigation System* (INS), um IMU, uma bússola magnética, um recetor GPS, um sistema de visão, um *scanning sonar* e, dependendo do âmbito da missão, outros sensores de carácter científico.

O ROAZ II, apresentado na Figura 2.15, foi desenvolvido para executar missões em mar aberto [39]. Mantendo um casco de *catamaran*, o ROAZ II tem 2 m de comprimento, 2,2 m de largura e uma massa de 200 kg. Esta segunda versão, dependendo da velocidade de operação e da configuração das baterias, tem uma autonomia de 10 h e consegue suportar aproximadamente 200 kg de *payload* [41]. Para além da mesma eletrónica do ROAZ, esta versão comporta ainda uma câmara termográfica e um sensor de condutividade, temperatura e profundidade da coluna de água (CTD).

Ambos os veículos realizaram múltiplas missões nos cenários para os quais foram projetados. O ROAZ participou numa missão de demonstração de tecnologia organizada pela *North Atlantic Treaty Organization* (NATO) perto de Troia, com o objetivo de demonstrar os vários sistemas de comunicação. O ROAZ II, que realizou diversas missões de levantamento batimétrico através do *scanning sonar*, vigiou e inspecionou durante dois dias o porto de Sesimbra através do seu sistema de visão, com o objetivo de detetar e identificar fontes de poluição na água [43].



Figura 2.15: ROAZ II [42]

2.2.5 Underactuated ASV

O objetivo do Underactuated ASV, proposto por Arvind Pereira [12], é manter o seu centro geométrico numa coordenada GPS específica por um período de 2 min a 3 min, permitindo a recolha de amostras da coluna de água.

Esta embarcação não controla diretamente a deriva lateral (*sway*), ou seja, trata-se de um ASV sub-atuado. O barco tem uma massa total 62,3 kg e o casco é plano, com 2,1 m de comprimento e 0,7 m de largura. Possui um leme e dois motores separados aproximadamente por 0,27 m, que permitem o deslocamento avante/atrás (*surge*) e a mudança de rumo (*yaw*). Devido à sua forma, a plataforma é significativamente influenciada pelo vento lateral, principalmente quando o seu eixo longitudinal se encontra perpendicular à direção do vento. A nível de eletrónica, para auxiliar a navegação, o ASV possui um IMU com um magnetómetro integrado, um recetor GPS e um anemómetro. A Figura 2.16 apresenta o Underactuated ASV e, de uma maneira conceptual, a arquitetura de controlo implementada.

Para o controlo do rumo (*heading*) adota um controlador Proporcional Derivativo (PD), que usa como referência o erro entre o *heading* desejado e o *heading* corrente da embarcação, obtido através do *Extended Kalman Filter* (EKF) implementado, que considera as medidas do GPS, IMU e da bússola. Aplica o método *LOS guidance* para determinar o *heading* e a distância à coordenada desejada, comparando a posição atual do ASV com a coordenada do destino. Implementa um controlador *feedforward* para o vento, com base nos dados obtidos pelo anemómetro, conjuntamente com um controlador não linear designado *Command Combination Logic*. Este último compara e otimiza as saídas (*outputs*) dos controladores referidos para garantir que o ASV atinge o destino e, uma vez chegado, coloca o seu eixo longitudinal paralelo à direção do vento, minimizando o efeito desta força externa na embarcação.

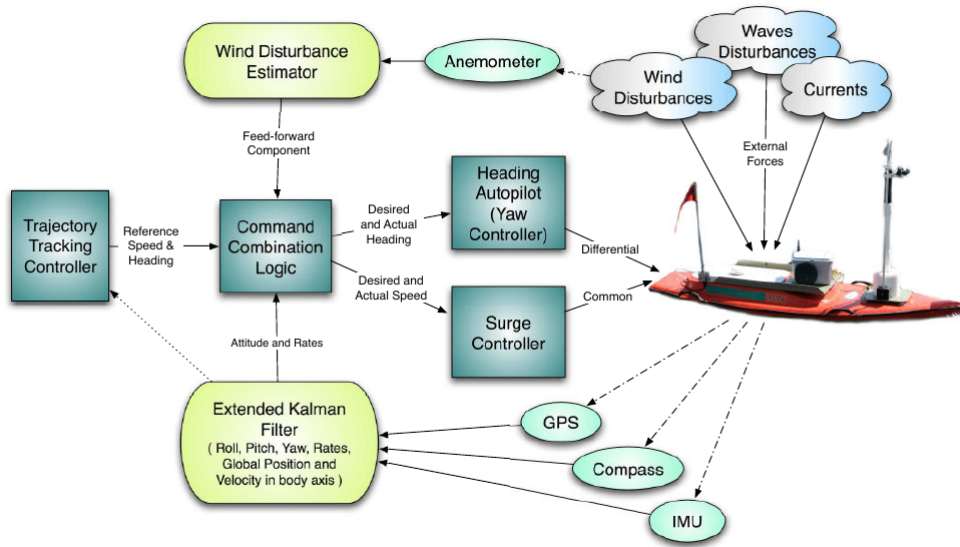


Figura 2.16: Underactuated ASV em modo *station keeping* [12]

Em condições reais, o ASV conseguiu manter a posição desejada dentro de um círculo de raio inferior a 3 m. A velocidade do vento variou de 0 m/s a 2,8 m/s e a direção entre $\pm 15^\circ$ durante os testes. Os resultados da simulação apresentaram um erro em regime permanente, tendo sido acrescentada uma componente integral ao controlador, melhorando significativamente a posição final do veículo face à posição desejada.

Como melhorias futuras, os autores sugerem: (i) a implementação de um controlador que considere as restantes forças externas, nomeadamente as correntes marítimas; e (ii) usar um sonar ou um sistema de visão para obter o *feedback* da posição da embarcação, passando a integrar mais informação no EKF.

2.2.6 WAM-V USV16

O *Wave Adaptive Modular-Vessel* (WAM-V) USV16 é um ASV com aproximadamente 5 m de comprimento e 204 kg de massa, desenvolvido pela Marine Advanced Research, Inc., Richmond, Califórnia, Estados Unidos da América [44]. Como se pode ver na Figura 2.17, trata-se de um semi-rígido com duplo casco e dois motores, um em cada casco, que permite o movimento em *surge*, *sway* e *yaw*. Na configuração vendida com seis baterias pode aguentar até 15 h a uma velocidade média de 2,57 m/s e carregar mais 68 kg de *payload*.

Este barco foi utilizado e testado em condições reais, com velocidades de vento até 2,5 m/s e pequenas correntes marítimas, por Sarda et al. [45, 46]. Possui três controladores não lineares diferentes para manter a posição e orientação do veículo pelo maior período de tempo possível. Pelos resultados obtidos em [45] foi possível



Figura 2.17: WAM-V USV16 durante os testes de *station keeping* [45]

concluir que o controlador *sliding mode* é o que apresenta melhores resultados no geral. Dependendo do tipo de aplicação, caso seja necessário manter apenas a posição do ASV desprezando a orientação, o controlador *backsteeping* pode ter melhores resultados. Em [46], Sarda acrescentou um controlador *feedforward* para o vento o que melhorou consideravelmente a *performance* dos controladores controlador *backsteeping* e PD quando o barco está perpendicular ao vento. O controlador *sliding mode* não teve melhorias significativas com esta nova implementação. Ainda neste estudo foi possível concluir que apenas um anemômetro é suficiente para caracterizar a velocidade e a direção do vento em embarcações de pequenas dimensões.

Huajin Qu em [47] realizou testes com a mesma plataforma em North Lake, Hollywood, na Flórida, com uma velocidade do vento média de 4 a 6 m/s. Qu usou também um controlador *sliding mode* para a navegação do WAM-V USV16 em conjunto com um controlador *feedforward* para o vento, utilizando a velocidade e a direção do vento aparente para fazer a modelação do vento. Os resultados mostraram que utilizar um controlador *feedforward* para o vento melhora significativamente a *performance* da manobra de *station keeping*, principalmente quando o vento é mais veloz. Os erros médios da posição e da orientação do WAM-V USV16 são apresentados na Tabela 2.2.

Tabela 2.2: Média dos erros da posição e orientação do WAM-V USV16, com e sem controlador *feedforward* de vento [47]

Erro	Sem controlador de vento	Com controlador de vento
Posição (m)	2,9	0,8
Orientação (°)	-17,3	-7,9

O WAM-V USV16 também foi utilizado por Sarda e Dhanak em [48] numa missão de *Launch and Recovery* (L&R) do Hydroid REMUS 100 [49]. O WAM-V 16 estava equipado com um sistema de navegação que incluía um IMU, um GPS e uma bússola digital, um SBC, um recetor e transmissor de frequência rádio e um anemômetro

ultrassónico. A missão foi realizada com sucesso, tendo o ASV mantido a posição e a orientação desejada e permitindo o lançamento e recolha do AUV. Concluiu-se também que com as condições atmosféricas, nomeadamente a ondulação, afetam a *station keeping*.

2.2.7 Wivenhoe ASV

O Wivenhoe é um *catamaran*, com 4,9 m de comprimento, desenvolvido para recolher, com precisão, dados relativos à qualidade da água e aos gases de efeito estufa em águas paradas, como rios ou lagos [50]. Este ASV tem de ser também capaz de operar continuamente pelo menos durante 24 h, detetar e evitar obstáculos do meio onde se encontra e operar com a velocidade do vento até 8 m/s. Para cumprir estes requisitos o Wivenhoe estava equipado, além dos vários sensores de carácter científico, com um GPS, uma bússola, um sensor de profundidade, um Pentium M a 1,4 GHz como CPU, um sensor de vento, um módulo Wi-Fi e, para detetar os obstáculos, um *laser scanner* e uma câmara. Como é possível visualizar na Figura 2.18, o Wivenhoe, tem também painéis fotovoltaicos para carregar as suas baterias.



Figura 2.18: Wivenhoe ASV [50]

A nível de planeamento de missões, era especificado um conjunto de *waypoints* com uma *tag* que definia o objetivo da missão, por exemplo *profile*, *station keeping*, *dock*. O ASV a navegar entre *waypoints* tentava manter uma linha reta, quando não era possível devido a um obstáculo, o veículo contornava-o, geralmente pelo lado com maior profundidade, e no final voltava à trajetória inicial.

Os testes para avaliar a *performance* de navegação deste *catamaran*, assim como, as tarefas de evitar obstáculos e recolher dados do exterior, foram realizados no lago de Wivenhoe, Queensland, Australia. Mesmo com ventos com velocidade de 10,3 m/s, o Wivenhoe, foi capaz de seguir a trajetória programada, conseguindo

também, detetar e evitar obstáculos e recolher dados relativos à temperatura da água e profundidade do lago, ao longo de 13 km.

2.3 Plataformas Propulsionadas por Ondas

2.3.1 Wave Glider

Desenvolvido por Roger Hine, fundador da Liquid Robotics [51] que comercializa o Wave Glider, este ASV aproveita a força das ondas do mar para navegar [52]. As asas do *glider* submerso permitem converter a força vertical das ondas numa força horizontal de deslocamento, alcançando uma velocidade média de 0,78 m/s. O veículo comporta um flutuador à superfície e um *glider* submerso, que estão ligados através de um cabo flexível de 8 m de comprimento, que estão representados na Figura 2.19. Possui dois painéis fotovoltaicos de 665 W h para carregamento das baterias. As baterias alimentam os sistemas de controlo, navegação, e comunicação e os sensores de *payload*, que podem pesar até 59 kg [53]. Para navegar, o Wave Glider utiliza como sensores um GPS, um acelerómetro de 3 eixos e uma bússola magnética. O sistema de comunicação contém um modem *Iridium* e um modem *Radio Frequency* (RF) de elevada largura de faixa.

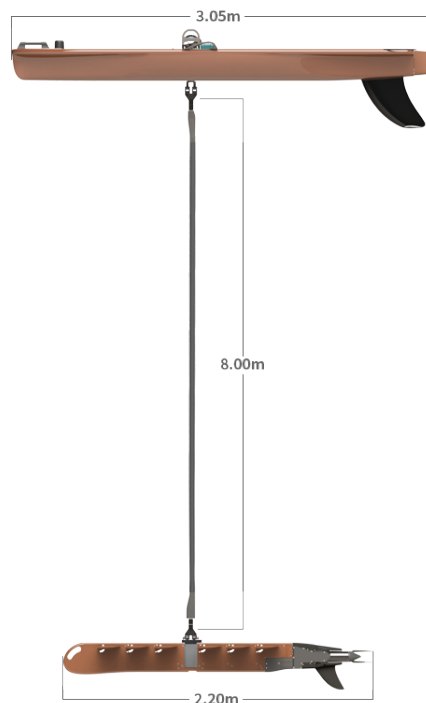


Figura 2.19: Wave Glider composto por flutuador (superfície) e *glider* (submerso) [51]

Em termos de *station keeping*, o Wave Glider consegue manter-se dentro de um círculo de raio de 25 m [52]. Além disso, executou uma missão no mar durante 247 dias contínuos. Nesse período, percorreu 11 500 km e suportou condições atmosféricas e marítimas adversas, com ventos de 21 m/s e ondas de 4,5 m de altura. O veículo, nessas condições, tende a comportar-se como uma âncora, o que é interessante para manter a posição.

O Wave Glider realizou diversas missões para organismos do NOAA [36]. O *Pacific Marine Environmental Laboratory* (PMEL) [54] utilizou o Wave Glider para estudar a coluna de água depois do degelo e os fenômenos ambientais associados. Posteriormente, o mesmo laboratório instalou um conjunto específico de sensores de âmbito científico para estudar o fluxo de carbono e a acidificação dos oceanos [55]. O *National Data Buoy Center* (NDBC) [56] utilizou o Wave Glider para estudar a possibilidade de complementar, ou até substituir, as suas boias meteorológicas e oceanográficas [57]. Nestes testes, o principal objetivo do NDBC foi avaliar o desempenho do ASV em termos de *station keeping* em diferentes ambientes. Os resultados mostram que, em geral, o Wave Glider é mais eficiente com ondas mais altas e baixa corrente marítima, resistindo à bioincrustação e ataques de tubarões. Em 2012 e 2013, o *Northwest Fisheries Science Center* [58] utilizou o Wave Glider para avaliar a sua eficácia em missões de pesquisa e recolha de dados acústicos. Nesta missão o Wave Glider rebocou um torpedo submerso. O *setup* desta missão é apresentado na Figura 2.20.



Figura 2.20: Wave Glider na missão do *Northwest Fisheries Science Center* [57]

2.3.2 Boia Robótica

Desenvolvida por Joe et al. (2018) com o objetivo de contribuir para a *Internet of Maritime Things*, a boia robótica é constituída por três sistemas: (i) propulsão, (ii)

recolha de energia e (iii) controlo de flutuação [59]. O sistema de propulsão destina-se a manobrar a boia, quando necessário, para manter a posição. É composto por um *wave glider*, semelhante ao do Wave Glider de Hine (2009) [52], mas otimizado para *station keeping*, aumentando o tamanho do leme e diminuindo o número de lâminas. A recolha de energia é feita através da *Wave Turbine* desenvolvida. Esta turbina é composta por duas camadas rotativas, que rodam em sentidos opostos, com quatro lâminas cada. Este sistema aproveita o movimento de subida e descida das ondas para fazer girar a *wave turbine*, gerando energia para alimentar a boia. Ambos os sistemas são apresentados na Figura 2.21.

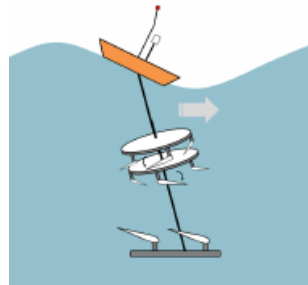


Figura 2.21: Boia robótica composta por flutuador (superfície), *wave turbine* (submersa a meio) e o *wave glider* (submerso no fundo) [59]

A boia robótica utiliza o controlo descrito em [60] para fazer *station keeping*. Consiste num controlador Proporcional Integral Derivativo (PID) que considera o erro da orientação do *wave glider*, atuando no leme para manter o veículo sempre apontado para o *waypoint* de *station keeping*. O ângulo do leme é colocado a zero no raio de 10 m em torno do *waypoint* desejado.

Foram realizados testes com ondas de 0,3 m de altura e frequência 0,4 Hz. A boia robótica permaneceu em *station keeping* dentro de uma área de 3 m de raio durante 25 min e a *wave turbine* conseguiu gerar em média 1,67 W, tendo um pico de 8 W.

2.4 Análise

As características mais relevantes dos ASV referidos para o presente trabalho são apresentadas na Tabela 2.3. Os pontos analisados foram:

- Fácil Logística considera a dimensão, que não deve ser superior a 3 m, a massa, que deve ser inferior a 100 kg, e a facilidade de transporte e operação do veículo, que não deve requerer mais de três pessoas.

- Autossustentabilidade contempla a capacidade de produção de energia do veículo para carregar as baterias de alimentação do sistema, indicando a fonte de energia utilizada.
- Autonomia consiste no tempo de operação do veículo, sem qualquer intervenção externa, referido pelo fabricante ou investigador.
- *Payload* corresponde à carga adicional que o veículo pode transportar, expressa em quilogramas, por exemplo, em sensores específicos.
- *station keeping* caracteriza a capacidade do veículo permanecer em torno de uma posição, expressa através do raio da área, em metros, resultante.

Os valores do OASIS e do Wivenhoe ASV são indicados em [38] e [50], respectivamente, como requisitos iniciais do veículo, não tendo sido verificados em testes. O raio de *station keeping* indicado na Tabela 2.3 para o WAM-V USV16, foi o testado por Qu em [47] sem controlador *feedforward* para o vento, sendo este o pior resultado obtido, ou seja, a área de maior raio. Na Tabela 2.3 é utilizado "✓" quando o ASV satisfaz a característica, "✗" quando não cumpre e "-" quando não há informação.

Tabela 2.3: Características relevantes dos ASV analisados

ASV	Fácil Logística	Autossustentabilidade (Fonte de energia)	Autonomia	Payload	station keeping (Raio)
Atlantis	✗	✗	15 h	-	✓ (Usa o motor auxiliar)
FASt	✓	✓ Solar	-	Poucos kg	✓ 20 m
SailBuoy	✓	✓ Solar	≤1 ano	15 kg	-
Saildrone Explorer	✗	✓ Solar	>1 ano	-	-
SKB 2	✓	✓ Solar	-	-	✓ 250 m
Unmanned Sailboat Platform	-	-	-	-	✓ 3 m
C-Enduro	✗	✓ Solar + Eólica	<3 meses	-	✓ 100 m
FLEXUS	✓	✗	± 1 h	5 kg	✓ 1 m
OASIS	✗	✓ Solar	Semanas a meses	363 kg	✓ -
ROAZ	✓	✓ Solar	-	50 kg	-
ROAZ II	✗	✗	10 h	200 kg	-

Eólica

Motor

Tabela 2.3: Características relevantes dos ASV analisados

ASV	Fácil Logística	Autossustentabilidade (Fonte de energia)	Autonomia	Payload	station keeping (Raio)
Underactuated ASV	✓	✗	-	-	✓ 3 m
WAM-V USV16	✗	✗	15 h	68 kg	✓ 3 m
Wivenhoe ASV	✗	✓ Solar	24 h	-	✓ -
Wave Glider	✗	✓ Solar	≤1 ano	59 kg	✓ 25 m
Boia Robótica	✓	✓ Ondas	-	-	✓ 3 m

Para além das características analisadas dos ASV, outro aspeto relevante para o presente trabalho são as técnicas implementadas para navegação e realização da manobra de *station keeping*. A Tabela 2.4 mostra os diferentes algoritmos utilizados, assim como as suas vantagens e desvantagens.

Tabela 2.4: Algoritmos implementados nos ASV analisados

Algoritmos	Vantagens	Desvantagens
Arcos e Segmentos de reta	Mudança suave de direção na transição de <i>waypoints</i>	Gera arcos para todos os <i>waypoints</i>
<i>Waypoints</i> LOS	Fácil de gerar	Mudança brusca de direção na transição de <i>waypoints</i>
LVFG	Mudança suave de direção Considera a velocidade do vento e da corrente	Cálculo complexo Navega numa única direção, recorrendo mais a um motor
<i>Conventional</i> DP	Bom para ambientes ideais	Ineficiente em ambientes dinâmicos
<i>Modified</i> DP	Poupa energia desligando os motores	Motores são frequentemente comutados Estima a velocidade do vento e da corrente
<i>Modified</i> DP + LVFG	Mudança suave de direção Poupa energia desligando os motores	Estrutura complicada

Em conjunto com os algoritmos de navegação e de *station keeping* referidos, foram utilizados controladores PID, PI, PD, *sliding mode* e *backstepping*. Além disso, nalguns casos foi ainda implementado um controlador controlador *feedforward* para o vento, tendo sido obtido bons resultados.

Outro ponto importante são as arquiteturas de controlo. Neste conjunto de ASV existem arquiteturas:

- Hierárquicas – compostas por uma camada superior de alto nível, como planeamento da missão, e camadas inferiores, com a função de interagir e controlar os atuadores.
- Modulares – compostas por módulos com funções distintas, facilitando a integração de novos sensores ou funcionalidades.
- Reativas – compostas por algoritmos de baixo nível, que adaptam o comportamento do veículo aos eventos do meio ambiente percecionados.

A nível de eletrónica, todos os veículos analisados comportam sensores de posicionamento, velocidade e tempo (GPS), vento (anemómetro) e inércia (IMU) em conjunto com sistemas de comunicação (*Iridium* e/ou Wi-Fi) e de processamento.

2.5 Sumário

Este capítulo apresenta o levantamento dos ASV propulsionados a vento, motor e ondas efetuado. Em relação a cada plataforma, detalha as suas características físicas, eletrónicas, e de controlo, os algoritmos de navegação e *station keeping* e, por fim, os resultados obtidos em missões ou simulações.

Um dos pontos a realçar é que todos os ASV propulsionados pelo vento utilizam uma vela, embora existam outros tipos de estruturas para impulsionar os barcos através do vento. Enqvist faz um breve resumo dos vários métodos existentes, concluindo e reforçando a pesquisa realizada, que as velas são as que têm mais vantagens e melhor se enquadram para os objetivos pretendidos [61].

O capítulo 2 encerra com a comparação dos ASV em termos de logística, sustentabilidade, autonomia, *Payload* e *station keeping*, que são os aspetos mais importantes para o presente trabalho, e identifica os principais algoritmos, controladores e eletrónica utilizados.

Capítulo 3

Navegação à Vela

Este capítulo 3 tem como objetivo apresentar os componentes principais de um veleiro e explicar os conceitos físicos envolvidos na navegação à vela.

Um barco à vela é composto por quatro componentes principais, apresentados na Figura 3.1, e são estes: vela, casco, patilhão ou quilha e leme, em Inglês designados *sail*, *hull*, *keel* e *rudder*, respetivamente.

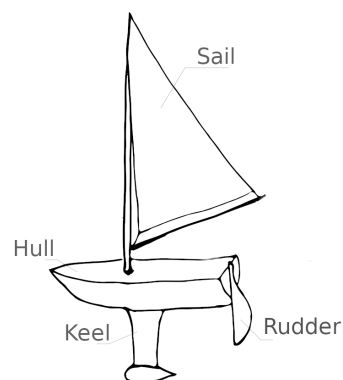


Figura 3.1: Principais componentes de um barco à vela

O casco de um barco pode ser dividido em quatro zonas, apresentadas na Figura 3.2 e são estas:

- proa ou *bow* - parte da frente do barco;

- popa ou *stern* - parte de trás do barco;
- bombordo ou *port* - parte esquerda do barco, estando dentro dele e virado para a proa;
- estibordo ou *starboard* - parte direita do barco, estando dentro dele e virado para a proa.

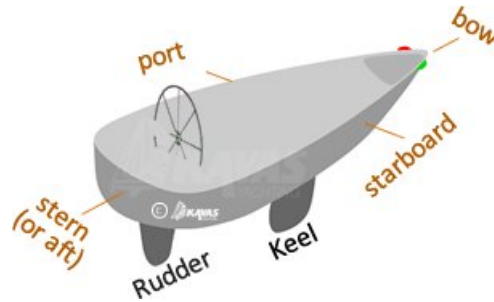


Figura 3.2: Diferentes zonas do casco e os seus elementos [62]

De uma forma geral um barco tem seis graus de liberdade, ou seja, tem três movimentos de translação e três de rotação. No eixo x o movimento de translação é designado *surge* e o de rotação *roll*. No eixo y o movimento de translação é designado *sway* e o de rotação *pitch*. Por fim, no eixo z , o movimento de translação designa-se *heave* e o de rotação *yaw* [63]. Na Figura 3.3 são apresentados os seis graus de liberdade referidos.

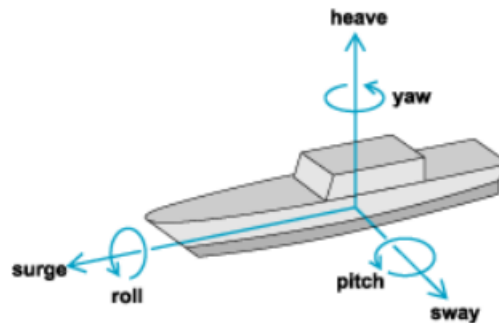


Figura 3.3: Seis graus de liberdade de um barco [13]

3.1 Navegação

Um veleiro é um sistema mecânico que flutua e interage com dois fluídos, o ar e a água, onde o movimento relativo nos diferentes fluídos gera as forças aerodinâmica e hidrodinâmica [64].

Em repouso, o comportamento de um veleiro explica-se através da lei fundamental da hidrostática, vulgo princípio de Arquimedes. A força de flutuação ($\vec{F}_B$) é

proporcional ao volume de água deslocado (∇) pelo barco e tem sentido contrário ao da força do peso do barco ($\vec{F}_W$) [65]. Estas forças podem ser calculadas através das Equações 3.1 e 3.2. Assim, o barco flutua quando a força de flutuação é igual ou maior que a do peso do barco [66].

$$\vec{F}_B = \rho_{water} \vec{g} \nabla \quad (3.1)$$

$$\vec{F}_W = m \vec{g} \quad (3.2)$$

Em movimento, o comportamento de um veleiro explica-se através das forças aerodinâmica e hidrodinâmica. Estas forças, que resultam da interação do veleiro com os diferentes meios (ar e água), podem ser divididas em força de arrasto (paralela ao fluxo do fluido) e força de sustentação (perpendicular ao fluxo do fluido) [11]. Neste contexto, existem três modos de navegação à vela: “a favor do vento” ou à popa, ao largo e “contra o vento” ou à bolina. A navegação ao largo pode ser explicada através dos outros dois modos. Um barco aprofado ao vento, i.e., com um ângulo de 0° com o vento, não consegue navegar. Assim, a navegação à bolina (ver Anexo A) consiste na realização de um percurso em “zig-zag” com um ângulo de cerca 40° em torno da direção do vento. Na navegação à popa, o veleiro navega a favor do vento, i.e., o vento entra pela popa, fazendo um ângulo de 180° com o vento.

De acordo com Kimball et al. [67], a velocidade do barco ($\vec{v}_b$) relativamente à água depende da velocidade dos dois fluidos envolvidos:

- velocidade do vento real ($\vec{v}_{tw}$), medida em relação à água, que se designa velocidade do vento aparente ($\vec{v}_{aw}$) quando medida em relação ao referencial do barco;
- velocidade da água ou corrente real ($\vec{v}_c$), medida em relação à costa ou fundo do mar, que se designa velocidade da corrente aparente ($\vec{v}_{ac}$) quando medida em relação ao referencial do barco.

3.1.1 Popa

Um veleiro com uma vela tradicional navega “a favor do vento” abrindo as velas para aproveitar o vento que entra pela popa, sendo impulsionado para a frente, como se apresenta na Figura 3.4.

Este modo de navegação justifica-se através da lei de Newton ($\vec{F} = m\vec{a}$). De uma forma simplificada, as partículas do ar quando embatem contra a superfície vélica



Figura 3.4: Veleiro a navegar à popa [67]

exercem uma força que a faz deslocar para a frente [67]. Esta força é quadraticamente proporcional à velocidade relativa do vento face ao barco ($\vec{v}_{aw}$) e diretamente proporcional à área vélica (A_v) e à densidade do meio, i.e., a densidade do ar (ρ_{air}). Nesta situação, a força gerada na vela ($\vec{D}_{sail}$) é paralela ao vento aparente ($\vec{v}_{aw}$) e designa-se força de arrasto (*drag*), sendo calculada pela Equação 3.3.

$$\vec{D}_{sail} = \frac{1}{2} \rho_{air} A_{sail} \vec{v}_{aw}^2 C_D(\alpha) \quad (3.3)$$

O coeficiente de arrasto (C_D) permite caracterizar o comportamento de um fluido (e.g., ar) num objeto (e.g., vela) [67]. Um coeficiente de arrasto baixo significa que o objeto se desloca com menor resistência no fluido; inversamente, um elevado coeficiente de arrasto indica que o objeto se desloca com maior resistência no fluido [68]. Este coeficiente (C_D) depende do ângulo de ataque (α) e da forma do aerofólio ou *airfoil* (vela) [13], ou seja, se é mais ou menos aerodinâmico. O vento aparente ($\vec{v}_{aw}$) permite calcular a força que o vento exerce na vela. À popa, o $\vec{v}_{aw}$ é calculado através da Equação 3.4.

$$\vec{v}_{aw} = \vec{v}_{tw} - \vec{v}_b \quad (3.4)$$

Neste modo de navegação, a interação da água com o casco e todos os seus componentes, cria uma força contrária à do vento na vela [67]. A Equação 3.5 permite calcular a força de arrasto no casco ($\vec{D}_{hull}$) relativamente à água.

$$\vec{D}_{hull} = \frac{1}{2} \rho_{water} A_{hull} \vec{v}_b^2 C_D \quad (3.5)$$

Uma vez que a área do casco submerso (A_{hull}) pode não ser simples de determinar

é possível estimá-la. Por um lado, o princípio de Arquimedes estipula que "a massa de água deslocada é igual à massa do barco (m_{boat})" [67]. Por outro lado, é possível assumir que o volume de água deslocada pelo casco de um barco é aproximadamente igual à metade do seu *length overall* (LOA). Recorrendo a estas aproximações a área do casco é dada pela Equação 3.6.

$$A_{hull} = \frac{2 m_{boat}}{\rho_{water} LOA} \quad (3.6)$$

Tal como era de esperar, é possível concluir que o veleiro inicia o movimento apenas quando tem aceleração, i.e., $\vec{D}_{sail} > \vec{D}_{hull}$ [67].

É de realçar que, ao invés do que acontece com uma vela tradicional, num veleiro com uma vela asa rígida pode não ser desejável maximizar a área vélica exposta ao vento [69]. À popa pode ser mais eficiente utilizar a vela asa como uma superfície de *lift*, como acontece com a vela tradicional à bolina.

3.1.2 Bolina

A navegação à bolina envolve as forças de arrasto e de sustentação produzidas pelo vento na vela e pela água no casco, sendo a força de sustentação a principal responsável pela navegação "contra o vento" [67]. Enquanto a força de arrasto ($\vec{D}$) tem a direção do movimento do veleiro, a força de sustentação ($\vec{L}$) ou *lift* é perpendicular. A Figura 3.5 apresenta as duas forças e a sua resultante. Dependendo do meio, a força resultante $\vec{F}$ designa-se força aerodinâmica ou força hidrodinâmica.

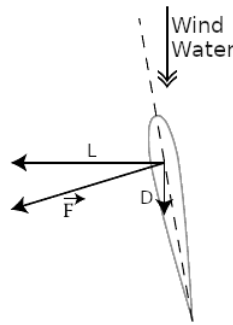


Figura 3.5: Força de sustentação ($\vec{L}$), arrasto ($\vec{D}$) e resultante ($\vec{F}$) numa vela asa rígida ou patilhão

Todas as forças envolvidas são representadas por um vetor, com uma magnitude, direção e um sentido. A magnitude do vetor é especificada por um número positivo e a unidade apropriada do que estiver a representar, no caso de uma força a unidade é Newton (N). A direção é determinada pela relação entre o vetor e o eixo de referência e, por fim, o sentido do vetor é especificado pela ordem de dois pontos numa reta paralela ao vetor [70].

A força aerodinâmica, que resulta da soma das forças de sustentação e de arrasto na vela, pode ser decomposta em duas componentes. A primeira, projetada paralelamente ao referencial local do veículo, designada por força de propulsão ($\vec{T}$), e a segunda componente, projetada perpendicularmente ao referencial local do barco, designada por força de inclinação ($\vec{H}$) [71]. A força de propulsão ($\vec{T}$), como o nome indica, vai permitir impulsionar o movimento do veleiro, enquanto a componente perpendicular ($\vec{H}$) vai criar um momento de rotação em *roll* (adornar o barco), que irá ser contrariado pelas componentes da força hidrodinâmica gerada no patilhão. A Figura 3.6 apresenta as forças relacionadas com a vela quando o barco navega à bolina.

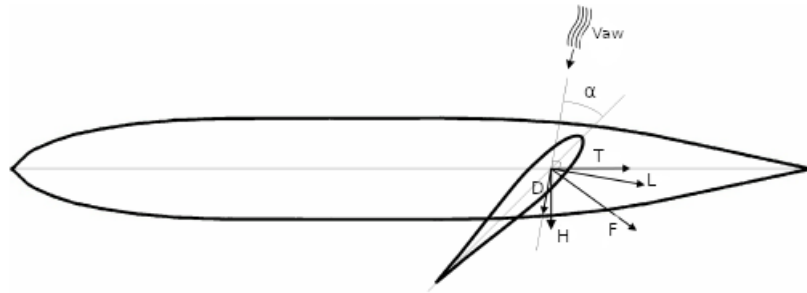


Figura 3.6: Representação das forças exercidas na vela e da força de propulsão e inclinação [8]

Na navegação à bolina é importante ter em consideração a magnitude, a direção e o sentido do vento real ($\vec{v}_{tw}$), do vento aparente ($\vec{v}_{aw}$) e da velocidade do barco ($\vec{v}_b$) [67]. A Equação 3.7 permite relacionar as três velocidades ($\vec{v}_{tw}$, $\vec{v}_{aw}$, $\vec{v}_b$), sendo a Equação 3.4 um caso particular.

$$\vec{v}_{tw} = \vec{v}_{aw} + \vec{v}_b \quad (3.7)$$

Enquanto a força aerodinâmica na vela é responsável pela propulsão no barco, o papel da força hidrodinâmica no casco e seus elementos é manter o barco estável e na trajetória desejada [64]. Em particular, o movimento de inclinação criado na vela pela força aerodinâmica é contrariado pela força hidrodinâmica gerada na interação do patilhão com a água [67], como representado na Figura 3.7, mostrando que a estabilidade do veleiro é assegurada pela componente de sustentação. O perfil hidrodinâmico do patilhão garante que o arrasto hidrodinâmico criado é bastante baixo.

A variação do ângulo de ataque (α) permite aumentar substancialmente a força de sustentação. A Figura 3.8 mostra que, para pequenos ângulos de ataque (tipicamente até 10°), a força de sustentação (*lift*) criada é maior que a força de arrasto

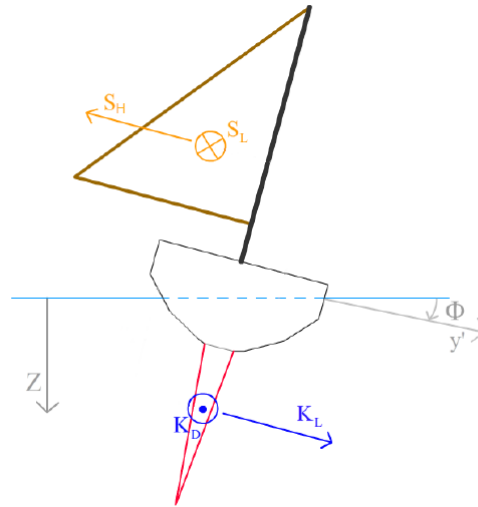


Figura 3.7: Representação da força de inclinação gerada na vela e de sustentação gerada no patilhão [72]

(*drag*) [67]. A relação entre a força de sustentação e de arrasto, além de depender do

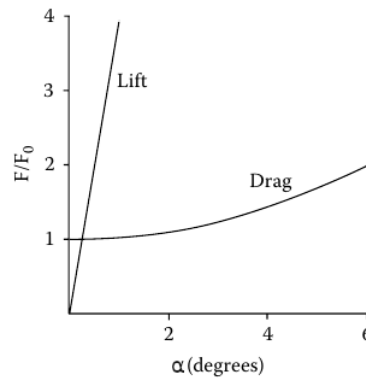


Figura 3.8: Força de sustentação e força de arrasto em função do ângulo de ataque [67]

ângulo de ataque do objeto com o fluido, depende também do perfil alar escolhido, do tamanho, da textura da superfície e do próprio fluido [67].

Outro elemento presente no casco, além do patilhão, é o leme. Um barco à vela, para seguir a trajetória desejada, depende da força de sustentação criada pela interação entre o leme e a água. Tal como acontece no patilhão, quando se muda a orientação do leme (altera-se o ângulo de ataque com a água), produz-se uma força de sustentação que permite direcionar a popa do barco para a trajetória desejada [67]. Em relação ao casco, considera-se que apenas cria uma força resistiva (*drag*) no sistema [73], tanto aerodinâmica como hidrodinâmica.

Assim, um veleiro necessita de aceleração para iniciar o movimento (a força de propulsão $\vec{T}$ gerada pelo vento na vela deve ser superior à força de arrasto produzida pela água no casco) [71] e de satisfazer a Equação 3.8 para ter estabilidade (a força produzida pelo vento deve ser igual e oposta à força produzida pela água) [67].

$$\vec{F}_{wind} + \vec{F}_{water} = 0 \quad (3.8)$$

A Figura 3.9 apresenta as forças aplicadas a um veleiro que navega com estabilidade.

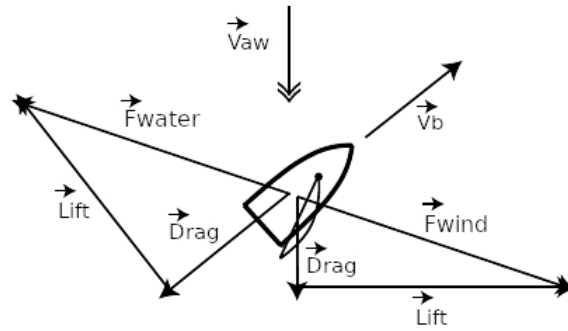


Figura 3.9: Forças exercidas pelo vento e pela água num barco que navega com estabilidade

A capacidade de resistir a variações em *roll*, *pitch* e *heave* de um veleiro depende igualmente da sua altura metacêntrica ($\overline{GM}$) [66]. A $\overline{GM}$ corresponde à distância entre o metacentro (M) e o centro gravítico (G) do barco, podendo ser medida longitudinalmente ou transversalmente em relação ao barco. Por sua vez, o metacentro (M) de um barco é o ponto de interseção de uma linha vertical imaginária, que passa pelo centro de flutuação (B) original, com outra linha vertical imaginária, relativamente ao novo plano e que passa pelo novo centro de flutuação (e.g. quando o barco está inclinado). A Figura 3.10 ilustra esta definição.

Dado que o momento em *pitch* num veleiro é limitado, a altura metacêntrica mais relevante para este tipo de veículo é a transversal [66]. Um barco metacêntricamente estável satisfaz a Equação 3.9. Adicionalmente, uma altura metacêntrica elevada garante estabilidade e robustez face a grandes ondas [66].

$$\overline{GM} \geq \overline{GM}_{min} \geq 0 \quad (3.9)$$

A Figura 3.11 ilustra a relação entre as variáveis que influenciam a estabilidade metacêntrica: a altura metacêntrica ($\overline{GM}$), o metacentro (M), o centro gravítico (G), o centro de flutuação (B) e a posição da quilha (K).

Resumindo, a altura metacêntrica indica a tendência do barco virar de borco.

3.2 Vela Asa

Silva et al. fazem a comparação entre as duas categorias de velas mais usadas em ASV autónomos [22]. A primeira, a vela tradicional, utilizada há mais de 5000 anos pelo ser humano no antigo Egipto [67], e a segunda, a vela asa, semelhante às asas dos aviões [8]. Quando utilizada em ASV autónomos, a vela convencional, apresenta algumas desvantagens em relação à vela asa rígida, sendo estas [22]:

- menos eficiência a navegar na maior parte das situações;
- exige mais atuadores e controlo;
- menos robustez mecânica;
- menor rácio *lift-to-drag*.

A escolha da forma da vela, quer seja tradicional ou asa rígida, é igualmente importante. Para uma mesma área vélica, a forma triangular ou trapezoidal apresenta um centro de pressão aerodinâmico mais baixo que a forma retangular. Assim, a forma triangular ou trapezoidal gera a mesma força de sustentação com um momento de inclinação menor, tornando-se mais estável [8, 69].

Uma vela asa é uma estrutura rígida com um corte transversal em forma de perfil alar (*airfoil*), sendo responsável pela geração da força de propulsão. O ângulo de ataque numa vela asa corresponde ao ângulo entre o vento aparente ($\vec{v}_{aw}$) e a corda média (*chord line*) do perfil alar [8] (ver Figura 3.12).

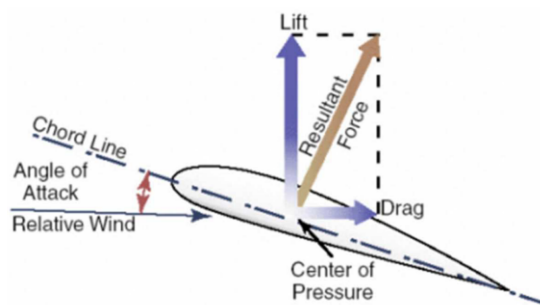


Figura 3.12: Forças presentes na vela asa rígida [11]

O perfil alar pode ser assimétrico, também designado de *cambered* ou curvo, como nas asas dos aviões, ou simétrico, como nas velas asa da maior parte dos ASV. Na Figura 3.13 é possível visualizar a diferença entre os dois tipos de perfil.

Os perfis alares curvos, tipicamente, permitem gerar um maior coeficiente de sustentação que os perfis simétricos [8]. No entanto, num ASV, é desejável navegar da independentemente da direção do vento, resultando na utilização de um perfil

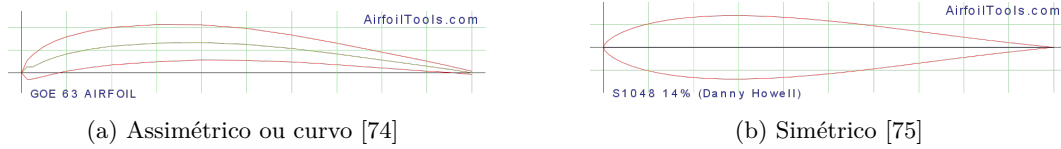


Figura 3.13: Perfis alares

alar simétrico. O desempenho dos *airfoils* simétricos pode ser melhorado através da adição de uma estrutura auxiliar (e.g. cauda ou *canard*), permitindo que a vela asa principal alcance a máxima sustentação possível [61].

3.2.1 Vela Asa com Cauda

No conjunto vela asa com cauda (*tailed wing*), apresentado na Figura 3.14, a vela principal roda livremente à volta do mastro sem criar grandes momentos de inclinação. A cauda permite que a vela principal alcance automaticamente o ângulo ideal em relação ao vento, sendo uma das principais vantagens desta configuração e, por isso, o *design* mais comum nos ASV [61].

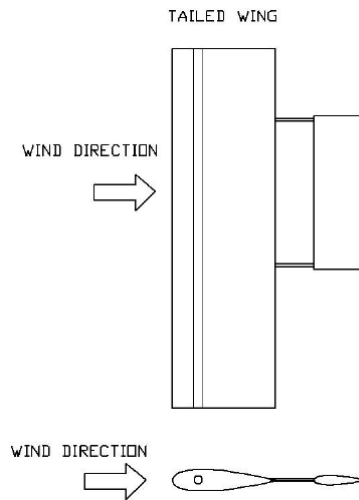


Figura 3.14: Esboço de um conjunto vela asa rígida com cauda [61]

O funcionamento de auto ajuste do conjunto vela asa com cauda em relação ao vento é relativamente simples (Figura 3.15). Enquanto a cauda estiver alinhada com o vento aparente (ângulo de ataque nulo), o conjunto atua como um cata-vento, como é apresentado na Figura 3.15 (superior). A cauda precisa de ser atuada para rodar, definir um ângulo de ataque não nulo com o vento aparente e gerar força de sustentação (ver Figura 3.15 (centro)). Esta ação produz um binário no mastro da vela asa principal, fazendo com que o conjunto (vela asa e cauda) se mova na direção oposta à da cauda, até equilibrar o binário entre a vela asa e a cauda, como é exemplificado na Figura 3.15 (inferior) [8].

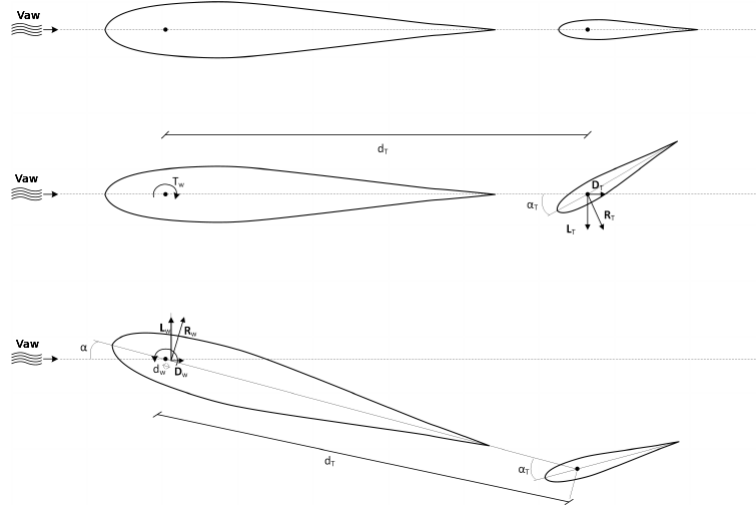


Figura 3.15: Forças e binários presentes no controlo do conjunto vela asa com cauda: cauda não atuada (superior), atuação da cauda (centro), conjunto vela asa e cauda em equilíbrio (inferior) [8]

Este momento de equilíbrio permite à vela asa ter um ângulo de ataque positivo em relação ao vento aparente, criando força de sustentação e, conseqüentemente, propulsionar o barco. Quando a direção do vento altera, a vela asa e a cauda vão rodar para uma nova posição, idêntica relativamente ao vento. Como o *lift* só pode ser gerado perpendicularmente ao vento aparente, o vetor da força de sustentação irá também rodar para a mesma posição relativamente ao vento. Em suma, a vela asa, a cauda e o vetor da força de sustentação irão rodar sempre em conjunto com o vento, como se fossem uma só unidade. O momento de equilíbrio entre a vela asa e a cauda permite à vela asa manter sempre o ângulo de ataque com o vento aparente, até o vento mudar de direção e entrar pelo bordo contrário do barco (estibordo ou bombordo). Neste caso a posição da cauda e, conseqüentemente, a da vela asa tem de ser invertida, correspondendo à manobra de *tacking* ou *jibing* (Anexo A.2) [8].

3.2.2 Vela Asa com *Canard*

O conjunto vela asa com *canard*, presente na Figura 3.16, é muito semelhante ao com cauda, com a diferença da parte auxiliar estar colocada na parte da frente da vela principal. Tem a vantagem, em relação à configuração com cauda, de permitir que o centro de massa da vela esteja melhor posicionado, devido à distribuição do peso. No entanto, este conjunto é bastante mais instável quando a vela principal roda livremente para atingir o ângulo ideal em relação ao vento [61].

Na Figura 3.17 são apresentadas as forças e binários presentes no conjunto vela asa com *canard*, onde sm representa a distância entre o ponto de rotação e o centro aerodinâmico da vela principal.

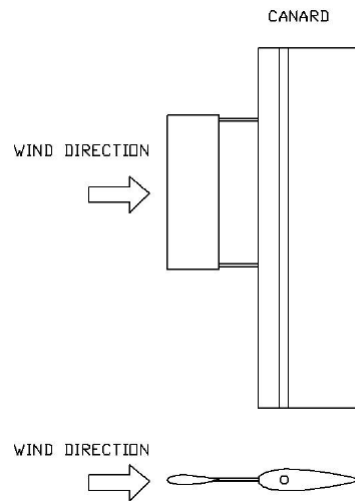


Figura 3.16: Esboço de um conjunto vela asa rígida com *canard* [61]

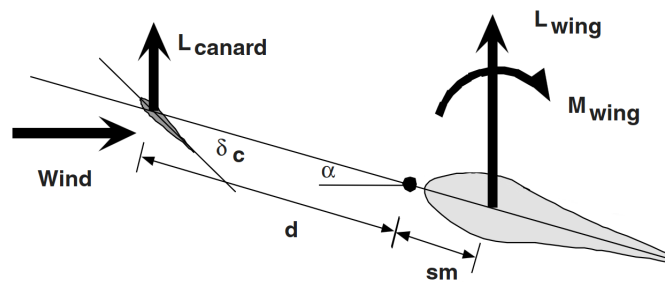


Figura 3.17: Forças e binários presentes no conjunto vela asa com *canard* [76]

3.3 Cascos

Segundo Silva et al. um casco ideal seria capaz de se auto endireitar caso ficasse emborcado, pequeno o suficiente para transportar e prevenir danos em outras embarcações numa eventual colisão, mas também grande o suficiente para navegar em mares agitados [8].

Um veleiro pode ter um mono-casco ou um multi-casco. Os *catamaran* e *trimaran*, ambos multi-casco, tem duas principais desvantagens relativamente aos mono-casco: maior peso e incapacidade de auto endireitar [77]. Os mono-casco podem ser agrupados segundo o LOA em três categorias: (i) inferior a 2 m, (ii) entre 3 m a 5 m e (iii) superior a 5 m. Os cascos pequenos - opção (i) - são baratos, leves, fáceis de transportar, manobrar e de construir. Além disso, não precisam de qualquer infraestrutura ou barco auxiliar para os colocar na água. Por outro lado, são extremamente sensíveis ao vento, ondas e não permitem uma grande capacidade de *payload*, por consequência, um curto período de operação. Os cascos maiores - opção (iii) - tem as características opostas às da opção (i). Os cascos médios - opção (ii) - têm sido cada vez mais adotados pelos ASV, vide o estudo apresentado no capítulo 2 e [8].

3.4 Sumário

Uma vez que se pretende desenvolver uma plataforma marítima de superfície autosustentável, os conceitos e a física inerente à navegação à vela são essenciais. Este capítulo apresenta:

- os elementos principais que constituem um barco à vela;
- as expressões utilizadas para cada zona do casco do barco;
- os graus de liberdade de um barco;
- a física intrínseca na navegação à vela (hidrostática, aerodinâmica, hidrodinâmica);
- os tipos de vela e possíveis configurações;
- os tipos de cascos.

Além da informação apresentada neste capítulo, existe informação complementar no Anexo A relativa a manobras e manobras (*tacking*, *gybing*, *station keeping*) da navegação à vela. Os conceitos apresentados irão suportar a especificação de uma proposta de solução.

Capítulo 4

Estação Marítima Autônoma

4.1 Enquadramento e Problema

Representando aproximadamente 70 % da superfície do nosso planeta [78], o oceano é simultaneamente um vasto *habitat* e uma fonte imensa de recursos energéticos, minerais e biológicos [79], tendo um papel fundamental na regulação climática [80]. Por estas razões, o interesse pela exploração dos oceanos tem vindo a aumentar por parte da comunidade científica, governos e indústria.

Os sistemas autónomos robóticos têm sido cruciais na exploração do ambiente marítimo [81]. Os veículos marítimos autónomos, sejam eles *Autonomous Underwater Vehicle* (AUV) ou *Autonomous Surface Vehicle* (ASV), têm inúmeras aplicações científicas, industriais ou militares [82] porque constituem uma forma segura e eficiente de: (i) alcançar locais inacessíveis ou perigosos para os humanos; e (ii) realizar tarefas complexas repetidamente sem fadiga. No entanto, a realização de missões oceânicas autónomas de longa duração apresenta dois grandes desafios: a comunicação sem fios e a autonomia energética dos veículos.

A comunicação sem fios é um problema que afeta as missões subaquáticas [83]. Dado que a radiofrequência é fortemente atenuada pela água salgada, a comunicação subaquática recorre a ondas acústicas, que apresentam baixa absorção e longo alcance debaixo de água [84]. À superfície, os ASV podem comunicar através de *modems* acústicos com veículos subaquáticos e, utilizando radiofrequência, com o exterior. Neste contexto, uma aplicação possível dos ASV é precisamente servirem de *gateway* de comunicação, como é o caso do *Swordfish* [85].

A autossuficiência energética apresenta-se como o principal fator limitativo da duração e alcance das missões realizáveis por estes *robots*. A maioria da investigação em robótica marítima é direcionada para veículos, subaquáticos ou de superfície, propulsionados por motores elétricos ou de combustão [22]. No entanto, os ASV podem produzir a sua energia a partir das fontes de energia renovável disponíveis, como o sol, a ondulação e o vento, e adotar sistemas eletrónicos de baixo consumo para alcançar a autonomia energética.

Este trabalho foi desenvolvido no LSA, um grupo de Investigação e Desenvolvimento (I&D) do Instituto Superior de Engenharia do Porto (ISEP), dedicado à investigação em sistemas autónomos e áreas afins, como navegação, controlo e coordenação de *robots* [86]. Dado que muitos dos *robots* desenvolvidos no LSA são subaquáticos, é necessário assegurar durante as missões a sua comunicação com o exterior. O presente trabalho pretende modelar, controlar e simular uma estação marítima autónoma de acompanhamento de um *Autonomous Underwater Vehicle*. Em particular, a estação simulada deverá navegar na vizinhança do AUV, quer esteja parado ou em movimento.

4.2 Objetivos

O principal objetivo desta dissertação é estudar a viabilidade de se desenvolver uma estação marítima que:

- seja propulsionada pelo vento;
- navegue com todas as maréações de forma otimizada e autónoma.

Por seu lado, o modelo a criar deverá resultar numa plataforma:

- energeticamente eficiente (minimiza o consumo de energia);
- energeticamente autossuficiente (comporta um sistema de geração de energia);
- estruturalmente robusta (suporta condições meteorológicas adversas).

4.3 Análise de Requisitos

Tendo em consideração os objetivos apresentados, o ASV deve satisfazer, por ordem de prioridade, os seguintes requisitos:

1. ser livre de âncoras;
2. navegar por *waypoints* autonomamente e otimizando a trajetória;
3. permanecer numa determinada área, ou seja, executar manobra de *station keeping*;

4. ser capaz de transportar o equipamento de base necessário para operar;
5. ser capaz de transportar o equipamento científico adicional;
6. gerar energia para alimentar o sistema eletrónico;
7. ser robusto para aguentar condições atmosféricas e marítimas adversas;
8. realizar missões por longos períodos de tempo, idealmente, pelo menos 1 ano.

4.4 Proposta de Solução

Com base na pesquisa apresentada anteriormente e tendo em consideração os objetivos e requisitos definidos para este projeto, pretende-se propor uma solução para uma estação marítima autónoma.

No capítulo 2 foram analisados diversos ASV propulsionados por motor, vento e ondulação. Uma vez que se pretende que a estação marítima seja autossustentável para realizar missões de longa duração, a pesquisa focou-se no aproveitamento da energia eólica e da ondulação. Optou-se por uma plataforma propulsionada pelo vento com um *design* que: (i) minimiza o consumo energético; (ii) suporta a instalação de sistemas de geração de energia; e (iii) é estruturalmente robusto.

O dimensionamento do ASV deve ser tal que permita carregar toda a eletrónica necessária – atuadores, sensores e sistemas de alimentação, computacional e de comunicação – e oferecer robustez para suportar condições atmosféricas e marítimas adversas. Outro aspeto a ter em conta é o transporte e toda a logística necessária para colocar a plataforma na água, fator limitante nas dimensões do ASV. Pela informação apresentada na Secção 3.3 um casco com um LOA entre 3 m e 5 m permite cumprir estes requisitos, não comprometendo a robustez da plataforma, mas também permitindo que o seu transporte seja possível sem grandes dificuldades, recorrendo a uma infraestrutura de auxílio.

Na Figura 4.1 pode-se observar o modelo 3D conceptual da estação marítima autónoma proposta.

4.4.1 Casco

Para a seleção do casco considerou-se principalmente dois fatores, (i) o *payload* que o ASV tem de suportar e (ii) ser robusto o suficiente para aguentar condições climáticas adversas.

O casco projetado é baseado nos cascos da classe internacional 2.4mR [87], que além de serem ágeis e estáveis, são projetados para navegar em condições climáticas adversas. Estes cascos geralmente são construídos em fibra de vidro e não permitem que o veleiro emborque, devido ao seu patilhão feito em chumbo [88]. Normalmente, os veleiros desta classe tem um casco com 4,16 m de LOA, um mastro

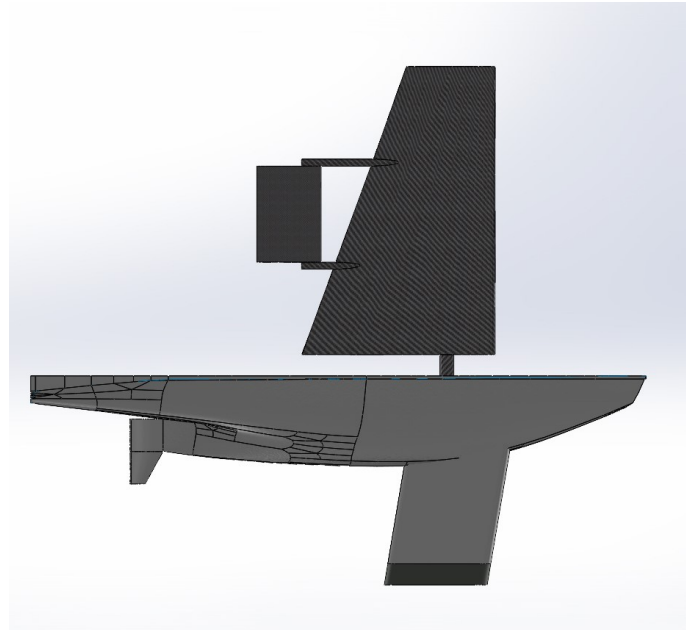


Figura 4.1: Estação marítima autónoma proposta

com 5 m de altura e uma vela com área de $7,5 \text{ m}^2$ [87], informação relevante para o dimensionamento da vela neste projeto.

As dimensões do casco utilizado são apresentadas na Figura 4.2, tendo um LOA de 4,36 m.

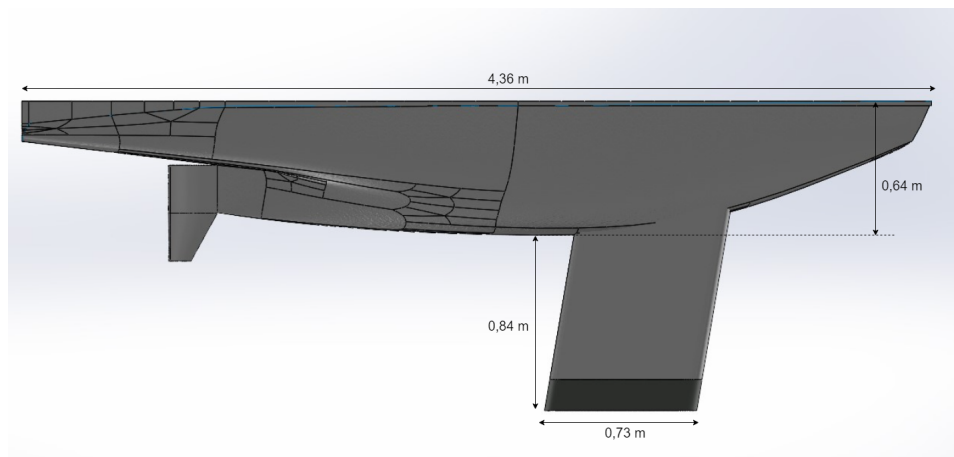


Figura 4.2: Dimensões do casco projetado

4.4.2 Vela Asa com Cauda

Uma vez definido o tipo de veículo e o seu casco, um ASV impulsionado pelo vento com mono-casco, falta definir o tipo de estrutura que permitirá a propulsão do *robot*. Considerando a informação da Secção 3.2 foi então escolhido, para impulsionar o

veículo, uma vela asa rígida trapezoidal com perfil simétrico e uma cauda, uma vez que este sistema permite reduzir o consumo de energia na atuação da vela asa.

Seleção do Perfil Alar

Na seleção do perfil alar o objetivo é conseguir obter o maior coeficiente de sustentação e a maior relação entre este coeficiente e o coeficiente de arrasto, não comprometendo a rigidez da vela asa, ou seja, também não deve ter uma espessura demasiado fina [8]. Outro ponto a ter em consideração na escolha do perfil alar é o efeito do número de Reynolds (R_e). O R_e , dado pela Equação 4.1, representa a relação entre as forças cinemáticas ou inerciais e as forças de viscosidade de um fluido, sendo ρ_{air} a densidade do meio, $\vec{v}_f$ a velocidade do fluido, L_c o comprimento característico e μ a viscosidade do meio [17].

$$R_e = \frac{\rho_{air} \vec{v}_f L_c}{\mu} \quad (4.1)$$

Elkaim em [17] refere ainda que o R_e adequado para a navegação à vela está entre 200 000 e 1 200 000.

Na literatura já foram utilizados vários perfis como o:

- Eppler E169 por Silva et al. [8],
- Eppler E479 por Miranda [89],
- NACA 0021 por Enqvist [61],
- NACA 0018 por Tretow [64].

Na seleção do perfil alar, todos consideraram as características aerodinâmicas dos vários perfis para valores do R_e baixos, entre 200 000 e 350 000.

Tendo em consideração todos os conceitos enunciados e os perfis alares utilizados nos projetos já referidos, no presente trabalho foi escolhido o *airfoil* Eppler E473 [90], possível de visualizar na Figura 4.3.

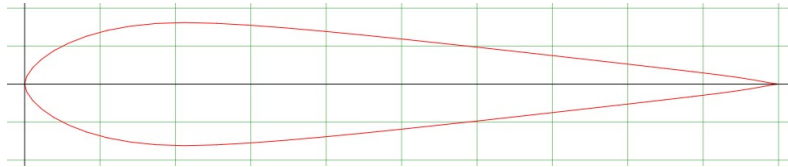


Figura 4.3: Perfil alar Eppler E473 [90]

As características aerodinâmicas deste perfil, para um $R_e = 200\,000$, são apresentadas na Figura 4.4. Analisando a Figura 4.4c, é possível verificar que para valores do ângulo de ataque entre 5° a 15° o perfil alar tem uma relação $C_L/C_D > 30$.

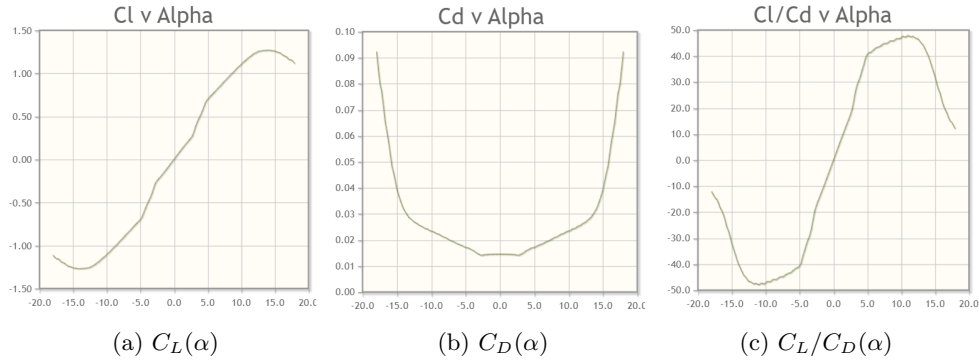


Figura 4.4: Principais características aerodinâmicas do perfil alar Eppler E473 [90]

Design

Com o perfil alar (Eppler E473) e a forma geométrica da vela (trapezoidal) definida é necessário dimensionar toda a estrutura (vela asa com cauda). Considerando que, tipicamente, numa vela tradicional o $C_L \approx 0.6$ e a relação C_L/C_D está entre 3 e 5 [17], a relação entre o perfil alar escolhido, para um α de 10° , e uma vela tradicional é dada pela Equação 4.2.

$$Ratio = C_{LE473}/C_{Ltradicional} \approx 1.84 \quad (4.2)$$

Este valor permite relacionar a área entre as duas velas, indicando que a área da vela asa deverá ser 1.84 vezes menor do que a área da vela tradicional. Como referido na Subsecção 4.4.1, um barco com um casco semelhante ao definido para este projeto tem uma área vélica de $7,5 \text{ m}^2$, resultando numa área para a vela asa rígida de $\approx 4,08 \text{ m}^2$. No entanto, uma vez que a relação C_L/C_D do perfil alar selecionado é consideravelmente superior à da vela tradicional, foi decidido que a área da vela asa principal do ASV projetado seria de $\approx 2,04 \text{ m}^2$. A forma geométrica da vela asa é um trapézio, como já mencionado, e a da cauda um retângulo. A cauda tem um perfil alar igual ao da vela asa principal (Eppler E473) e uma área vélica de $\approx 0,31 \text{ m}^2$. Na Figura 4.5 são apresentadas as dimensões do conjunto projetado.

Segundo Enqvist em [61], o mastro num *airfoil* simétrico deve ser posicionado um quarto da corda média atrás do bordo de ataque, coincidindo com o seu centro aerodinâmico e de pressão. Para a projeção e dimensionamento do conjunto vela asa e cauda, além do trabalho de Enqvist, foi também considerado o de Tretow [64], Silva et al. [8] e Miranda [89].

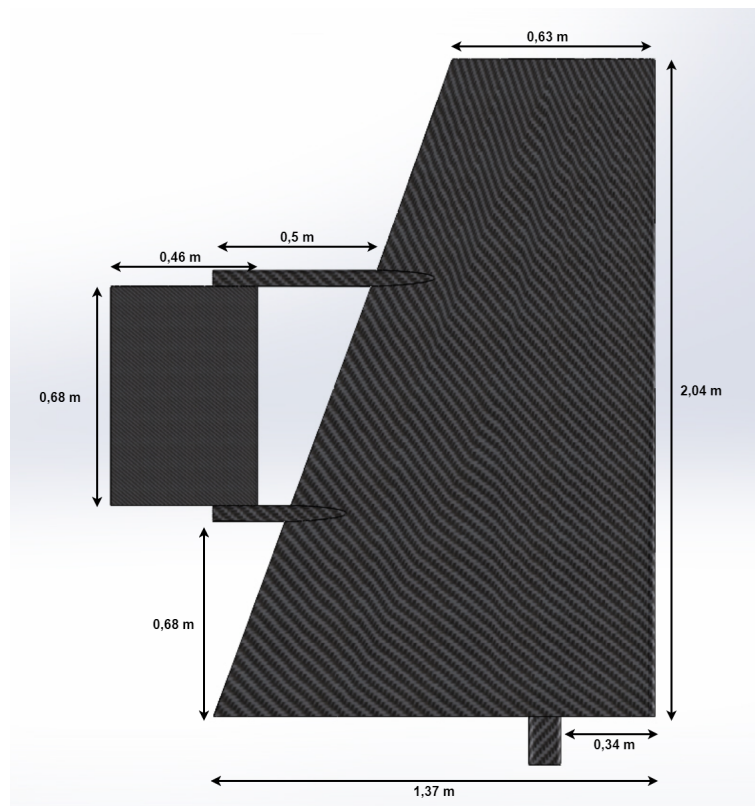


Figura 4.5: Dimensões do conjunto vela asa e cauda projetado

Tabela 4.1: Ficha técnica da estação marítima modelada

LOA	4,36 m
Largura	0,80 m
<i>Draft</i>	1,28 m
Material do casco	Fibra de vidro
Massa	515,71 kg
<i>Payload</i>	88,21 kg
Propulsão	Vento
Tipo de vela	Asa rígida com cauda
Perfil alar	Eppler E473
Área vélica	2,35 m ²
Material da vela	Fibra de carbono

4.4.3 Ficha Técnica

Os materiais do casco e da vela foram escolhidos considerando que a plataforma deve ser robusta e aguentar condições atmosféricas adversas, optando assim por materiais resistentes.

O valor do *payload* foi calculado através da Equação 4.3, onde F_B representa a força de flutuação gerada pelo volume do fluido deslocado pelo veleiro (Equação 3.1) e $\vec{F}_W$ a força do peso do veleiro (Equação 3.2). O cálculo da $\vec{F}_W$ considerou apenas o peso do casco, vela asa, cauda, leme e quilha, excluindo qualquer equipamento eletrónico e eléctrico.

$$\vec{F}_P = \vec{F}_B - \vec{F}_W \quad (4.3)$$

A aceleração gravítica ($g = 9,8 \text{ m/s}^2$) e a densidade do fluido (ρ_f) são conhecidas, uma vez que o fluido é a água do mar ($\rho_f = 1023,6 \text{ kg/m}^3$), faltando apenas calcular o volume submerso (V_{sub}). O volume total (V_t) do casco, da quilha e do leme obtido no SolidWorks é de $\approx 0,85 \text{ m}^3$, mas nem todo o casco está submerso. Considerando que o *draft* da plataforma é de $\approx 1,28 \text{ m}$, a altura do casco fora de água é de $0,2 \text{ m}$, obtendo no um volume emerso (V_e) de $\approx 0,26 \text{ m}^3$. O volume submerso (V_{sub}) da plataforma é calculado através da Equação 4.4.

$$V_{sub} = V_t - V_e \quad (4.4)$$

Substituindo os valores obtidos na Equação 3.1, resulta que $F_B = 5918,5 \text{ N}$. De igual forma, substituindo os valores relativos à Equação 4.3, obtém-se que a força máxima que o barco suporta para o *draft* definido é $F_P = 864,50 \text{ N}$. O *payload* resultante, que corresponde à massa adicional que o veículo consegue suportar nas condições definidas, é de $88,21 \text{ kg}$. Este valor foi obtido pela divisão da F_P pela componente da aceleração gravítica.

4.5 Sumário

Neste capítulo revisitou-se o problema, detalhando-se os objetivos do projeto e, consequentemente, os seus requisitos, e propôs-se uma solução. A plataforma proposta é um veleiro com um casco de LOA de $4,36 \text{ m}$ em fibra de vidro, inspirado no modelo da classe internacional 2.4mR, e um conjunto de velas asas rígidas em fibra de carbono com perfil alar simétrico Eppler E473, composto por vela principal e cauda

com área vélica de $2,04\text{m}^2$ e $0,31\text{m}^2$, respetivamente. O veleiro projetado tem de massa 516kg e suporta um *payload* de 88kg .

Capítulo 5

Simulação

Este capítulo descreve a integração do Gazebo com o ROS, a importação do modelo do veleiro para o Gazebo e os algoritmos de controlo e navegação autónoma por *waypoints* e *station keeping* desenvolvidos para simular o comportamento do veleiro. Para os menos familiarizados com o tópico, no Anexo B é apresentada uma análise dos *softwares* utilizados, assim como alguns *plugins* do Gazebo.

5.1 Packages

A integração Gazebo com o ROS é efetuada através da reutilização do *metapackage* `gazebo_ros_pkgs`, que fornece as interfaces necessárias para simular um *robot* no Gazebo a partir do ROS. Foram desenvolvidos os seguintes quatro *packages* ROS para simular a estação marítima autónoma no Gazebo:

- `tese_sim`, que contém o ficheiro `.launch`, carrega o mundo do Gazebo e o modelo do *robot* especificados;
- `liftdrag_plugin`, que implementa o *plugin* descrito na Subsecção B.4.1, simula as forças aerodinâmicas ou hidrodinâmicas aplicadas num perfil alar;
- `sailboat_description`, que define o modelo do veleiro apresentado na Secção 4.4, define através de vários ficheiros `.xacro` todos os *links* e *joints* que compõem o *robot*, assim como os *plugins* que interagem com o *robot* ou simulam os sensores utilizados;

- `sailboat_control`, que implementa os *plugins* `flap_control` e `usv_control` referidos na Secção B.4, é responsável pelo controlo da estação marítima autónoma.

A Figura 5.1 apresenta o diagrama de blocos da simulação que compreende o Gazebo, a interface entre o Gazebo e o ROS, o ROS e os *packages* desenvolvidos.

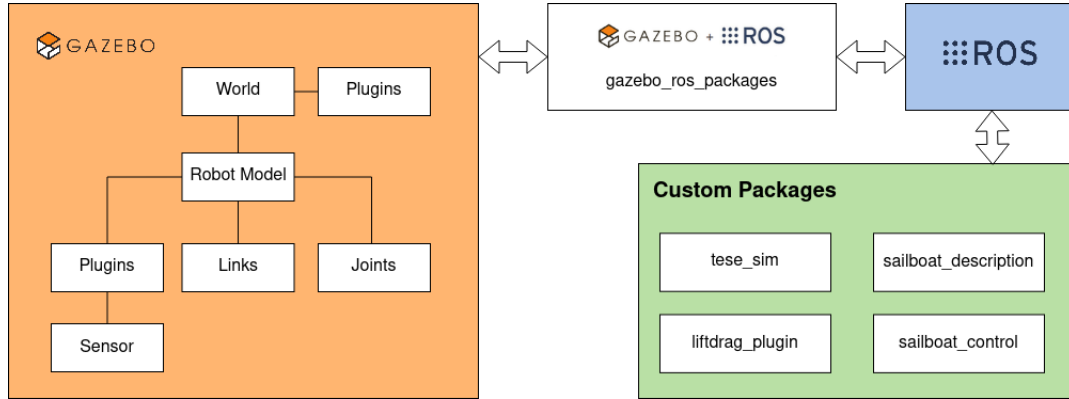


Figura 5.1: Diagrama de blocos da simulação

5.2 Modelo do Veleiro

O *package* `sailboat_description` é responsável pela descrição do modelo do *robot*. O ficheiro `sailboat.xacro` carrega todos os ficheiros `.xacro` do modelo do veleiro, incluindo as características físicas, a interação com o mundo e os sensores. É o caso do ficheiro `sailboat.urdf.xacro`, que define o modelo 3D do *robot* através de *links* e *joints*. Cada *link* possui as suas propriedades inerciais, de colisão e visuais, como apresentado no Anexo C.1. A Figura 5.2 apresenta o modelo implementado, assim como a posição e orientação das *joint* que conectam os vários *links*. A estação marítima autónoma apresenta no simulador o aspeto apresentado na Figura 5.3.

5.3 Controlo

O *package* `sailboat_control` implementa o controlo da estação marítima autónoma através do *plugin* `usv_control`. Este *plugin* controla o ângulo de ataque da vela asa através da cauda e a direção do USV através do leme.

5.3.1 Controlo da Vela Asa

De forma a minimizar o consumo de energia, a definição do ângulo de ataque da vela asa é feito através do controlo da cauda. Assim, foi desenvolvido um controlador PID que tem como entradas o ângulo de ataque atual (α_a) e o ângulo de ataque desejado ($\alpha_d = 10^\circ$). O valor do ângulo de ataque desejado para o perfil alar E473

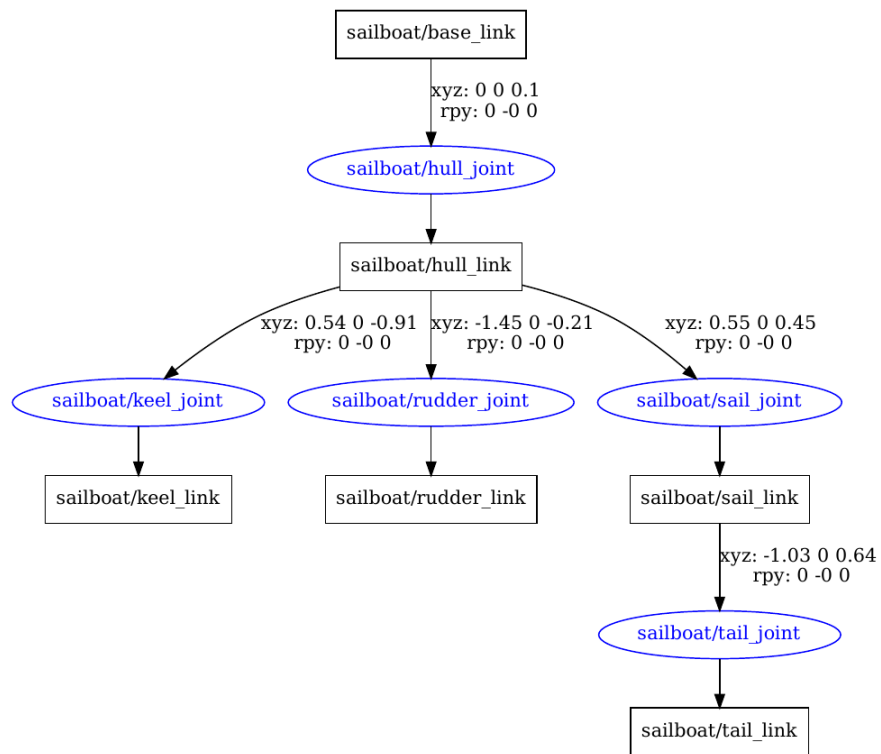
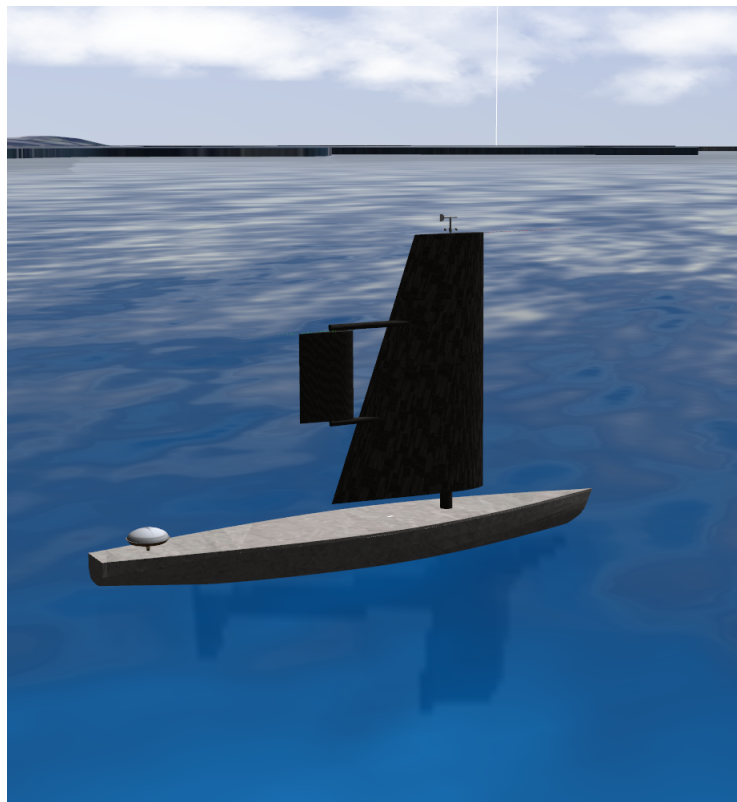
Figura 5.2: Diagrama do modelo do *robot* implementado

Figura 5.3: Estação Marítima Autónoma no Gazebo

foi obtido através da análise da Figura 4.4c. O ângulo de ataque atual é calculado através do produto cruzado dos vetores que caracterizam o vento aparente ($\vec{v}_{aw}$) e a orientação da vela asa ($\vec{s}$), usando a Equação 5.1.

$$\begin{aligned} \|\vec{v}_{aw} \times \vec{s}\| &= \|\vec{v}_{aw}\| \|\vec{s}\| \sin(\alpha_a) \Leftrightarrow \\ \Leftrightarrow \alpha_a &= \sin^{-1} \left(\frac{\|\vec{v}_{aw} \times \vec{s}\|}{\|\vec{v}_{aw}\| \|\vec{s}\|} \right) \end{aligned} \quad (5.1)$$

O produto cruzado é preferível face ao produto escalar porque retorna o ângulo entre os dois vetores e o sentido de rotação da vela asa em relação ao vento. O ângulo de ataque é positivo se a vela rodar no sentido horário e negativo se rodar no sentido anti-horário em relação ao vento. Na Figura 5.4 é apresentado o controlador PID implementado para a posição angular cauda, onde $e(t)$ é o erro entre o ângulo de ataque desejado e o atual, $\int_0^t e(\tau) d\tau$ o erro integral e $\frac{de(t)}{dt}$ o erro derivativo. K_p , K_i e K_d são os ganhos proporcional, integral e derivativo, respetivamente.

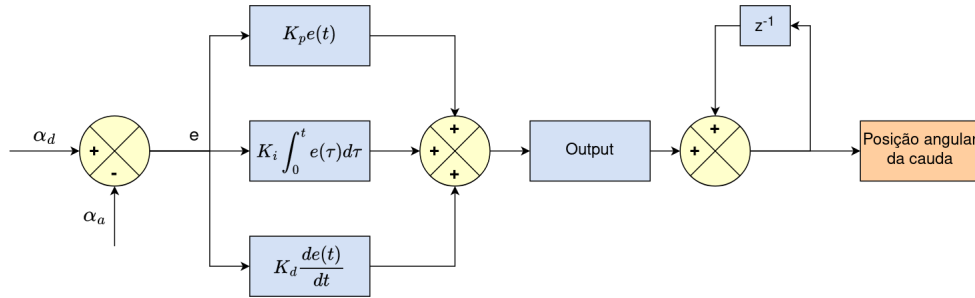


Figura 5.4: Controlador PID da cauda

Este controlador é implementado na função `MainSail_Control()` definida no *plugin* `usv_control`. Esta função, além da implementação do controlador PID, é responsável pela rotina de controlo do ângulo de ataque entre a vela asa e o vento aparente, atuando apenas na cauda. Na Figura 5.5 é possível observar o fluxograma da função. O limite para a posição angular da cauda foi definido de $\pm 30^\circ$. Quando a missão for concluída, ou seja, o veículo atingir o último *waypoint*, a posição angular da cauda fica a 0° , alinhando com o vento e aguardando uma nova missão.

5.3.2 Controlo do Leme

O controlo da direção da estação marítima autónoma é realizado através do leme. Foi desenvolvido um controlador PD para controlar a posição angular do leme que tem como entradas a orientação atual do USV (ψ) e a orientação desejada (ψ_d), considerando as coordenadas do *waypoint* (W_x, W_y) a atingir e da posição atual do USV (B_x, B_y). A orientação desejada ψ_d é calculada através da Equação 5.2, onde

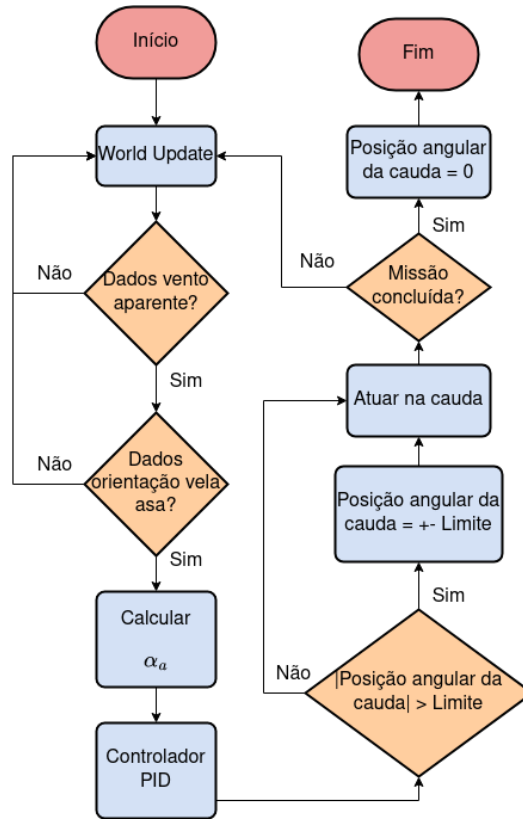


Figura 5.5: Fluxograma da função MainSail_Control()

D_{bw_x} e D_{bw_y} correspondem à distância entre as coordenadas do *waypoint* e do barco, respetivamente.

$$\psi_d = \text{atan2}(D_{bw_y}, D_{bw_x}) \quad (5.2)$$

$$D_{bw_x} = W_x - B_x$$

$$D_{bw_y} = W_y - B_y$$

Na Figura 5.6 é apresentado o diagrama de blocos do controlador PD implementado para o leme, onde $e(t)$ é o erro entre a orientação da plataforma (ψ) e a orientação desejada (ψ_d) dentro do intervalo $[-180^\circ, 180^\circ]$, $\frac{de(t)}{dt}$ é o erro derivativo, e K_p e K_d são os ganhos proporcional e derivativo.

Este controlador está implementado na função `Navigation_Control()` do *plugin usv_control1*. Esta função altera o modo de navegação em função do ângulo entre a rota para o *waypoint* corrente e o vento. Assim, quando este ângulo se aproxima ou corresponde à área de não navegação do diagrama polar, a plataforma tem de

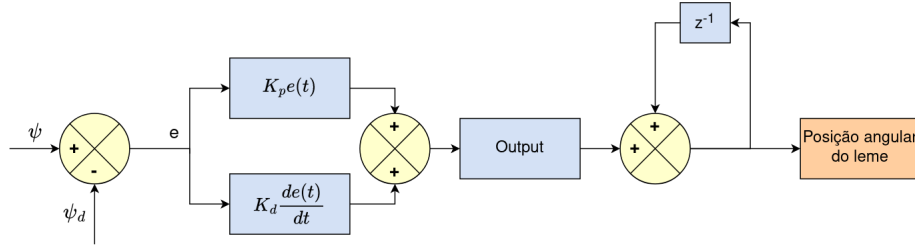
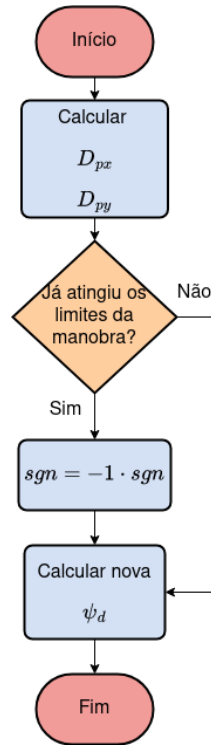


Figura 5.6: Controlador PD do leme

ziguezaguar para avançar em direção ao *waypoint*. Esta manobra designa-se viragem por davante ou *tacking* se for contra o vento e viragem em roda ou *gybing* se for a favor do vento. Nestes casos, a função ativa o modo genérico de viragem ou *turning* e, nos restantes casos, permanece no modo normal de navegação, calculando e aplicando ψ_d ao controlador PD. Na Figura 5.7 é apresentado o fluxograma da sub-rotina `Turning()`.

Figura 5.7: Fluxograma da sub-rotina `Turning()`

Esta sub-rotina aplica a Equação 5.3 para determinar a nova orientação desejada (ψ_d), onde θ_{tw} é a direção do vento real, sgn varia entre 1 e -1 dependendo da direção desejada e δ corresponde ao ângulo da zona de não navegação.

$$\psi_d = \theta_{tw} + sgn \delta \quad (5.3)$$

Sempre que a distância percorrida no bordo corrente excede a distância máxima definida, *sgn* muda o sinal permitindo ao veículo fazer uma trajetória em zigue-zague. A distância percorrida pelo USV no bordo corrente D_p é calculada através da Equação 5.4, onde (B_x, B_y) são as coordenadas da posição corrente do veleiro e (P_x, P_y) as coordenadas da posição da última viragem de bordo.

$$D_p = \sqrt{(B_x - P_x)^2 + (B_y - P_y)^2} \quad (5.4)$$

A decisão de se definir a distância máxima a percorrer por bordo em vez de se minimizar o número de viragens foi tomada pensando na manobra de *station keeping*.

Por fim, a função `Navigation_Control()` verifica ainda se o USV já atingiu o *waypoint* corrente e, em caso afirmativo, avança para o próximo. Neste contexto, considera-se que a plataforma atinge um *waypoint* quando se encontra a uma distância máxima de 10 m. A distância D_{bw} entre o barco e o *waypoint* é obtida através da Equação 5.5.

$$D_{bw} = \sqrt{D_{bw_x}^2 + D_{bw_y}^2} \quad (5.5)$$

A Figura 5.8 contém o fluxograma desta função. O limite definido para a posição angular do leme foi $\pm 45^\circ$.

5.4 Algoritmo de Station Keeping

A manobra de *station keeping* pretende manter o USV numa área de raio r centrada num *waypoint* $W = (W_x, W_y)$. Para tal, são calculados quatro novos *waypoints*, igualmente espaçados, localizados à distância r e a 45° de W (Equação 5.6).

$$\begin{aligned} W_1 &= (W_x + r \cos 45^\circ, W_y - r \sin 45^\circ) \\ W_2 &= (W_x + r \cos 45^\circ, W_y + r \sin 45^\circ) \\ W_3 &= (W_x - r \cos 45^\circ, W_y - r \sin 45^\circ) \\ W_4 &= (W_x - r \cos 45^\circ, W_y + r \sin 45^\circ) \end{aligned} \quad (5.6)$$

A Figura 5.9 apresenta a manobra de *station keeping* com os círculos azuis a representar os *waypoints*, a vermelho a trajetória teórica da plataforma e a tracejado laranja a área de *station keeping*.

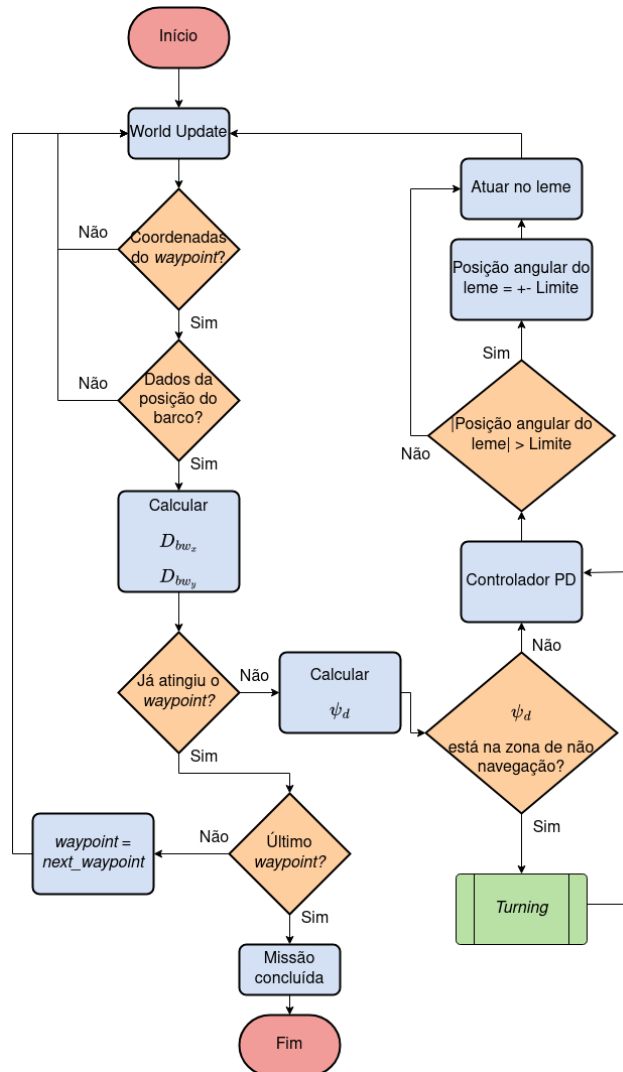


Figura 5.8: Fluxograma da função Navigation_Control()

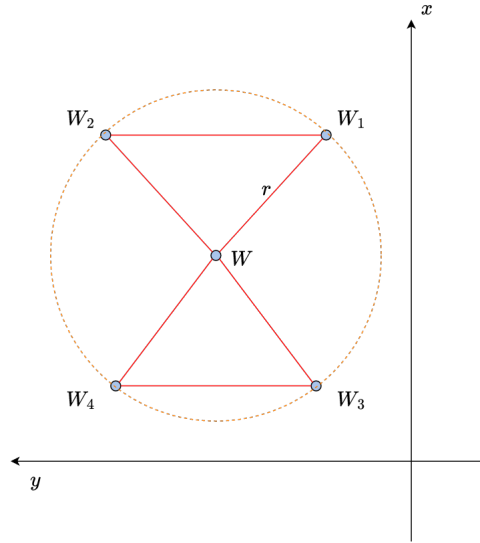


Figura 5.9: Esboço da manobra de *station keeping*

A manobra de *station keeping* implementada consiste num ciclo infinito de passagem pelos cinco *waypoints* definidos, começando por W . A sequência de passagem definida – $[W, W_1, W_2, W, W_3, W_4, W]$ – garante que o ASV regressa ao centro W após visitar, por ordem numérica, cada par dos restantes *waypoints*.

5.5 Sumário

Neste capítulo descreveu-se o modelo 3D da plataforma marítima autónoma e a sua importação para o Gazebo assim como os *packages* de suporte à simulação reutilizados e desenvolvidos. Em particular, foi detalhado o sistema de controlo – PID do ângulo de ataque da vela asa através da cauda e PD da direção através do leme – e de navegação – as funções `MainSail_Control()`, `Navigation_Control()` e de *station keeping* – realizados. Este é o conjunto de software que permitiu simular e caracterizar o desempenho da plataforma marítima autónoma.

Capítulo 6

Testes e Resultados

Este capítulo apresenta os testes realizados para validar o comportamento da plataforma desenvolvida, os resultados obtidos na simulação e a sua respetiva análise.

6.1 Alinhamento da Vela Asa ao Vento

Uma vela asa sem controlo ($\alpha = 0^\circ$) deve-se alinhar com o vento. Se, por exemplo, o vento estiver a 90° , sentido este-oeste, e a vela asa tiver uma orientação inicial de 0° , a vela deverá, após um período transitório, alcançar a orientação final de -90° (alinhada ao vento). A Figura 6.1 documenta as várias fases do teste realizado. O vento, com uma velocidade de 8 m/s, e a vela asa estão com as condições enunciadas. A Figura 6.1a corresponde ao instante inicial da simulação ($t = 0$ s). As Figuras 6.1b e 6.1c, que correspondem aos instantes $t = 7$ s e $t = 15$ s, ilustram o período transitório em que a vela oscila em busca do equilíbrio. A Figura 6.1d apresenta o estado final de equilíbrio. Na ausência de perturbações, a partir de $t = 20$ s, a vela apresenta um ângulo de ataque $\alpha = 0^\circ$. Nas imagens apresentadas a seta azul indica a orientação do vento e a linha tracejada vermelha a orientação da vela.

Na Figura 6.2 são apresentados os resultados do alinhamento da vela asa para várias velocidades do vento (3 m/s, 5 m/s e 8 m/s). Analisando os resultados, é possível concluir que a vela asa alinha-se com o vento ($\alpha = 0^\circ$) ao final de 17,5 s, com uma margem de erro de $\pm 1^\circ$.

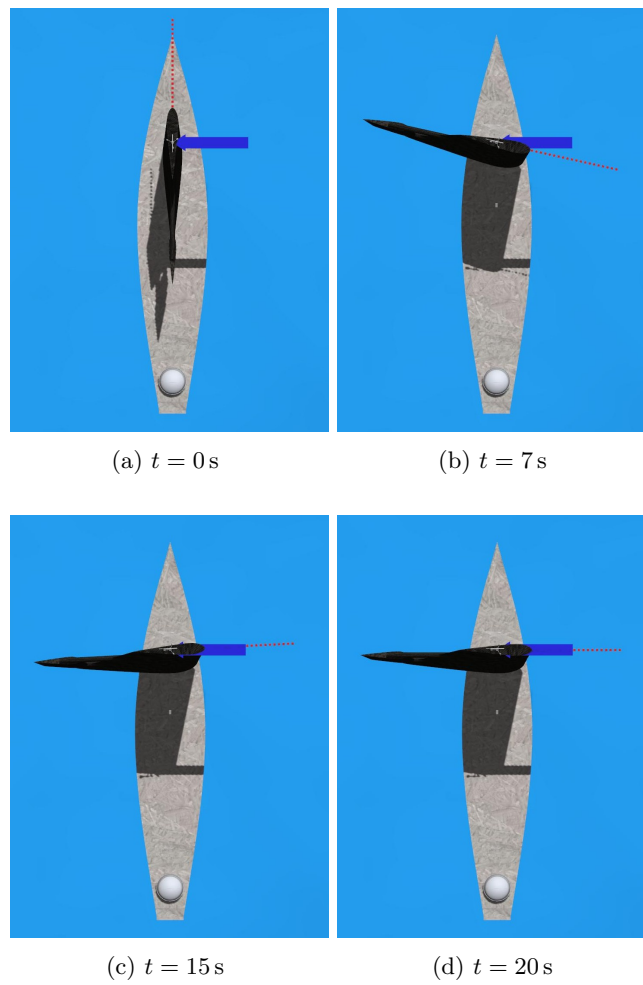


Figura 6.1: Alinhamento da vela asa ao vento sem controlo

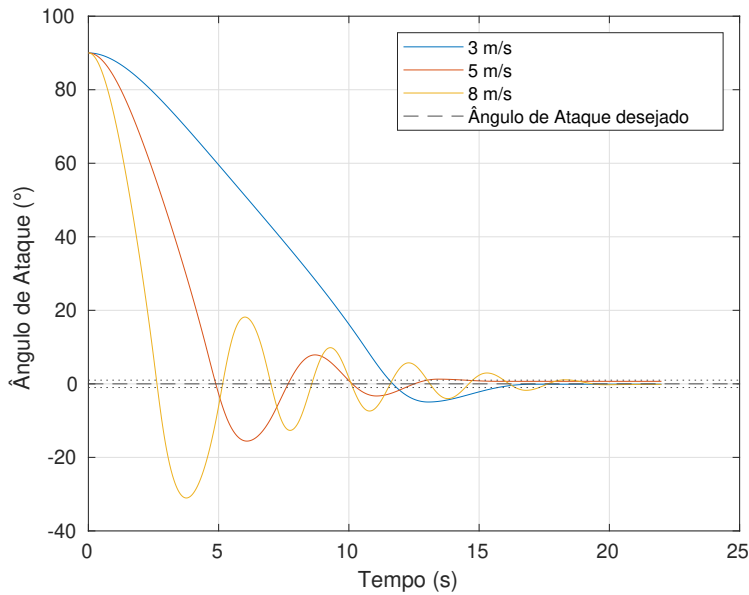


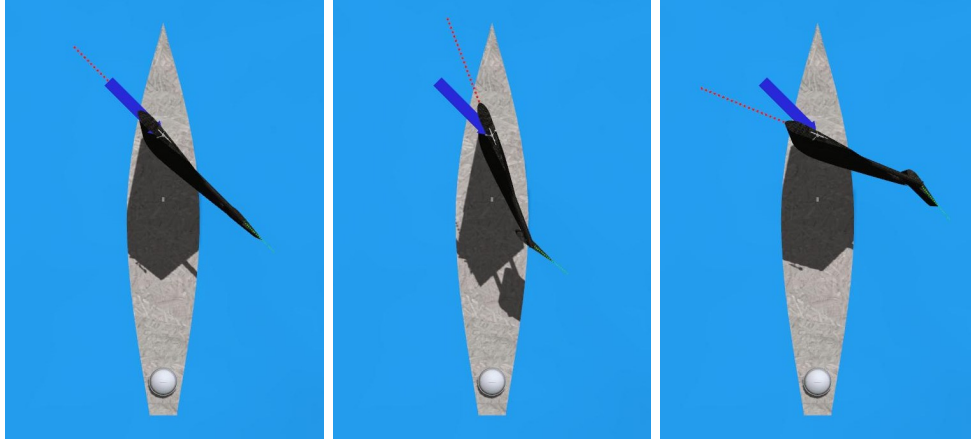
Figura 6.2: Resposta temporal do alinhamento da vela asa para diferentes velocidades do vento

6.2 Controlo da Vela Asa

Estes testes pretendem confirmar o correto controlo do ângulo de ataque da vela asa através da cauda (Subsecção 3.2.2 na Figura 3.15).

O primeiro teste realizou-se com vento a -45° , sentido noroeste-sudeste, e velocidade de 8 m/s. Atuou-se na cauda através do tópico `/sailboat/control_flap`. A Figura 6.3 apresenta os resultados: (i) sem atuar na cauda o conjunto está alinhado com o vento (Figura 6.3a); (ii) aplicando um ângulo positivo à cauda, a vela asa roda no sentido negativo (Figura 6.3b); e (iii) aplicando um ângulo negativo na cauda, a vela asa roda no sentido positivo (Figura 6.3c). Nas três figuras apresentadas a seta azul indica a orientação do vento, a linha tracejada vermelha a orientação da vela e a linha tracejada verde a orientação da cauda.

Uma vez validado o correto funcionamento do conjunto de vela asa e cauda, validou-se o controlo do ângulo de ataque da vela asa ($\alpha_d = 10^\circ$) através da cauda com um vento de -90° , sentido oeste-este, e de 8 m/s. A Figura 6.4 mostra a resposta temporal do controlo implementado com diferentes ganhos. Com base nestes resultados, os restantes testes utilizam $K_p = 0.0003$ e $K_d = 0.0001$. A Figura 6.5 apresenta a resposta temporal com diferentes velocidades do vento. Tal como era de esperar, quanto menor a velocidade do vento, maior é o tempo que a vela asa leva para alcançar o ângulo de ataque desejado ($\alpha_d = 10^\circ$).



(a) Vela asa e cauda alinhados (b) Cauda com ângulo positivo com o vento (c) Cauda com ângulo negativo

Figura 6.3: Resposta da vela asa para diferentes ângulos da cauda

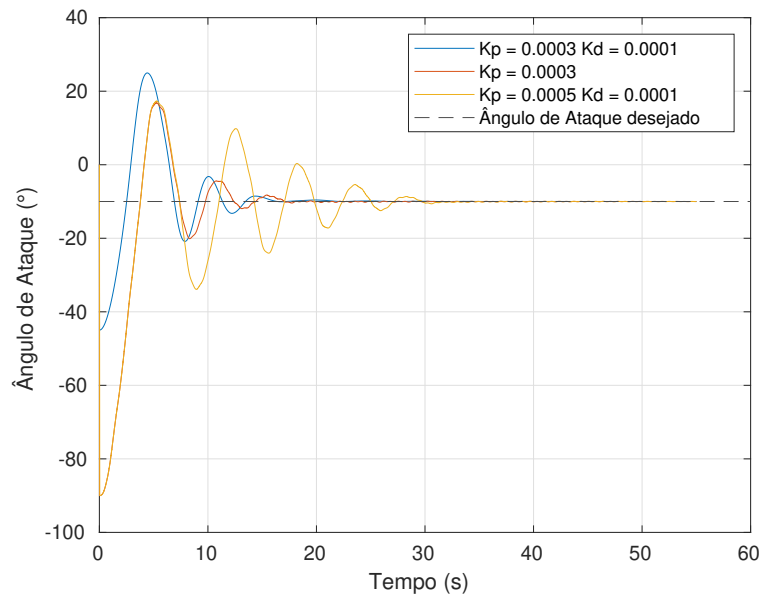


Figura 6.4: Resposta temporal para diferentes ganhos do controlo implementado para o ângulo de ataque da vela asa

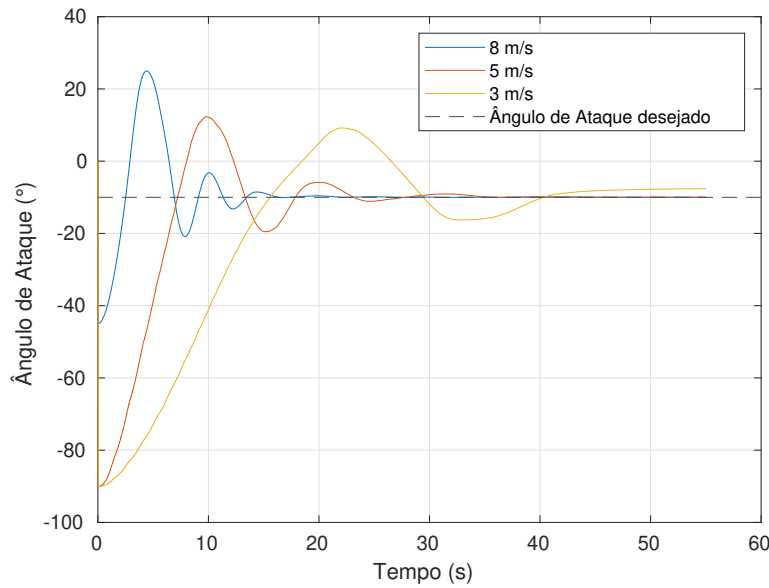


Figura 6.5: Resposta temporal para diferentes velocidades do vento do controlo implementado para o ângulo de ataque da vela asa

6.3 Diagrama Polar

O diagrama polar de um veleiro caracteriza a sua velocidade com diferentes condições de vento (direção e velocidade). A Figura 6.6 foi obtida com a direção do vento a variar entre 0° a 360° em relação à direção de deslocamento do veleiro e com velocidades de 6 m/s, 8 m/s e 14 m/s. Estas velocidades ilustram, grosso modo, a gama operacional do veleiro em termos de condições de vento, e correspondem à velocidade mínima ($\pm 0,6$ m/s), intermédia (± 1 m/s) e máxima ($\pm 2,6$ m/s) de deslocamento do veleiro. Tal como era esperado, o diagrama mostra que: (i) independentemente da intensidade do vento, a manobra em que o veleiro é mais veloz é ao largo (ângulo do vento relativo ao barco de $\pm 90^\circ$; e (ii) o veleiro, em manobras idênticas, atinge maior velocidade com vento mais forte.

6.4 Controlo de Direção

Estes testes pretendem validar o controlo da direção da estação marítima autónoma através do leme e, consequentemente, percorrer uma trajetória definida por uma lista de *waypoints*.

O primeiro teste, ilustrado na Figura 6.7, pretende validar o seguimento de *waypoints* no modo normal de navegação. A direção do vento é de 90° , sentido este-oeste, e a velocidade de 10 m/s. As coordenadas da posição inicial do ASV são (0,0) e a orientação inicial é de 0° . As cruzes vermelhas denotam os *waypoints*. O algoritmo considera que o veículo atinge o *waypoint* quando a distância é inferior a

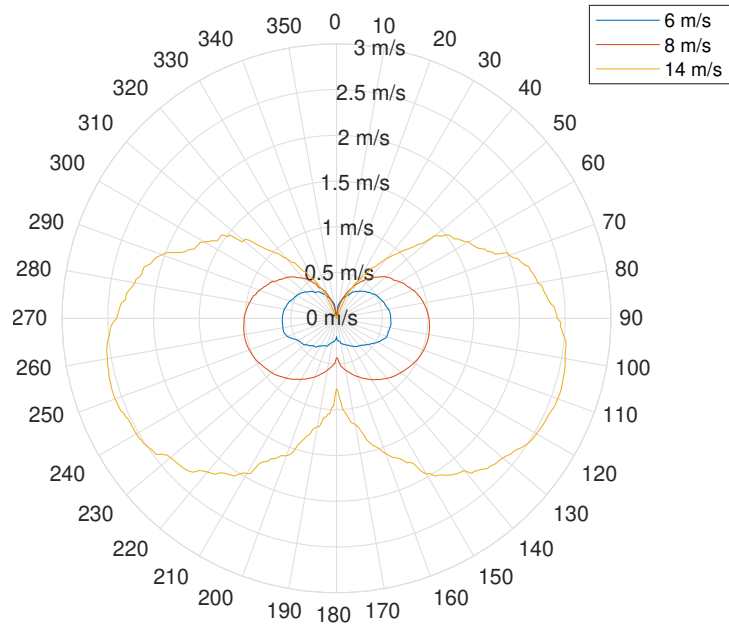


Figura 6.6: Diagrama polar da estação marítima modelada

10 m, correspondendo às áreas definidas pelas circunferências pretas. O tracejado azul descreve a trajetória.

O segundo teste, apresentado na Figura 6.8, avalia o modo *turning* de navegação, ou seja, quando o ângulo entre a rota para *owaypoint* e o vento corresponde à área de não navegação do diagrama polar. A direção do vento é de -90° , sentido oeste-este, e a velocidade de 10 m/s. A posição inicial do ASV é (0,0) e a orientação inicial é 90° . A distância máxima a percorrer por bordo, representada pelo tracejado preto, foi de ± 30 m. A Figura 6.8 mostra que o ASV consegue atingir o *waypoint*, zigzagueando sem nunca ultrapassar a distância máxima por bordo.

Por fim, testou-se o comportamento do USV numa missão que combina os dois modos de navegação. O vento tem uma direção de 0° e velocidade 10 m/s. A posição e a orientação inicial da plataforma é (0,0) e 0° , respetivamente. A Figura 6.9 mostra o trajeto realizado. Para atingir o primeiro *waypoint*, o veículo utiliza o modo *turning* para navegar contra o vento; para alcançar o segundo *waypoint*, como o vento é favorável, o USV navega no modo normal.

6.5 Station Keeping

O teste de validação do algoritmo de *station keeping* foi realizado com vento a 0° , sentido norte-sul, e velocidade 10 m/s. A posição e a orientação inicial da plataforma é (0,0) e 0° , respetivamente. A Figura 6.10 apresenta o resultado. A cruz laranja identifica o *waypoint* de *station keeping* e as cruzes vermelhas os *waypoints* gerados pelo algoritmo. A área circular de *station keeping* definida possui um $r = 50$ m. O

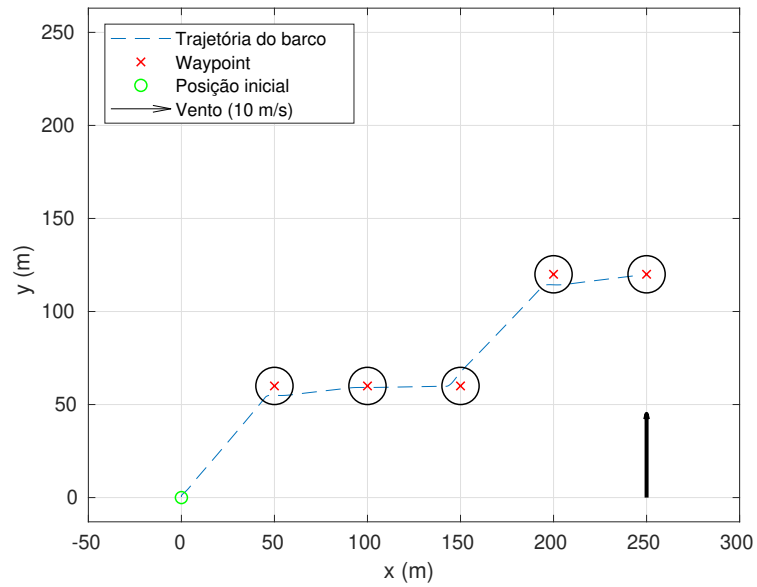


Figura 6.7: Seguimento de *waypoints* no modo normal de navegação

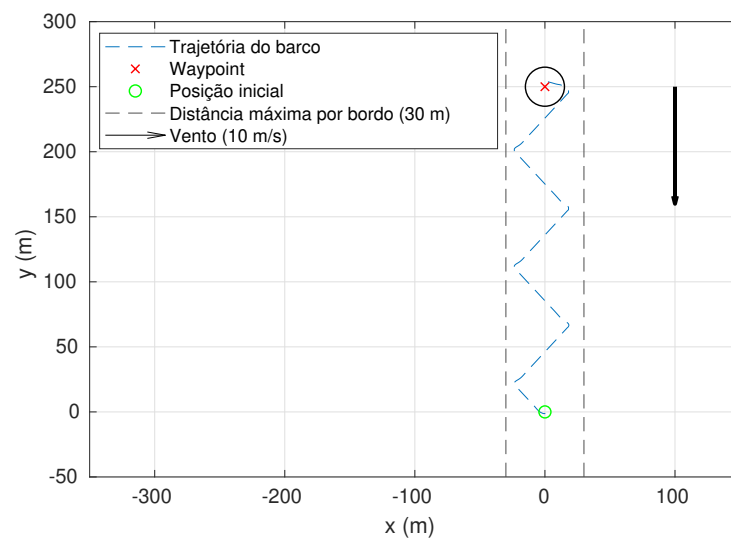


Figura 6.8: Seguimento de *waypoints* no modo *turning* de navegação

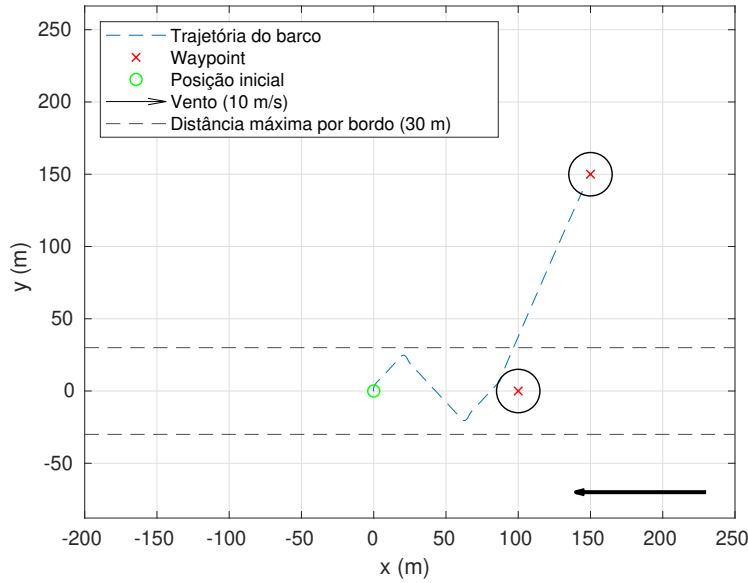


Figura 6.9: Seguimento de *waypoints* utilizando os dois modos de navegação

o algoritmo considera que o veículo atinge o *waypoint* quando a distância for inferior a 10 m, correspondendo às áreas contidas pelas circunferências pretas.

6.6 Sumário

Este capítulo apresenta os resultados dos testes realizados para validar o modelo e os algoritmos de controlo e navegação implementados.

Primeiro, testou-se se, na ausência de controlo, as velas se alinham ao vento. A Figura 6.2 demonstra que, independentemente da velocidade do vento, as velas se alinham ao vento ($\alpha = 0$) ao final de 17,5 s.

De seguida, configurou-se e analisou-se o comportamento do controlo do ângulo de ataque da vela asa em relação ao vento através da cauda. De acordo com as características do perfil alar Eppler E473 selecionado, o ângulo de ataque desejado é $\alpha_d = \pm 10^\circ$. A Figura 6.3 mostra que, como era esperado, o comportamento do sistema corresponde ao apresentado na Figura 3.15. O controlador PID da cauda foi testado experimentalmente com diferentes ganhos e velocidades do vento. Os resultados da Figura 6.4, obtidos com diferentes ganhos, permitiram selecionar os seguintes ganhos: $(K_p; K_i; K_d) = (0,0003; 0,0000; 0,0001)^1$. A Figura 6.5 agrega os resultados obtidos com os ganhos selecionados e diferentes velocidades de vento.

Procedeu-se à construção do diagrama polar da estação marítima autónoma aplicando-se ventos variáveis em direção (0° a 360°) e intensidade (6 m/s, 8 m/s

¹Pelos testes realizados concluiu-se que o ganho integral (K_i) não tinha grande relevância nos resultados finais optando por não se usar, passando a ser um controlador PD.

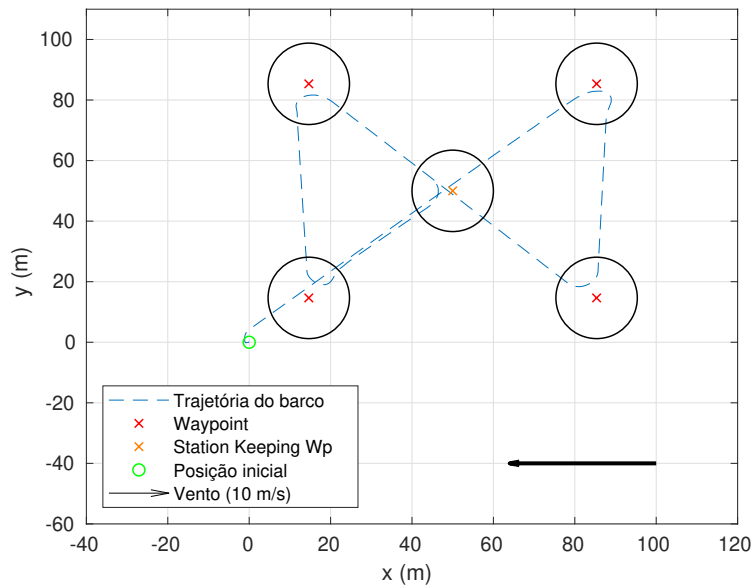


Figura 6.10: Manobra de *station keeping* através do seguimento de *waypoints*

e 14 m/s). O objetivo é caracterizar o comportamento do barco, dentro da sua gama de operação, face às condições do vento. As velocidades do vento escolhidas correspondem às velocidade mínima, intermédia e máxima da plataforma experimentalmente determinadas. A Figura 6.6 apresenta os resultados. A plataforma, que consegue navegar com ângulos com o vento entre $\pm 35^\circ$ a $\pm 170^\circ$, é sempre mais veloz ao largo. A velocidade intermédia de deslocamento é 1 m/s, conseguindo atingir um velocidade máxima de 2,6 m/s.

Para testar o controlo de direção em conjunto com os dois modos de navegação foram definidas as três missões representadas nas Figuras 6.7, 6.8 e 6.9. Na primeira, o USV percorre cinco *waypoints* navegando ao largo no modo normal; na segunda, navega à bolina no modo *turning* em direção a um *waypoint*; e, na terceira, navega nos dois modos de navegação para alcançar todos os *waypoints* definidos.

Por fim, foi testado o algoritmo de *station keeping*, um dos principais objetivos do trabalho. Dado um *waypoint* para *station keeping* o veículo calculou quatro *waypoints* equidistantes, como explicado na Secção 5.4, a uma distância de 50 m. Os resultados obtidos são apresentados na Figura 6.10, onde se verifica que a estação marítima consegue executar o algoritmo de *station keeping*.

Capítulo 7

Conclusão

7.1 Discussão de Resultados

Esta dissertação descreve a modelação, controlo e simulação de uma estação marítima autónoma. A plataforma concebida, que consiste num veleiro autónomo, apresenta um casco de 4,36 m de LOA e está equipado com um conjunto vela asa rígida trapezoidal e cauda. A vela principal, com uma área de $\approx 2 \text{ m}^2$, e a cauda, com $\approx 0,3 \text{ m}^2$, possuem um perfil alar simétrico Eppler E473. O veículo, com cerca de 516 kg de massa, suporta um *payload* de 88 kg.

Todos os componentes físicos da estação marítima autónoma foram modelados no SolidWorks. A simulação da plataforma foi efetuada através das ferramentas ROS e Gazebo. Foram utilizados *plugins* do Gazebo, que simulam as forças da hidrostática (reutilizado) e da hidro/aerodinâmica (alterado), e desenvolvidos dois novos *plugins* responsáveis pelo controlo do sistema e por atuar nas juntas do leme e da cauda do modelo. O sistema de controlo implementa dois controladores: (i) um controlador PD do ângulo de ataque da vela asa em relação ao vento, via cauda; e (ii) um controlador PD da direção da plataforma, via leme. A simulação e controlo foram testados exaustivamente e os resultados validam o correto comportamento do modelo. A plataforma consegue operar com velocidades do vento entre os 6 m/s e 14 m/s, conseguindo atingir uma velocidade máxima de $\approx 2,6 \text{ m/s}$. A velocidade intermédia de navegação do veleiro é de $\approx 1 \text{ m/s}$. A plataforma, em termos de navegação, consegue realizar missões definidas através de listas de *waypoints*, incluindo missões de *station keeping*, essenciais para operar como estação marítima autónoma.

7.2 Dificuldades

O desenvolvimento da solução debateu-se com problemas de escolha e integração dos componentes do ambiente de desenvolvimento e de desenvolvimento da própria simulação.

Na composição do ambiente de desenvolvimento surgiram dificuldades de: (i) compatibilidade entre versões do sistema operativo e das ferramentas ROS e Gazebo; (ii) compatibilidade entre o simulador Gazebo e as bibliotecas de simulação da dinâmica de fluidos; e (iii) documentação e atualização das bibliotecas de simulação.

A correta simulação do comportamento físico da plataforma colocou grandes desafios de: (i) parametrização hidrodinâmica da plataforma; e (ii) alteração dos *plugins* responsáveis pela simulação da interação plataforma-ambiente para passarem a simular corretamente o comportamento de um veleiro.

7.3 Trabalho Futuro

Em primeiro lugar, pretende-se simular o comportamento da estação com perturbações, sejam elas resultantes de correntes, ondas ou variações do vento, obtendo resultados mais próximos da realidade. A simulação das perturbações produzidas por correntes marítimas e ondas implica melhorar a parametrização hidrodinâmica da plataforma através de ferramentas de *Computational Fluid Dynamics* (CFD).

Em termos de navegação, pretende-se adicionar um novo modo que permita proteger o ASV quando confrontado com condições atmosféricas adversas. O novo modo definiria dinamicamente o ângulo de ataque desejado da vela, explorando a relação entre as forças de sustentação e arrasto para diferentes ângulos de ataque do perfil alar, incluindo, em casos extremos, o alinhamento da vela ao vento para proteger a estação.

Por fim, pretende-se construir e testar a plataforma em ambiente real. A plataforma, depois de devidamente equipada, poderá ser colocada no mar por períodos variáveis e em diferentes locais. Na qualidade de estação marítima, permitirá realizar diferentes tipos de missões, nomeadamente, missões dedicadas à recolha e transmissão de dados, à retransmissão de comunicações subaquáticas ou ao reenca-minhamento de comunicações de outras plataformas para terra.

7.4 Sumário

O desenvolvimento deste projeto permitiu estudar a viabilidade da construção de uma estação marítima autónoma, indo ao encontro dos objetivos referidos no Capítulo 1. A plataforma foi integralmente modelada de raiz, recorrendo ao *software*

SolidWorks, e a sua análise foi feita através do simulador Gazebo. O desenvolvimento e comunicação entre módulos, incluindo *plugins*, assim como a interface com o simulador foi assegurado pelo ROS. A simulação do comportamento físico da plataforma foi assegurado através da reutilização, alteração e desenvolvimento de *plugins* para o Gazebo. Considerando os testes e resultados obtidos, a estação marítima autônoma apresenta um bom desempenho, tendo sido capaz de executar com sucesso a manobra de *station keeping* (um dos principais requisitos). Alguns dos requisitos enunciados na Secção 4.3 não foram validados, mas foram considerados. É o caso da autossuficiência energética, que não pode ser verificada através deste tipo de simulação. No entanto, o modelo concebido garante que existe espaço e condições para suportar um sistema de painéis fotovoltaicos.

O ambiente de desenvolvimento e simulação adotado neste projeto pode ser reutilizado em trabalhos futuros com este ou outros USV na exploração de, nomeadamente, novos algoritmos de navegação e outros perfis alares. Contudo, a componente responsável pela simulação da hidrodinâmica deve ser, como referido na Secção 7.3, melhorada.

Por último, a realização deste projeto foi um desafio simultaneamente exigente e gratificante, tanto a nível académico como pessoal. Contribuiu para um aumento e consolidação de conhecimentos sobre o comportamento físico dos veleiros, navegação à vela e sobre as ferramentas SolidWorks, Gazebo e ROS, que serão certamente úteis, quer na vida pessoal, quer na profissional.

Referências

- [1] United States Coast Guard, “Automatic identification system (AIS) overview.” Disponível em <https://www.navcen.uscg.gov/automatic-identification-system-overview>. (Último acesso em 20/10/2022). [Citado na página xv]
- [2] BeiDou Navigation Satellite System, “BeiDou navigation satellite system.” Disponível em <http://en.beidou.gov.cn/>. (Último acesso em 19/01/2023). [Citado na página xv]
- [3] D. Crane, *Dictionary of Aeronautical Terms*. Aviation Supplies & Academics, Inc, 7th ed., 2020. [Citado nas páginas xv e xvi]
- [4] C. A. R. Hoare, “Communicating sequential processes,” *Communications of the ACM*, vol. 21, no. 8, pp. 666–677, 1978. [Citado na página xv]
- [5] H. K. Khalil, “Nonlinear systems: third edition,” *Prentice Hall*, vol. 115, 2002. [Citado na página xv]
- [6] R. Singh, “Chapter 13 - model-based control system design and evaluation for continuous tablet manufacturing processes (via direct compaction, via roller compaction, via wet granulation),” in *Process Systems Engineering for Pharmaceutical Manufacturing* (R. Singh and Z. Yuan, eds.), vol. 41 of *Computer Aided Chemical Engineering*, pp. 317–351, Elsevier, 2018. [Citado na página xv]
- [7] B. Fallahi, L. Cheng, and M. Tavakoli, “Chapter two - state observation and feedback control in robotic systems for therapy and surgery,” in *Control Systems Design of Bio-Robotics and Bio-mechatronics with Advanced Applications* (A. T. Azar, ed.), pp. 33–73, Academic Press, 2020. [Citado na página xv]
- [8] M. F. Silva, B. Malheiro, P. Guedes, and P. Ferreira, “Airfoil selection and wingsail design for an autonomous sailboat,” in *Robot 2019: Fourth Iberian Robotics Conference* (M. F. Silva, J. Luís Lima, L. P. Reis, A. Sanfeliu, and D. Tardioli, eds.), (Cham), pp. 305–316, Springer International Publishing, 2020. [Citado nas páginas viii, xvi, 34, 38, 39, 40, 42, 47 e 48]
- [9] Iridium Communications, “Iridium communications.” Disponível em <https://www.iridium.com/>. (Last accessed in 19/01/2023). [Citado na página xvi]

-
- [10] The Microtransat Challenge, “About the microtransat.” Disponível em <https://www.microtransat.org/>. (Último acesso em 19/01/2023). [Citado na página xvi]
- [11] C. Alves, P. Aguiar, and L. Sebastião, “Sailbot autonomous marine robot of eolic propulsion,” Master’s thesis, Instituto Superior Técnico, 2010. [Citado nas páginas viii, xvi, 31 e 38]
- [12] A. Pereira, J. Das, and G. S. Sukhatme, “An experimental study of station keeping on an underactuated asv,” in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3164–3171, Sep. 2008. [Citado nas páginas vii, xvi, 17, 18 e 91]
- [13] J. Melin, “Modeling, control and state-estimation for an autonomous sailboat,” Master’s thesis, Uppsala University, Suécia, 2015. [Citado nas páginas vii, xvii, 30 e 32]
- [14] J. E. Manley, “Unmanned surface vehicles, 15 years of development,” in *OCEANS 2008*, pp. 1–4, 2008. [Citado na página 3]
- [15] SailboatData, “Prindle 19.” Disponível em <https://sailboatdata.com/sailboat/prindle-19?units=metric>. (Último acesso em 21/10/2022). [Citado na página 3]
- [16] G. Elkaim and R. Kelbley, “Station keeping and segmented trajectory control of a wind-propelled autonomous catamaran,” in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 2424–2429, Dec 2006. [Citado nas páginas vii, 3 e 4]
- [17] G. H. Elkaim, *System identification for precision control of a wingsailed GPS-guided catamaran*. PhD thesis, Stanford University, California, 2001. [Citado nas páginas ix, 3, 47, 48 e 90]
- [18] J. C. Alves and N. A. Cruz, “Fast - an autonomous sailing platform for oceanographic missions,” in *OCEANS 2008*, pp. 1–7, 2008. [Citado na página 4]
- [19] J. C. Alves and N. A. Cruz, “A mission programming system for an autonomous sailboat,” in *2014 Oceans - St. John’s*, pp. 1–7, Sep. 2014. [Citado nas páginas vii, 5 e 6]
- [20] Offshore Sensing AS - SailBuoy, “Sailbuoy - unmanned surface vessel.” Disponível em <http://www.sailbuoy.no/>. (Último acesso em 31/10/2022). [Citado nas páginas vii e 6]

-
- [21] Offshore Sensing AS - SailBuoy, “Products - datasheet.” Disponível em <http://www.sailbuoy.no/products>. (Último acesso em 27/10/2022). [Citado na página 6]
- [22] M. F. Silva, A. Friebe, B. Malheiro, P. Guedes, P. Ferreira, and M. Waller, “Rigid wing sailboats: A state of the art survey,” *Ocean Engineering*, vol. 187, p. 106150, 2019. [Citado nas páginas 6, 38 e 44]
- [23] C. Meinig, N. Lawrence-Slavas, R. Jenkins, and H. M. Tabisola, “The use of saildrones to examine spring conditions in the bering sea: Vehicle specification and mission performance,” in *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–6, 2015. [Citado nas páginas 6 e 7]
- [24] Saildrone, “Saildrone explorer product card.” Disponível em <https://www.saildrone.com/technology/vehicles>. (Último acesso em 27/10/2022). [Citado nas páginas vii e 7]
- [25] Saildrone, “Antarctic circumnavigation.” Disponível em <https://www.saildrone.com/missions/antarctica>. (Último acesso em 27/10/2022). [Citado nas páginas 7 e 8]
- [26] Saildrone, “Arctic survey of ice, co2 & fisheries.” Disponível em <https://www.saildrone.com/missions/us-arctic-survey>. (Último acesso em 27/10/2022). [Citado na página 7]
- [27] E. D. Cokelet, C. Meinig, N. Lawrence-Slavas, P. J. Stabeno, C. W. Mordy, H. M. Tabisola, R. Jenkins, and J. N. Cross, “The use of saildrones to examine spring conditions in the bering sea,” in *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–7, 2015. [Citado na página 7]
- [28] J. N. Cross, C. W. Mordy, H. M. Tabisola, C. Meinig, E. D. Cokelet, and P. J. Stabeno, “Innovative technology development for arctic exploration,” in *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–8, 2015. [Citado nas páginas 7 e 8]
- [29] NOAA-PMEL, “Noaa’s 2021 saildrone observations of atlantic hurricanes for improvement of intensity forecasts.” Disponível em <https://www.pmel.noaa.gov/saildrone-hurricane2021/index.html>. (Último acesso em 27/10/2022). [Citado na página 8]
- [30] L. J. Frizzell-Makowski, R. A. Shelsby, J. Mann, and D. Scheidt, “An autonomous energy harvesting station-keeping vehicle for persistent ocean surveillance,” in *OCEANS’11 MTS/IEEE KONA*, pp. 1–4, Sep. 2011. [Citado nas páginas vii, 8 e 9]

- [31] Z. Pan, Y. Cao, D. Yang, Q. Yu, and Y. Qiu, “Architecture and design of unmanned sailboat control system with open source, accessibility and reliability,” in *2021 IEEE International Conference on Unmanned Systems (ICUS)*, pp. 520–526, Oct 2021. [Citado nas páginas vii, 9, 10 e 11]
- [32] H. Oh, S. Kim, A. Tsourdos, and A. Savvaris, “Comparison of station keeping strategies for long endurance autonomous surface vehicle,” *Journal of Marine Science and Technology*, vol. 25, no. 1, pp. 13–25, 2020. [Citado nas páginas xi, 11 e 12]
- [33] J. P. Storey, J. L. Hammond, J. E. G-H-Cater, B. W. Metcalfe, and P. R. Wilson, “Modelling dynamic photovoltaic arrays for marine applications,” in *2016 IEEE 17th Workshop on Control and Modeling for Power Electronics (COMPEL)*, pp. 1–8, 2016. [Citado nas páginas vii e 11]
- [34] European Commission (Horizon H2020 Programme), “Road-, air-, and water-based future internet experimentation.” Disponível em <http://www.rawfie.eu/>. (Último acesso em 20/10/2022). [Citado na página 13]
- [35] B. Ferreira, A. Coelho, M. Lopes, A. Matos, C. Gonçalves, S. Kandasamy, R. Campos, and J. Barbosa, “Flexible unmanned surface vehicles enabling future internet experimentally-driven research,” in *OCEANS 2017 - Aberdeen*, pp. 1–6, June 2017. [Citado nas páginas vii e 13]
- [36] National Oceanic and Atmospheric Administration. Disponível em <https://www.noaa.gov/>. (Último acesso em 25/10/2022). [Citado nas páginas 14 e 22]
- [37] J. R. Higinbotham, P. G. Hitchener, and J. R. Moisan, “Development of a new long duration solar powered autonomous surface vehicle,” in *OCEANS 2006*, pp. 1–6, Sep. 2006. [Citado nas páginas vii e 14]
- [38] J. R. Higinbotham, J. R. Moisan, C. Schirtzinger, M. Linkswiler, J. Yungel, and P. Orton, “Update on the development and testing of a new long duration solar powered autonomous surface vehicle,” in *OCEANS 2008*, pp. 1–10, Sep. 2008. [Citado nas páginas vii, 15 e 24]
- [39] H. Ferreira, C. Almeida, H. Silva, J. Almeida, E. Silva, and A. Martins, “ROAZ and ROAZ II autonomous surface vehicle design and implementation,” in *International Lifesaving Congress*, pp. 1–6, 2007. [Citado nas páginas 15 e 16]
- [40] H. Ferreira, A. Martins, A. Dias, C. Almeida, J. M. Almeida, and E. Silva, “ROAZ autonomous surface vehicle design and implementation,” *Robótica Controlo, Automação, Instrumentação*, pp. 22–27, 2007. [Citado nas páginas vii e 16]

-
- [41] A. Matos, E. Silva, J. Almeida, A. Martins, H. Ferreira, B. Ferreira, J. Alves, A. Dias, S. Fioravanti, D. Bertin, and V. Lobo, “Unmanned maritime systems for search and rescue,” in *Search and Rescue Robotics*, ch. 5, pp. 77–92, Rijeka: IntechOpen, 2017. [Citado na página 16]
- [42] M. M. Marques, A. Martins, A. Matos, N. Cruz, J. M. Almeida, J. C. Alves, V. Lobo, and E. Silva, “Rex 2014 - robotic exercises 2014 multi-robot field trials,” in *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–6, 2015. [Citado nas páginas vii e 17]
- [43] H. Ferreira, C. Almeida, A. Martins, J. Almeida, A. Dias, G. Silva, and E. Silva, “Environmental modeling with precision navigation using ROAZ autonomous surface vehicle,” in *IROS 2012-IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1–6, 2012. [Citado na página 16]
- [44] Marine Advanced Robotics, Inc, “Marine advanced robotics, inc.” Disponível em <https://wam-v.com/>, 2004. (Último acesso em 29/09/2022). [Citado na página 18]
- [45] E. I. Sarda, I. R. Bertaska, A. Qu, and K. D. von Ellenrieder, “Development of a usv station-keeping controller,” in *OCEANS 2015 - Genova*, pp. 1–10, May 2015. [Citado nas páginas vii, 18 e 19]
- [46] E. I. Sarda, H. Qu, I. R. Bertaska, and K. D. von Ellenrieder, “Station-keeping control of an unmanned surface vehicle exposed to current and wind disturbances,” *Ocean Engineering*, vol. 127, pp. 305–324, 2016. [Citado nas páginas 18 e 19]
- [47] H. Qu and K. D. von Ellenrieder, “Adaptive wind feedforward control of an unmanned surface vehicle for station keeping,” in *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–10, Oct 2015. [Citado nas páginas xi, 19 e 24]
- [48] E. I. Sarda and M. R. Dhanak, “Launch and recovery of an autonomous underwater vehicle from a station-keeping unmanned surface vehicle,” *IEEE Journal of Oceanic Engineering*, vol. 44, pp. 290–299, April 2019. [Citado na página 19]
- [49] Woods Hole Oceanographic Institution, “Remus 100.” Disponível em <https://www2.whoi.edu/site/osl/vehicles/remus-100/>. (Último acesso em 29/09/2022). [Citado na página 19]
- [50] M. Dunbabin, A. Grinham, and J. Udy, “An autonomous surface vehicle for water quality monitoring,” in *Australasian conference on robotics and automation (ACRA)*, pp. 2–4, 2009. [Citado nas páginas vii, 20 e 24]

-
- [51] Liquid Robotics, “The wave glider.” Disponível em <https://www.liquid-robotics.com/wave-glider/overview/>. (Último acesso em 25/10/2022). [Citado nas páginas vii e 21]
- [52] R. Hine, S. Willcox, G. Hine, and T. Richardson, “The wave glider: A wave-powered autonomous marine vehicle,” in *OCEANS 2009*, pp. 1–6, 2009. [Citado nas páginas 21, 22 e 23]
- [53] Liquid Robotics, “The wave glider spec sheet.” Disponível em <https://www.info.liquid-robotics.com/wave-glider-spec-sheet>. (Último acesso em 05/11/2022). [Citado na página 21]
- [54] Pacific Marine Environmental Laboratory. Disponível em <https://www.pmel.noaa.gov/>. (Último acesso em 25/10/2022). [Citado na página 22]
- [55] S. Willcox, C. Meinig, C. L. Sabine, N. Lawrence-Slavas, T. Richardson, R. Hine, and J. Manley, “An autonomous mobile platform for underway surface carbon measurements in open-ocean and coastal waters,” in *OCEANS 2009*, pp. 1–8, 2009. [Citado na página 22]
- [56] National Data Buoy Center. Disponível em <https://www.ndbc.noaa.gov/>. (Último acesso em 25/10/2022). [Citado na página 22]
- [57] J. E. Manley, A. Leonardi, and C. Beaverson, “Research to operations: Evaluating unmanned surface vehicles,” in *OCEANS 2016 MTS/IEEE Monterey*, pp. 1–5, Sep. 2016. [Citado nas páginas vii e 22]
- [58] National Marine Fisheries Service, “Northwest fisheries science center.” Disponível em <https://www.fisheries.noaa.gov/about/northwest-fisheries-science-center>. (Último acesso em 25/10/2022). [Citado na página 22]
- [59] H. Joe, H. Roh, J. Pyo, J. Gu, J. Kim, B.-J. Kim, and S.-C. Yu, “Development of energy-harvesting unit and propulsion unit for a robotic buoy system,” in *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–6, 2015. [Citado nas páginas vii e 23]
- [60] N. D. Kraus, “Wave glider dynamic modeling, parameter identification and simulation,” Master’s thesis, University of Hawaii at Manoa, Honolulu, 2012. [Citado na página 23]
- [61] T. J. Enqvist, “Wingsail arrangements for robotic sailboat,” Master’s thesis, Åland University of Applied Science, 2016. [Citado nas páginas viii, 28, 39, 40, 41, 47 e 48]

- [62] Kavas Yachting, “Beginner nautical terms.” Disponível em <https://www.kavas.com/sailors-guide/chapter-a-the-basics/a3-beginner-nautical-terms.html>. (Último acesso em 15/11/2022). [Citado nas páginas vii e 30]
- [63] T. Fossen, *Guidance and Control of Ocean Vehicles*. Wiley, 1994. [Citado na página 30]
- [64] C. Tretow, “Design of a free-rotating wing sail for an autonomous sailboat,” Master’s thesis, KTH Royal Institute of Technology School of Engineering Sciences, Suécia, 2017. [Citado nas páginas 30, 34, 47 e 48]
- [65] M. Dhanak and N. Xiros, *Springer Handbook of Ocean Engineering*. Springer Handbooks, Springer International Publishing, 2016. [Citado na página 31]
- [66] T. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011. [Citado nas páginas viii, 31, 36 e 37]
- [67] J. Kimball, *Physics of Sailing*. CRC Press, 2009. [Citado nas páginas vii, viii, 31, 32, 33, 34, 35, 36 e 38]
- [68] H. Heisler, “14 - vehicle body aerodynamics,” in *Advanced Vehicle Technology (Second Edition)* (H. Heisler, ed.), pp. 584–634, Oxford: Butterworth-Heinemann, second edition ed., 2002. [Citado na página 32]
- [69] P. F. Rynne and K. D. von Ellenrieder, “Development and preliminary experimental validation of a wind- and solar-powered autonomous surface vehicle,” *IEEE Journal of Oceanic Engineering*, vol. 35, pp. 971–983, 10 2010. [Citado nas páginas 33 e 38]
- [70] D. Marghitu, *Mechanical Engineers Handbook*. Academic Press series in engineering, Academic Press, 2001. [Citado na página 33]
- [71] G. Hugh Elkaim and C. Lee Boyce, “Experimental aerodynamic performance of a self-trimming wing-sail for autonomous surface vehicles,” *IFAC Proceedings Volumes*, vol. 40, no. 17, pp. 271–276, 2007. 7th IFAC Conference on Control Applications in Marine Systems. [Citado nas páginas 34 e 36]
- [72] K. L. Wille, “Autonomous sailboats modeling, simulation, control,” Master’s thesis, Norwegian University of Science and Technology, Noruega, 2016. [Citado nas páginas viii, ix, 35 e 90]
- [73] L. Xiao and J. Jouffroy, “Modeling and nonlinear heading control of sailing yachts,” *IEEE Journal of Oceanic Engineering*, vol. 39, pp. 256–268, 4 2014. [Citado na página 35]

- [74] Airfoil Tools, “Goe 63 airfoil (goe63-il).” Disponível em <http://airfoiltools.com/airfoil/details?airfoil=goe63-il>. (Último acesso em 08/01/2023). [Citado na página 39]
- [75] Airfoil Tools, “S1048 14% (danny howell) (s1048-il).” Disponível em <http://airfoiltools.com/airfoil/details?airfoil=s1048-il>. (Último acesso em 08/01/2023). [Citado na página 39]
- [76] G. H. Elkaim, “Autonomous surface vehicle free-rotating wingsail section design and configuration analysis,” *Journal of Aircraft*, vol. 45, pp. 1835–1852, 2008. [Citado nas páginas viii e 41]
- [77] P. H. Miller, M. Hamlet, and J. Rossman, “Continuous improvements to USNA sailbots for inshore racing and offshore voyaging,” in *Robotic Sailing 2012* (C. Sauzé and J. Finnis, eds.), (Berlin, Heidelberg), pp. 49–60, Springer Berlin Heidelberg, 2013. [Citado na página 42]
- [78] S. Yang, C. Liu, Y. Liu, J. An, and X. Xiang, “Generic and flexible unmanned sailboat for innovative education and world robotic sailing championship,” *Frontiers in Robotics and AI*, vol. 8, 3 2021. [Citado na página 43]
- [79] L. A. Levin, B. J. Bett, A. R. Gates, P. Heimbach, B. M. Howe, F. Janssen, A. McCurdy, H. A. Ruhl, P. Snelgrove, K. I. Stocks, D. Bailey, S. Baumann-Pickering, C. Beaverson, M. C. Benfield, D. J. Booth, M. Carreiro-Silva, A. Colação, M. C. Eblé, A. M. Fowler, K. M. Gjerde, D. O. B. Jones, K. Katsumata, D. Kelley, N. Le Bris, A. P. Leonardi, F. Lejzerowicz, P. I. Macreadie, D. McLean, F. Meitz, T. Morato, A. Netburn, J. Pawlowski, C. R. Smith, S. Sun, H. Uchida, M. F. Vardaro, R. Venkatesan, and R. A. Weller, “Global observing needs in the deep ocean,” *Frontiers in Marine Science*, vol. 6, 2019. [Citado na página 43]
- [80] S. Aracri, F. Giorgio-Serchi, G. Suaria, M. E. Sayed, M. P. Nemitz, S. Mahon, and A. A. Stokes, “Soft robots for ocean exploration and offshore operations: A perspective,” *Soft Robotics*, vol. 8, pp. 625–639, 12 2021. [Citado na página 43]
- [81] C. Wong, E. Yang, X.-T. Yan, and D. Gu, “An overview of robotics and autonomous systems for harsh environments,” in *2017 23rd International Conference on Automation and Computing (ICAC)*, pp. 1–6, IEEE, 9 2017. [Citado na página 43]
- [82] L. Zhang, D. peng Jiang, J. xin Zhao, and S. Ma, “An auv for ocean exploring and its motion control system architecture,” *The Open Mechanical Engineering Journal*, vol. 7, pp. 40–47, 10 2013. [Citado na página 43]

- [83] J. Heidemann, W. Ye, J. Wills, A. Syed, and Y. Li, “Research challenges and applications for underwater sensor networking,” in *IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006.*, pp. 228–235, IEEE, 2006. [Citado na página 43]
- [84] H. Kaushal and G. Kaddoum, “Underwater optical wireless communication,” *IEEE Access*, vol. 4, pp. 1518–1547, 2016. [Citado na página 43]
- [85] H. Ferreira, R. Martins, E. Marques, J. Pinto, A. Martins, J. Almeida, J. Sousa, and E. P. Silva, “Swordfish: an autonomous surface vehicle for network centric operations,” in *OCEANS 2007 - Europe*, pp. 1–6, IEEE, 2007. [Citado na página 43]
- [86] Instituto Politécnico do Porto, “Laboratório de Sistemas Autónomos.” Disponível em <https://www.ipp.pt/investigacao/centros-de-investigacao/engenharia/crob>. (Último acesso em 05/01/2023). [Citado na página 44]
- [87] The International 2.4mR Class Association, “2.4mR.” Disponível em <https://www.inter24metre.org/about-the-24mr-keelboat-sailing>. (Último acesso em 06/01/2023). [Citado nas páginas 45 e 46]
- [88] Kama Yachts, “Proton 2.4mR - technical data.” Disponível em <https://kamayachts.com/en-proton.html>. (Último acesso em 06/01/2023). [Citado na página 45]
- [89] A. C. B. Miranda, “Modelação, simulação e controlo de um veleiro com vela asa,” Master’s thesis, Instituto Superior de Engenharia do Porto, Porto, 2021. [Citado nas páginas 47, 48 e 89]
- [90] Airfoil Tools, “Epppler 473 airfoil.” Disponível em <http://airfoiltools.com/airfoil/details?airfoil=e473-il>. (Último acesso em 08/01/2023). [Citado nas páginas viii, 47 e 48]
- [91] S. Jollands, “Points of sailing.” Disponível em <https://www.safe-skipper.com/points-of-sail/>. (Último acesso em 15/11/2022). [Citado nas páginas ix e 90]
- [92] Open Robotics. Disponível em <https://www.openrobotics.org/>. (Último acesso em 19/07/2023). [Citado na página 93]
- [93] ROS - Robot Operating System. Disponível em <https://www.ros.org/>. (Último acesso em 18/07/2023). [Citado na página 93]
- [94] ROS - Robot Operating System. Disponível em <https://www.ros.org/blog/ecosystem/>. (Último acesso em 18/07/2023). [Citado na página 93]

-
- [95] ROS - Robot Operating System. Disponível em <https://docs.ros.org/en/foxy/Releases.html>. (Último acesso em 18/07/2023). [Citado na página 93]
- [96] ROS - Robot Operating System. Disponível em <http://wiki.ros.org/Distributions>. (Último acesso em 18/07/2023). [Citado na página 93]
- [97] ROS - Robot Operating System. Disponível em <https://docs.ros.org/en/foxy/Tutorials/Advanced/Simulators/Simulation-Main.html>. (Último acesso em 18/07/2023). [Citado na página 94]
- [98] Unreal Engine. Disponível em <https://www.unrealengine.com/en-US>. (Último acesso em 18/07/2023). [Citado na página 94]
- [99] Unity. Disponível em <https://unity.com/>. (Último acesso em 18/07/2023). [Citado na página 94]
- [100] Cyberbotics. Disponível em <https://cyberbotics.com/>. (Último acesso em 19/07/2023). [Citado na página 94]
- [101] Open Dynamics Engine. Disponível em <https://www.ode.org/>. (Último acesso em 19/07/2023). [Citado na página 94]
- [102] Gazebo. Disponível em <https://gazebo.org/home>. (Último acesso em 19/07/2023). [Citado na página 94]
- [103] Dynamic Animation and Robotics Toolkit. Disponível em <https://dartsim.github.io/>. (Último acesso em 19/07/2023). [Citado na página 94]
- [104] OGRE. Disponível em <https://www.ogre3d.org/>. (Último acesso em 19/07/2023). [Citado na página 94]
- [105] NVIDIA. Disponível em <https://developer.nvidia.com/rtr/ray-tracing/optix>. (Último acesso em 19/07/2023). [Citado na página 94]
- [106] code-iai. Disponível em <https://github.com/code-iai/ROSIntegration>. (Último acesso em 19/07/2023). [Citado na página 95]
- [107] Open Robotics. Disponível em <https://github.com/osrf/vrx>. (Último acesso em 20/07/2023). [Citado na página 96]
- [108] Rhys Mainwaring. Disponível em https://github.com/srmainwaring/asv_sim. (Último acesso em 20/07/2023). [Citado na página 96]
- [109] Rhys Mainwaring. Disponível em https://github.com/srmainwaring/asv_wave_sim. (Último acesso em 20/07/2023). [Citado na página 96]

-
- [110] Rhys Mainwaring. Disponível em <https://github.com/srmainwaring/rs750>. (Último acesso em 20/07/2023). [Citado na página 96]
- [111] Paravisi, M.; H. Santos, D.; Jorge, V.; Heck, G.; Gonçalves, L. M.; Amory. Disponível em https://github.com/disaster-robotics-proalertas/usv_sim_lsa. (Último acesso em 20/07/2023). [Citado na página 96]
- [112] Gazebo Classic. Disponível em <https://github.com/gazebosim/gazebo-classic>. (Último acesso em 09/08/2023). [Citado na página 97]
- [113] Gazebo, “Aerodynamics.” Disponível em <http://classic.gazebosim.org/tutorials?tut=aerodynamics&cat=physics>. (Último acesso em 09/08/2023). [Citado nas páginas ix e 99]

Anexo A

Mareações e Manobras

A.1 Mareações

A navegação à vela está sempre condicionada pela orientação e intensidade do vento e dependendo da trajetória pretendida é necessário alterar a configuração da vela. A Figura A.1 apresenta as diferentes tipos de mareações à vela [89]:

- bolina ou *close-hauled* - o barco navega o mais próximo possível da linha do vento, aproximadamente até 45° . As velas estão bem caçadas;
- bolina folgada ou *close-reach* - semelhante à bolina mas as velas estão ligeiramente mais folgadas, uma vez que o ângulo entre a trajetória do barco e a direção do vento é maior;
- largo ou *beam-reach* - o vento entra por uma das partes laterais do barco (estibordo ou bombordo). As velas estão mais folgadas do que na bolina;
- largo folgado ou *broad-reach* - o vento começa a entrar pela popa do barco, as velas são mais folgadas;
- popa ou *training run* - o barco navega a favor do vento. As velas estão mais folgadas do que no largo;
- popa arrasada ou *run* - quando o rumo do barco é paralelo à direção do vento. As velas estão completamente folgadas.

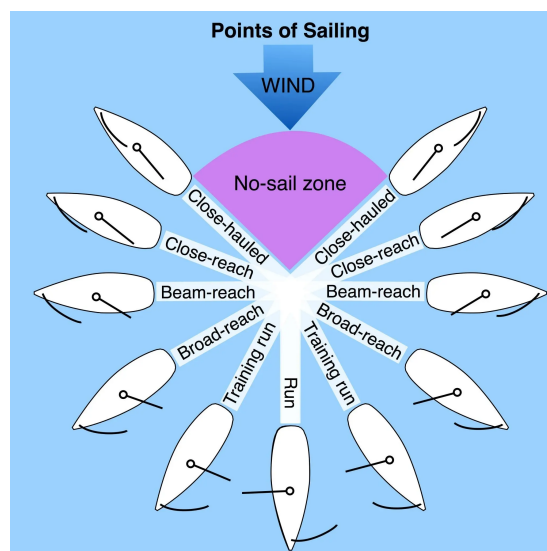


Figura A.1: Diferentes tipos de mareações [91]

A.2 Manobras

Um veleiro que navega contra o vento necessita fazer uma trajetória em ziguezague, passando o vento de um bordo para o outro. Esta manobra, exemplificada na Figura A.2, denomina-se de viragem por davante ou *tacking*.

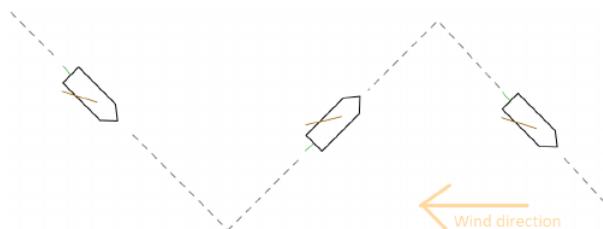


Figura A.2: Manobra viragem por davante [72]

Um veleiro que navega a favor do vento executa a manobra de viragem em roda ou *gybing* (Figura A.3) para virar de bordo.

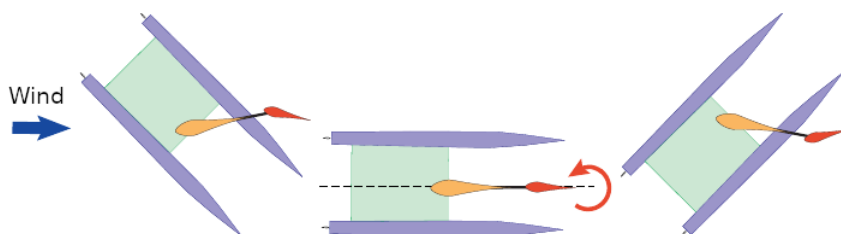


Figura A.3: Manobra viragem em roda [17]

Ambas as manobras de *tacking* e *gybing* exigem a viragem de bordo das velas quando o vento passa de estibordo para bombordo ou vice-versa.

Outra manobra relevante para o presente trabalho é a manutenção da posição relativa ou *Station keeping* entre a embarcação autónoma e outro objeto, local ou embarcação [12], minimizando o efeito de perturbações externas, como o vento, ondulação e correntes marítimas.

Anexo B

Ambiente de Desenvolvimento e Simulação

Este Anexo B tem como objetivo apresentar e analisar todos os recursos necessários para a realização da simulação: o ROS, o simulador, o ambiente de simulação escolhido e os *plugins* desenvolvidos e utilizados.

B.1 ROS

A *framework* ROS destina-se à comunicação com e desenvolvimento de aplicações para *robots*. O ROS, que é mantido pela Open Robotics [92], consiste num conjunto de bibliotecas de *software* e ferramentas *open-source* de suporte ao desenvolvimento de aplicações robóticas [93]. As suas principais vantagens são: (i) o sistema de comunicação padrão ou *middleware*, que gere a interface entre o *software* e *hardware* do *robot* através de processos (nós) que subscrevem e publicam mensagens [94]; e (ii) a vasta comunidade de utilizadores e de desenvolvimento, que contribui com inúmeras bibliotecas de mensagens, interface com sensores e navegação.

A Open Robotics vai lançando novas distribuições do ROS para manter o código compatível com a evolução do Linux. Atualmente existem três distribuições estáveis, duas de ROS2 [95] e uma de ROS1 [96], denominadas Iron, Humble e Noetic, respetivamente. Face ao ROS1, o ROS2 conserva o sistema de comunicação padrão e apresenta novas funcionalidades e diferenças ao nível da arquitetura.

B.2 Simuladores

O simulador utilizado deve preencher três requisitos:

- suportar a simulação aerodinâmica e hidrodinâmica;
- permitir visualizar o comportamento do *robot* durante a simulação;
- ser integrável com ROS.

Foram analisados quatro simuladores compatíveis com o ROS: o Webots e o Gazebo [97] seguidos do Unreal Engine [98] e do Unity [99]. Os dois primeiros são referidos na documentação oficial do ROS e os dois restantes estão mais associados à indústria de jogos.

B.2.1 Webots

O Webots é um simulador *open-source* para robótica desenvolvido pela Cyberbotics Ltd. [100]. Permite simular, modelar e programar *robots* em ambientes 3D e é utilizado nas áreas da indústria, investigação e educação. Este simulador, que é compatível com as linguagens de programação C/C++, Java, Python e MATLAB, pode ser instalado em sistemas operativos baseados em Linux, Windows e macOS. Utiliza o motor de física *Open Dynamics Engine* (ODE) [101], que é baseado em C/C++ Webots, para simular a dinâmica de e a colisão entre objetos. Para a renderização 3D utiliza o motor de renderização WRENJS, desenvolvido pela própria Cyberbotics e baseado em WebGL 2. O Webots é um dos simuladores sugeridos na documentação oficial do ROS para a simulação de *robots*.

B.2.2 Gazebo

O Gazebo é um conjunto de bibliotecas *open-source* em C++ que permite simular múltiplos *robots* com dinâmica complexa e interação entre objetos em ambientes 3D [102]. Atualmente, é desenvolvido pela Open Robotics e suportado por sistemas operativos baseados em Linux, Windows e macOS. As novas versões do Gazebo suportam os motores de física *Dynamic Animation and Robotics Toolkit* (DART) [103] e ODE, utilizando por omissão o DART, e o *Object-Oriented Graphics Rendering Engine* (OGRE) [104] ou o OptiX da NVIDIA [105] para a renderização 3D. O Gazebo, além de ser o simulador mais utilizado no LSA, apresenta uma maior comunidade de utilizadores e é recomendado na documentação oficial do ROS.

B.2.3 Unreal Engine

O Unreal Engine é uma ferramenta para criação de conteúdos 3D desenvolvida em C++ pela Epic Games. É utilizado principalmente para o desenvolvimento de jogos de computadores e consolas, sendo suportado por uma variedade de sistemas

operativos e plataformas como Linux, Windows, macOS e consolas de jogos. O Instituto de Inteligência Artificial da Universidade de Bremen desenvolveu um *plugin* [106] que permite a integração do Unreal Engine 4 com o ROS, apenas suportado em Linux e Windows. Este *plugin*, que implementa a estrutura básica de dados do ROS, permite a comunicação padrão através do método publicação/subscrição, não disponibilizando contudo a totalidade dos recursos do ROS. O Unreal Engine 4 utiliza para a simulação dinâmica de plataformas e de colisões o motor de física PhysX da NVIDIA. O motor de renderização é proprietário, mas recorre às bibliotecas DirectX11 e DirectX12 da Microsoft para a interface entre o *software* e *hardware*.

B.2.4 Unity

O Unity é uma ferramenta utilizada principalmente no desenvolvimento de jogos 3D. A linguagem de programação utilizada é C++ e C# e é suportado por múltiplas plataformas como consolas, telemóveis e computadores com sistemas operativos Linux, Windows e macOS. O motor de física utilizado é baseado no PhysX da NVIDIA e o motor de renderização é o OctaneRender. O Unity é integrável com o ROS através de uma biblioteca desenvolvida pela própria Unity.

B.2.5 Análise e Comparação

Apesar de todos os simuladores apresentados serem compatíveis com o ROS, apenas o Gazebo e o Webots asseguram uma integração total. Além disso, o Unreal Engine e o Unity destinam-se primordialmente ao desenvolvimento de jogos, apresentando uma curva de aprendizagem íngreme e uma comunidade e documentação de apoio focada no desenvolvimento de jogos. Entre o Gazebo e o Webots, optou-se pelo Gazebo porque satisfaz todos os requisitos, permitindo simular o comportamento dinâmico de *robots* em terra, no mar e ar, e é utilizado no LSA. Na Tabela B.1 são apresentadas as características principais dos quatro simuladores referidos.

Tabela B.1: Características dos diferentes simuladores

	Webots	Gazebo	Unreal Engine	Unity
Linguagem	C/C++ Python Java MATLAB	C++	C++	C++ C#
Sistema Operativo	Linux Windows macOS	Linux Windows macOS	Linux Windows	Linux Windows macOS
Motor de física	ODE	DART ODE	PhysX	PhysX
Motor de renderização	WRENJS	OGRE OptiX	Unreal	OctaneRender
Integração ROS	Sim	Sim	Sim	Sim

B.3 Ambiente de Simulação

Uma vez selecionado o simulador Gazebo, é necessário escolher um ambiente de simulação que incorpore o comportamento aerodinâmico e hidrodinâmica em cenários 3D. Estes ambientes de simulação consistem em bibliotecas de *plugins* desenvolvidas sobre o Gazebo.

B.3.1 Virtual RobotX

O *VirtualRobotX* (VRX) é um ambiente de simulação desenvolvido pela Open Robotics, mesma empresa que desenvolve o ROS, com o objetivo de simular plataformas do tipo USV em ambientes marítimos [107]. Este ambiente de simulação, que é compatível o ROS 2, o Gazebo Garden e a versão mais recente do Ubuntu, foi criado para a simulação do cenário da competição VRX, que envolve a realização de provas (e.g. *station keeping* e seguimento de *waypoints*) com o *robot* WAM-V. O VRX possui *plugins* de simulação da hidrostática (flutuação) e de perturbações ambientais (ondas) e documentação e tutoriais de apoio. Contudo, não simula as forças aerodinâmicas (*lift* e *drag*) geradas na vela e leme, obrigando a desenvolver um novo *plugin*.

B.3.2 ASV Simulator

O ASV Simulator é um ambiente de simulação de plataformas de navegação à superfície no Gazebo [108] que é compatível com as atuais versões do simulador e do ROS. Este ambiente de simulação reutiliza duas bibliotecas (*packages*): (i) o *asv_wave_sim* [109] de simulação e visualização de ondas; e (ii) o *rs750* [110] que modela o veículo de superfície utilizado. O *rs750* é um veículo que, à semelhança do *robot* que se pretende simular, utiliza uma vela e um leme para se deslocar. Assim, o ASV Simulator já inclui um *plugin* de simulação das forças aerodinâmicas (*lift* e *drag*). Ao contrário do VRX, apresenta pouca documentação de apoio.

B.3.3 USVSim

O USVSim é um ambiente de simulação de veículos de superfície (e.g. embarcações à vela e a motor) [111]. Este ambiente de simulação combina vários *packages* para testar os veículos em cenários sujeitos a perturbações ambientais (correntes marítimas, ondas e vento). O USVSim possui documentação de apoio e algumas demos de teste. No entanto, como foi desenvolvido para o ROS Kinetic, apenas é compatível com o Ubuntu 16.04, que está desatualizado.

B.3.4 Análise e Comparação

Analisando os três ambientes de simulação apresentados, o USVSim é o único que não tem compatibilidade com as versões de ROS e de Gazebo mais recentes. Isto pode dificultar o desenvolvimento da simulação, uma vez que o Ubuntu 16.04 foi lançado em 2016 e os computadores atuais ficam bastante limitados por falta de *drivers*. Entre o VRX e o ASV Simulator, este último parece ser a melhor opção, uma vez que tem toda a aerodinâmica e hidrodinâmica necessário para a simulação, inclusive um exemplo de simulação de um veleiro. No entanto, a documentação do ASV Simulator é insuficiente para ajudar a desenvolver um novo *plugin*. O VRX implica desenvolver um *plugin* que simule as forças *lift* e *drag* geradas na vela e no leme. Este *plugin* pode ser desenvolvido de raiz, usando para inspiração o *plugin* correspondente do ASV Simulator. Além disso, o VRX apresenta uma documentação bastante completa e detalhada, o que é uma vantagem. Pelas razões apresentadas o VRX foi o ambiente de simulação escolhido. Na Tabela B.2 é possível de visualizar as principais características dos três ambientes de simulação.

Tabela B.2: Características dos diferentes ambientes de simulação

	VRX	ASV Simulator	USVSim
Aerodinâmica	Não	Sim	Sim
Hidrodinâmica	Incompleto	Sim	Sim
Documentação	Sim	Não	Sim
Atualizado	Sim	Sim	Não

B.4 Plugins

De seguida, são apresentados os principais *plugins* utilizados na simulação: (i) LiftDragPlugin, (ii) flap_control, (iii) usv_control.

B.4.1 LiftDragPlugin

O LiftDragPlugin é um *plugin* do gazebo-classic [112] que permite simular as forças aerodinâmicas ou hidrodinâmicas exercidas num perfil alar. Este *plugin* foi originalmente desenvolvido para aeronaves, assumindo que a velocidade do vento real é sempre zero. Por esta razão, foi necessário alterar o *plugin* para contemplar não só a velocidade linear do próprio veículo, mas também a velocidade do fluido (ar ou água), passando-se a calcular as forças *lift* e *drag* geradas nos vários componentes (vela, cauda, leme e patilhão) com base nestas duas componentes. Adicionalmente, a interface do LiftDragPlugin com o ROS foi igualmente alterada para subscrever um tópico com a velocidade linear real do vento. Na Figura B.1 apresenta-se o diagrama de funcionamento do *plugin* com a vela.

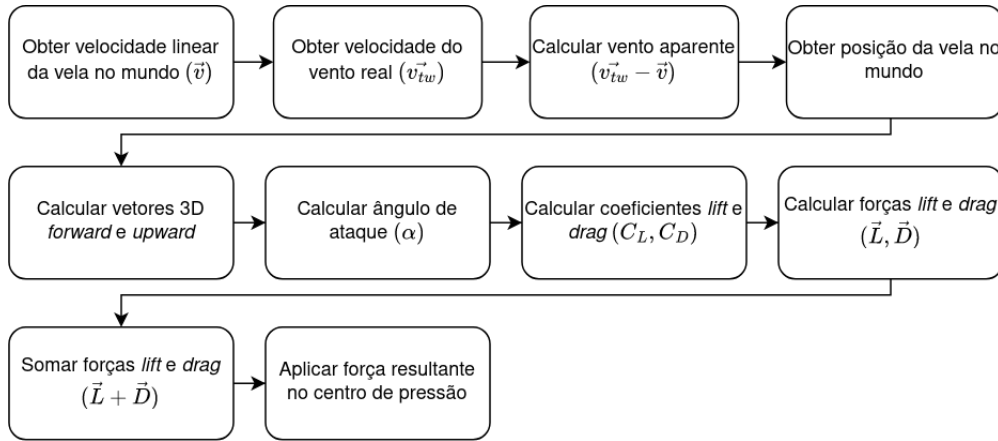


Figura B.1: Funcionamento do LiftDragPlugin para a vela

Com este *plugin*, o perfil alar e o fluído envolvente são definidos através dos seguintes parâmetros:

- `link_type`: tipo de componente (*sail*, *rudder* ou *keel*);
- `a0`: α inicial;
- `cla`: inclinação da curva de *lift* antes do ponto de *stall*;
- `cda`: inclinação da curva de *drag* antes do ponto de *stall*;
- `cma`: inclinação da curva de *momentum* antes do ponto de *stall*;
- `alpha_stall`: α onde inicia o ponto de *stall*;
- `cla_stall`: inclinação da curva de *lift* depois do ponto de *stall*;
- `cda_stall`: inclinação da curva de *drag* depois do ponto de *stall*;
- `cma_stall`: inclinação da curva de *momentum* depois do ponto de *stall*;
- `cp`: centro de pressão. As forças geradas vão ser aplicadas neste ponto;
- `area`: área do perfil alar;
- `fluid_density`: densidade do fluído que está em contacto com o perfil alar;
- `forward`: vetor 3D que representa a direção do movimento no eixo do *link*;
- `upward`: vetor 3D que representa a direção da força *lift* ou *drag*;
- `radial_symmetry`: *flag* caso o perfil alar seja simétrico;
- `link_name`: nome do *link* onde a força gerada será aplicada.

Os parâmetros do perfil alar são obtidos através da linearização das curvas $C_l(\alpha)$, $C_d(\alpha)$, $C_m(\alpha)$, obtidas experimentalmente, onde α , C_l , C_d e C_m representam o ângulo de ataque, o coeficiente de *lift*, o coeficiente de *drag* e o coeficiente do *pitching moment*, respetivamente. A Figura B.2 ilustra a linearização da curva $C_l(\alpha)$.

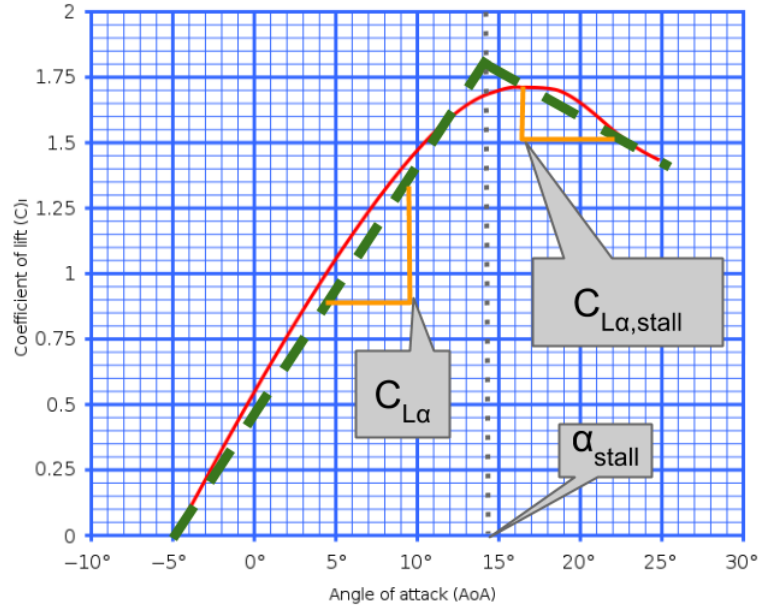


Figura B.2: Linearização da $C_l(\alpha)$ [113].

A Listagem B.1 apresenta a parametrização de um perfil alar para o LiftDragPlugin.

```

1 <robot xmlns:xacro="http://ros.org/wiki/xacro">
2   <gazebo>
3     <plugin name="sail_liftdrag" filename="libliftdrag_plugin.so">
4       <link_type>rudder</link_type>
5       <a0>0.0</a0>
6       <cla>4.3155</cla>
7       <cda>0.0624</cda>
8       <cma>0.0</cma>
9       <alpha_stall>0.2355</alpha_stall>
10      <cla_stall>-2.5267</cla_stall>
11      <cda_stall>1.067</cda_stall>
12      <cma_stall>0.0</cma_stall>
13      <cp>0.0 0.0 0.0</cp>
14      <area>2.04</area>
15      <fluid_density>1.2041</fluid_density>
16      <forward>1 0 0</forward>
17      <upward>0 1 0</upward>
18      <radial_symmetry>true</radial_symmetry>
19      <link_name>sailboat/sail_link</link_name>
20    </plugin>
21  </gazebo>
22 </robot>

```

Listagem B.1: Exemplo de utilização do *plugin* LiftDragPlugin

B.4.2 Plugin flap_control

O *plugin* `flap_control` foi desenvolvido para permitir atuar na cauda da vela asa e no leme do USV através de tópicos ROS. Este *plugin* consiste num nó ROS que: (i) subscreve o tópico da posição angular da junta cauda (`/sailboat/control_flap`) e o tópico da posição angular da junta do leme (`/sailboat/control_rudder`); (ii) determina a nova posição e orientação das juntas da vela asa, da cauda e do leme sempre que recebe uma nova posição angular; e (iii) publica a nova posição e orientação da vela asa, da cauda ou do leme nos tópicos `/sail_pose`, `/tail_pose`, `/rudder_pose`. Na Figura B.3 representa o funcionamento deste *plugin*.

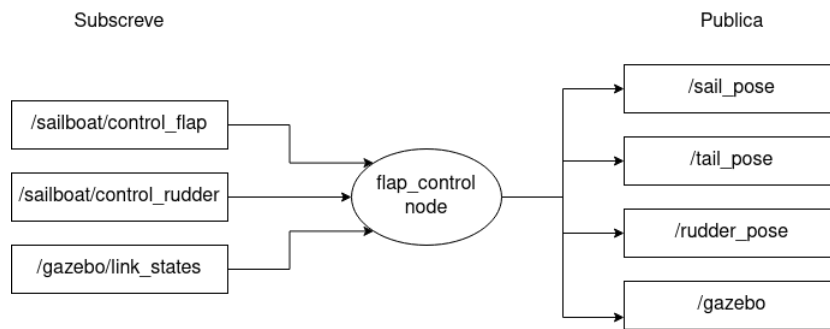


Figura B.3: Diagrama de comunicação do *plugin* `flap_control`

B.4.3 Plugin usv_control

O *plugin* `usv_control` é responsável pelo controlo do conjunto vela asa e cauda e da direção do leme. Consiste num nó ROS que: (i) subscreve o tópico do vento aparente (`/sailboat/sensors/anemometer`), o tópico da posição e orientação da vela asa (`/sail_pose`) e o tópico da posição e orientação do ASV (`/sailboat/state`); (ii) recebe a lista de *waypoints* da missão; (iii) processa a informação recebida; e (iv) publica a posição angular da cauda no tópico `/sailboat/control_flap` e a posição angular do leme no tópico `/sailboat/control_rudder`. A Figura B.4 representa o funcionamento deste *plugin*.

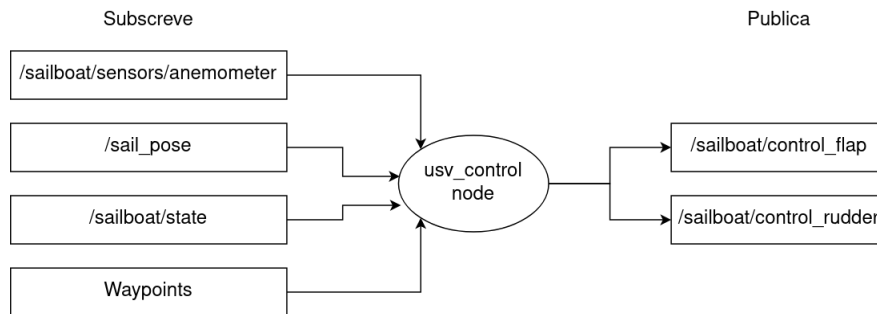


Figura B.4: Diagrama de comunicação do *plugin* `usv_control`

Anexo C

Código

C.1 Ficheiro URDF

```
1 <robot xmlns:xacro="http://ros.org/wiki/xacro" name="sailboat">
2   <link name="sailboat/base_link">
3     </link>
4   <joint name="sailboat/hull_joint" type = "fixed">
5     <parent link="sailboat/base_link"/>
6     <child link="sailboat/hull_link"/>
7     <origin xyz="0 0 0.1" rpy="0 0 0" />
8   </joint>
9
10  <!------- HULL ----->
11  <link name="sailboat/hull_link">
12    <visual>
13      <origin xyz="-0.79 0 0.26" rpy="0 0 0" />
14      <geometry>
15        <mesh filename="package://sailboat_description/models/meshes/
          hull_thick_material.dae" />
16      </geometry>
17    </visual>
18    <collision name="collision">
19      <origin xyz="-0.79 0 0.26" rpy="0 0 0"/>
20      <geometry>
21        <mesh filename="package://sailboat_description/models/meshes/hull_origin.STL"
          />
22      </geometry>
23    </collision>
24    <inertial>
25      <mass value="184.04"/>
```

```

26         <inertia ixx="5.99" ixy="0.0" ixz="0.0" iyy="230.06" iyz="0.0" izz="235.86"/>
27         <origin xyz="0 0 0.1" rpy="0 0 0"/>
28     </inertial>
29 </link>
30
31 <!------- RUDDER ----->
32 <link name="sailboat/rudder_link">
33     <visual name="visual">
34         <geometry>
35             <mesh filename="package://sailboat_description/models/meshes/leme_material.dae
36                 " />
37         </geometry>
38         <origin xyz="-0.12 0 0" rpy="0 0 0"/>
39     </visual>
40     <collision name="collision">
41         <geometry>
42             <mesh filename="package://sailboat_description/models/meshes/leme_origin.STL"
43                 />
44         </geometry>
45         <origin xyz="-0.12 0 0" rpy="0 0 0"/>
46     </collision>
47     <inertial>
48         <mass value="13.9" />
49         <inertia ixx="0.3" ixy="0.0" iyy="0.3" ixz="0.0" iyz="0.0" izz="0.1" />
50         <origin xyz="-0.12 0 0" rpy="0 0 0"/>
51     </inertial>
52 </link>
53 <joint name="sailboat/rudder_joint" type="revolute">
54     <parent link="sailboat/hull_link"/>
55     <child link="sailboat/rudder_link"/>
56     <origin xyz="-1.45 0 -0.21" rpy="0 0 0"/>
57     <dynamics damping="0.1"/>
58     <axis xyz="0 0 1"/>
59     <limit effort="100" velocity="0" lower="-1.57" upper="1.57" />
60 </joint>
61
62 <!------- KEEL ----->
63 <link name="sailboat/keel_link">
64     <visual name="visual">
65         <geometry>
66             <mesh filename="package://sailboat_description/models/meshes/keel_material.dae"
67                 />
68         </geometry>
69         <origin xyz="0 0 0" rpy="0 0 0"/>
70     </visual>
71     <collision name="collision">
72         <geometry>
73             <mesh filename="package://sailboat_description/models/meshes/keel_origin.STL"
74                 />
75         </geometry>
76         <origin xyz="0 0 0" rpy="0 0 0"/>
77     </collision>
78     <inertial>
79         <mass value="211.39" />
80         <inertia ixx="34.28" ixy="0.0" iyy="44.11" ixz="0.0" iyz="0.0" izz="10.65"/>
81         <origin xyz="-0.64 0 0" rpy="0 0 0"/>
82     </inertial>

```

```

79   </link>
80   <joint name="sailboat/keel_joint" type="revolute">
81     <parent link="sailboat/hull_link"/>
82     <child link="sailboat/keel_link"/>
83     <origin xyz="0.54 0 -0.91" rpy="0 0 0"/>
84     <axis xyz="0 0 1"/>
85     <limit effort="1000" velocity="5" lower="0" upper="0" />
86   </joint>
87
88   <!------- SAIL ----->
89   <link name="sailboat/sail_link">
90     <visual name="visual">
91       <geometry>
92         <mesh filename="package://sailboat_description/models/meshes/wingsail_v2.dae" /
93         >
94       </geometry>
95       <origin xyz="-0.08 0 0.85" rpy="0 0 0"/>
96     </visual>
97     <collision name="collision">
98       <geometry>
99         <mesh filename="package://sailboat_description/models/meshes/wingsail_E473_v2
100         .STL" />
101       </geometry>
102       <origin xyz="-0.08 0 0.85" rpy="0 0 0"/>
103     </collision>
104     <inertial>
105       <mass value="94.69" />
106       <inertia ixx="32.18" ixy="0.0" iyy="42.47" ixz="0.0" iyz="0.0" izz="10.77"/>
107       <origin xyz="-0.08 0.0 0.85" rpy="0 0 0"/>
108     </inertial>
109   </link>
110   <joint name="sailboat/sail_joint" type="revolute">
111     <parent link="sailboat/hull_link"/>
112     <child link="sailboat/sail_link"/>
113     <origin xyz="0.55 0 0.45" rpy="0 0 0"/>
114     <dynamics damping="10.0" friction="0.5"/>
115     <axis xyz="0 0 1"/>
116     <limit effort="100" velocity="70" lower="-3.1415926535897931" upper="3.1415926535897931
117     " />
118   </joint>
119
120   <!------- TAIL ----->
121   <link name="sailboat/tail_link">
122     <visual name="visual">
123       <geometry>
124         <mesh filename="package://sailboat_description/models/meshes/tail_material.dae"
125         />
126       </geometry>
127       <origin xyz="0 0 0.34" rpy="0 0 0"/>
128     </visual>
129     <collision name="collision">
130       <geometry>
131         <mesh filename="package://sailboat_description/models/meshes/tail_origin.STL" /
132         >
133       </geometry>
134       <origin xyz="0 0 0.34" rpy="0 0 0"/>
135     </collision>

```

```

131     <inertial>
132     <mass value="11.69" />
133     <inertia ixx="0.49" ixy="0.0" iyy="0.69" ixz="0.0" iyz="0.0" izz="0.2"/>
134     <origin xyz="0 0 0.34" rpy="0 0 0"/>
135     </inertial>
136 </link>
137 <joint name="sailboat/tail_joint" type="revolute">
138     <origin xyz="-1.03 0 0.64" rpy="0 0 0"/>
139     <parent link="sailboat/sail_link"/>
140     <child link="sailboat/tail_link"/>
141     <dynamics damping="1.0" friction="0.002"/>
142     <axis xyz="0 0 1"/>
143     <limit effort="10000" velocity="0" lower="-3.1415926535897931" upper="
144         3.1415926535897931" />
145 </joint>
146 </robot>

```

Listagem C.1: Ficheiro URDF com o modelo do *robot* implementado