



ALEXANDRE PIEDADE MOREIRA DE MOURA ILDEFONSO

Licenciado em Ciências da Engenharia Eletrotécnica e de
Computadores

**PLANEAMENTO DE ROTAS PARA VELEIROS
AUTÓNOMOS UTILIZANDO CAMPOS
POTENCIAIS ARTIFICIAIS**

MESTRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

Universidade NOVA de Lisboa
Novembro, 2021



PLANEAMENTO DE ROTAS PARA VELEIROS AUTÓNOMOS UTILIZANDO CAMPOS POTENCIAIS ARTIFICIAIS

ALEXANDRE PIEDADE MOREIRA DE MOURA ILDEFONSO

Licenciado em Ciências da Engenharia Eletrotécnica e de Computadores

Orientador: Professor Doutor Luís Filipe dos Santos Gomes
Professor Associado com Agregação, Universidade Nova de Lisboa

Planeamento de rotas para veleiros autónomos utilizando campos potenciais artificiais

Copyright © Alexandre Piedade Moreira de Moura Ildefonso, Faculdade de Ciências e Tecnologia, Universidade NOVA de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade NOVA de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

AGRADECIMENTOS

Em primeiro lugar, um agradecimento especial ao meu professor e orientador Luís Gomes, por todo o apoio e ajuda disponibilizada ao longo da realização da dissertação, assim como a autonomia dada na abordagem dos diferentes problemas, tornando o tema mais interessante e desafiante.

Gostaria de agradecer a todos os meus professores que estiveram comigo ao longo deste cinco anos, que, de certa forma, também contribuíram para a realização desta dissertação.

Agradeço ainda aos meus pais, à minha namorada e todos os meus amigos que me acompanharam ao longo do meu percurso e que me ajudaram a chegar até aqui.

RESUMO

O desenvolvimento de projetos na área da robótica móvel, nomeadamente a navegação autónoma de embarcações não tripuladas para aplicações civis, tem aumentado nos últimos anos, tendo como objetivo tornar os navios autónomos mais eficientes sem a necessidade da intervenção humana.

As embarcações autónoma, nomeadamente os veleiros, começaram a ganhar importância e destaque em diversas tarefas, como a exploração de vastas áreas do oceano sem colocar a vida humana em risco, na redução do número de acidentes ou ainda numa vertente mais competitiva.

Nesta dissertação é apresentada uma estratégia responsável pelo planeamento de rota de um veleiro autónomo de pequena escala, evitando obstáculos estáticos e móveis.

Trata-se de uma abordagem baseada em campos potenciais artificiais, onde, a partir de dados recebidos em tempo real como a direção do vento, a posição atual do veleiro, o próximo ponto de destino e ainda informação enviada por obstáculos presentes no ambiente marítimo, é calculado o próximo rumo de navegação, garantindo o planeamento de uma rota segura e livre de colisões.

São inicialmente apresentados os diferentes elementos do sistema proposto, tendo como referência trabalhos prévios, fazendo um enquadramento do algoritmo desenvolvido assim como os formatos de mensagens enviadas pelos diferentes tipos de obstáculos. São depois definidos os campos potenciais artificiais associados a cada elemento da navegação (ponto de destino, obstáculos) e estratégias para lidar com as zonas não navegáveis devidas à direção do vento, assim como para evitar colisões com obstáculos em diferentes situações.

O algoritmo foi desenvolvido em Python, utilizando o IDE Visual Studio Code, onde depois de implementado, foi submetido a diferentes testes, realizando o cálculo do trajeto para diferentes cenários, recorrendo ainda a um Arduino para emular o comportamento dos obstáculos.

Palavras-chave: Veleiro, Navegação Autónoma, Zonas Não Navegáveis, Campos Potenciais Artificiais, Arduino, Planeamento De Rota, Obstáculos, Python, Visual Studio Code.

ABSTRACT

The development of projects in the mobile robotics area, namely the autonomous navigation of unmanned ships for civil applications, has been rapidly growing over the past years in order to make these autonomous ships more efficient without the necessity of human intervention.

In this dissertation, a strategy responsible for the path planning of a small-scale autonomous sailboat is introduced, avoiding collisions with static and mobile obstacles.

The approach presented is based on artificial potential fields, where, thru real-time data received, such as wind direction, sailboat position, next destination point, and messages from obstacles in the environment, calculates the next goal alignment, guaranteeing a safe route free of collisions.

The different elements of the proposed system are initially described, taking previous works as reference, presenting the proposed algorithm, as well as the format of the messages sent by the different types of obstacles.

The artificial potential fields associated with the different elements present in the navigation are computed (namely goal point and obstacles), and adequate strategies to deal with the no-go zone associated with the wind direction and to avoid collisions in different scenarios are defined.

The algorithm was developed in Python using Visual Studio Code IDE. After being implemented, it was submitted to different tests calculating the path in different scenarios and using an Arduino to simulate the behavior of an obstacle by sending messages.

Keywords: Sailboat, Autonomous Navigation, No Go Zones, Artificial Potential Fields, Arduino, Path Planning, Obstacles, Python, Visual Studio Code.

ÍNDICE

Índice de Figuras	xi
Índice de Tabelas	xvi
Glossário	xvii
1 INTRODUÇÃO	1
1.1 Enquadramento e Motivação	1
1.2 Objetivo	1
1.3 Contribuição	2
1.4 Estrutura da dissertação	2
2 Estado da Arte	4
2.1 O que é um veleiro?	4
2.2 Classificação de veleiros segundo o tipo de casco	6
2.2.1 Monocasco	6
2.2.2 Catamarã	6
2.2.3 Trimarã	7
2.3 Posições de navegação	8
2.3.1 Zona não navegável	8
2.3.2 Mareação à bolina (e bolina cerrada)	9
2.3.3 Mareação pelo través ou largo	10
2.3.4 Mareação aberta ou alheta	11
2.3.5 Mareação à popa arrasada	11
2.4 Aerodinâmica do veleiro	12
2.4.1 Tipos de vento e forças existentes num veleiro	12
2.4.2 Força de propulsão	14
2.5 Veleiros autónomos e competições existentes	16
2.5.1 World Robotic Sailing Championship and Internacional Robotic Sailing	17

2.5.2	Microtransat Challenge	17
2.5.3	International Robotic Sailing Regatta	17
2.6	Algoritmos de planeamento de rota existentes	18
2.6.1	Algoritmos de pesquisa em grafos	18
2.6.2	Campo Potencial Artificial	21
2.6.3	Algoritmo baseado em colónias de formigas	21
2.6.4	Conclusões	22
2.7	Trabalhos relacionados com o algoritmo baseado em campos potenciais artificiais	23
2.7.1	COLREGS-Constrained Real-time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields	23
2.7.2	A Balance Artificial Potential Field Method for Autonomus Surface Vessel Navigation in Unstructured Riverine Environment	25
2.7.3	A Reactive COLREGs-Compliant Navigation Strategy for Autonomous Maritime Navigation	27
2.7.4	Toward an Autonomous Sailing Boat	28
2.7.5	Reactive path planning for autonomous sailboat	31
2.7.6	Conclusões	33
3	Algoritmo de Campos Potenciais Artificiais	35
3.1	Problema dos Mínimos Locais	38
3.2	Funções dos campos potenciais	39
3.2.1	Campo Potencial Atrativo	40
3.2.2	Campo Potencial Repulsivo	45
3.2.3	Campo Potencial do Vento	48
3.2.4	Aglomerados de obstáculos estáticos	57
3.2.5	Obstáculos dentro da zona não navegável	59
3.2.6	Incorrer em Zonas não Navegáveis	66
4	Sistema proposto	70
4.1	Arquitetura do sistema	70
4.1.1	Arquitetura de Hardware	70
4.1.2	Arquitetura de Software	71
4.1.3	Fluxograma do Sistema	73
4.2	Formato das mensagens	75
5	Resultados	82
5.1	Simulações em Computador	82
5.1.1	Parâmetros iniciais e constantes	82
5.1.2	Cenários sem obstáculos	83
5.1.3	Cenários com 1 obstáculo	85
5.1.4	Cenários com aglomerados de obstáculos	89

ÍNDICE

5.1.5	Cenário com obstáculos móveis	91
5.1.6	Cenários com vários obstáculos	93
5.1.7	Comparação de Resultados	95
5.2	Validações laboratoriais	96
6	Conclusões e Trabalhos Futuros	99
6.1	Conclusões	99
6.2	Trabalhos Futuros	100
	Bibliografia	101

ÍNDICE DE FIGURAS

1.1	Possível cenário dinâmico.	2
2.1	Protótipo do veleiro.	4
2.2	Orientação e diferentes partes do veleiro.	5
2.3	Exemplo de um veleiro monocasco.	6
2.4	Exemplo de um catamarã.	7
2.5	Exemplo de um trimarã.	7
2.6	Diferentes posições de navegação de um veleiro.	8
2.7	Zona não navegável.	9
2.8	Veleiro nº1 - Mareação à bolina. Veleiro nº2 - Mareação à bolina Cerrada.	9
2.9	Veleiro a bolinar.	10
2.10	Mareação pelo través ou largo.	10
2.11	Mareação aberta ou alheta.	11
2.12	Mareação à popa arrasada.	11
2.13	Representação vetorial dos diferentes ventos sentidos num veleiro	12
2.14	Vectores sentidos (força do vento) num veleiro numa situação de mareação à popa arrasada.	13
2.15	Representação vetorial de (algumas) das forças sentidas num veleiro.	13
2.16	Divisão do vento aparente (\vec{V}_a) quando em contacto com a vela e respectiva força resultante (\vec{V}_p).	14
2.17	Decomposição do vetor da força de propulsão na vela (\vec{V}_p) e da força resistiva na quilha (\vec{V}_q).	14
2.18	Representação das componentes do vetor da força de propulsão na vela (\vec{V}_p) e da força resistiva na quilha (\vec{V}_q) segundo o eixo apresentado.	15
2.19	Veleiro Autónomo FAST.	16
2.20	Exemplo do algoritmo Dijkstra.	18
2.21	Exemplo do algoritmo procura do melhor primeiro.	19
2.22	Comparação dos três algoritmos. A* na figura mais à esquerda, no meio Dijkstra e na direita Procura do Melhor Primeiro.	19

2.23	Comparação entre o algoritmo A^* (à esquerda), e Θ^* (à direita).	20
2.24	Exemplo de planeamento do caminho utilizando o método baseado em campos potenciais artificiais.	21
2.25	Exemplo do algoritmo baseado em colónias de formigas.	22
2.26	Ilustração dos diferentes cenários de encontro.	23
2.27	Estratégia utilizada para evitar colisões.	24
2.28	Estratégia para evitar colisões.	25
2.29	Resultado da simulação com base num rio real.	26
2.30	Área de efeito com um raio de D_o centrada no barco (ponto azul).	27
2.31	Esquema das zonas de vento a favor e contra. Nas zonas com pontos a azul P_h tem um valor diferente de 0.	29
2.32	Simulações com diferentes parâmetros P_h .	30
2.33	Simulação em condições reais com o trajeto seguido a amarelo.	30
2.34	Trajetórias obtidas por simulação com ventos constantes vindos de Oeste (linha com bola a preto) e vindos do Sudoeste (linha com quadrado a azul).	32
2.35	Trajetória obtida por simulação com ventos dinâmicos.	32
2.36	Trajetórias obtidas por simulação utilizando apenas o algoritmo (linha com bola a preto) e o algoritmo com o simulador das forças aerodinâmicas e hidrodinâmicas (linha com quadrado a azul).	33
3.1	Campo potencial atrativo (verde) no ponto de chegada e repulsivo (vermelho) no obstáculo.	36
3.2	Exemplo do vetor gradiente no ponto a vermelho.	37
3.3	Exemplo do vetor gradiente no ponto a vermelho, o simétrico do gradiente (seta a verde) e o ponto de chegada (ponto amarelo).	37
3.4	Ilustração de um mínimo global e local.	38
3.5	Cenário com um mínimo local devido à presença de um obstáculo côncavo.	38
3.6	Cenário com um mínimo local devido à presença de dois obstáculos muito próximos.	39
3.7	Campo potencial global. Ponto de chegada (mínimo global) representado pelo círculo a vermelho.	39
3.8	Vista lateral do campo potencial global.	39
3.9	Campo potencial cónico.	40
3.10	Campo potencial cónico num referencial de duas dimensões (azul) e o respetivo simétrico do gradiente (laranja).	41
3.11	Campo potencial parabólico.	41
3.12	Campo potencial parabólico num referencial de duas dimensões (azul) e o respetivo simétrico do gradiente (laranja).	42
3.13	Campo potencial atrativo	43
3.14	Campo potencial atrativo e o gradiente associado	44
3.15	Gradiente do campo potencial atrativo	44

3.16 Gradiente parabólico associado ao campo potencial atrativo para distâncias inferiores a δ	45
3.17 Campo potencial repulsivo gerado por um obstáculo.	46
3.18 Campo potencial repulsivo gerado por um obstáculo e o respetivo simétrico do gradiente num referencial a duas dimensões.	47
3.19 Simétrico do gradiente de um campo potencial repulsivo.	47
3.20 Exemplo de diagrama polar.	48
3.21 Diagrama polar utilizado.	49
3.22 Definição gráfica do atan2.	49
3.23 Atan2 em coordenadas polares.	50
3.24 Cálculo do ângulo da direção a seguir.	50
3.25 Gradiente repulsivo para um campo potencial cónico associado à ZNN.	51
3.26 Gradiente repulsivo associado à ZNN e as diferentes partes consideradas.	52
3.27 Caminho obtido com sucessivas mudanças de bordo para direções de navegação próximas da direção do vento.	52
3.28 Simulação de navegação na ZNN.	53
3.29 Comparação entre o caminho obtido (verde) e o pretendido (laranja).	53
3.30 Estratégia utilizada para navegar numa ZNN.	54
3.31 Simulação para um cenário com 0 obstáculos e com vento a 0°	55
3.32 Simulação para um cenário com 0 obstáculos e com vento a 30°	55
3.33 Simulação para um cenário com 0 obstáculos e com vento a 45°	56
3.34 Exemplo de um pontão de pedra.	57
3.35 Exemplo de um pontão de rochas.	57
3.36 Exemplo de um pontão decomposto em obstáculos estáticos virtuais.	58
3.37 Comparação entre as duas estratégias. Aumento do raio de influência do obstáculo (lado esquerdo), e diminuição da distância entre os diferentes obstáculos (lado direito).	58
3.38 Exemplo de navegação com um pontão composto por 12 obstáculos e vento a 0°	59
3.39 Zonas consideradas dentro da ZNN	59
3.40 Estratégia de navegação com obstáculos dentro da ZNN para a 1ª zona.	60
3.41 Simulação de navegação na ZNN com 1 obstáculo e vento a 0°	60
3.42 Simulação de navegação na ZNN com 1 obstáculo e vento a 0°	61
3.43 Limitação da estratégia utilizada para obstáculos na 1ª zona.	62
3.44 Representação de um obstáculo na 2ª zona.	62
3.45 Área de possível colisão com obstáculos na 2ª zona.	63
3.46 Cenário de possível colisão com obstáculos na 2ª zona.	64
3.47 Estratégia de navegação com obstáculos dentro da ZNN na 2ª zona.	65
3.48 Limitação da estratégia utilizada para obstáculos na 2ª zona	66
3.49 Simulação de navegação com 1 obstáculo e vento a 0°	66
3.50 Simulação de navegação com 56 obstáculos e vento a 0°	67

3.51	Simulação de navegação com 14 obstáculos e vento a 0°.	68
3.52	Matriz com a lista de obstáculos obtida para o cenário da figura 3.51.	69
4.1	Interligações dos subsistemas do arquitetura de hardware e os protocolos utilizados.	70
4.2	Diagrama de blocos do software.	71
4.3	Estrutura hierárquica do controlador	72
4.4	Fluxograma global do sistema.	73
4.5	Fluxograma do algoritmo baseado em campos potenciais artificiais	74
4.6	Representação de obstáculos.	75
4.7	Formato da mensagem enviada por um obstáculo móvel.	76
4.8	Exemplo de um cenário com um obstáculo móvel.	77
4.9	Formato da mensagem enviada por um pontão.	78
4.10	Simulação para um cenário com 1 obstáculo e com vento a 0°, sem ter em consideração o campo Manobras.	79
4.11	Simulação para um cenário com 1 obstáculo e com vento a 0°, sem ter em consideração o campo Manobras.	80
4.12	Simulação para um cenário com 1 obstáculo e com vento a 0°, tendo em consideração o campo Manobras.	80
4.13	Fluxograma do tratamento dos diferentes tipos de mensagens por parte do algoritmo.	81
5.1	Simulação para um cenário com 0 obstáculos e com vento a -10°.	83
5.2	Simulação para um cenário com 0 obstáculos e com vento a 0°	83
5.3	Simulação para um cenário com 0 obstáculos e com vento a 100°	84
5.4	Simulação de um cenário com 1 obstáculo e com vento a 90°.	85
5.5	Simulação para um cenário com 1 obstáculo e com vento a 90°.	85
5.6	Simulação para um cenário com 1 obstáculo e com vento a 90°.	86
5.7	Simulação para um cenário com 1 obstáculo e com vento a -113°.	87
5.8	Simulação para um cenário com 1 obstáculo e com vento a -113°.	87
5.9	Simulação para um cenário com 1 obstáculo e com vento a 75°.	88
5.10	Simulação para um cenário com 1 obstáculo e com vento a 0°.	88
5.11	Simulação para um cenário com 50 obstáculos e com vento a 10°.	89
5.12	Simulação para um cenário com 50 obstáculos e com vento a 0°.	90
5.13	Simulação para um cenário com 50 obstáculos e com vento a -120°.	90
5.14	Simulação para um cenário com 50 obstáculos e com vento a 0°.	91
5.15	Simulação para um cenário com 50 obstáculos e com vento a 0°.	91
5.16	Simulação para um cenário com 1 obstáculo e com vento a 0°.	92
5.17	Simulação para um cenário com 2 obstáculos e com vento a -110°.	93
5.18	Simulação para um cenário com 3 obstáculos e com vento a 0°.	93
5.19	Simulação para um cenário com 52 obstáculos e com vento a -90°.	94

5.20 Simulação para um cenário com 2 obstáculos e com vento a -10°	94
5.21 Simulação para um cenário com 28 obstáculos e com vento a -10°	94
5.22 Simulação para um cenário com 2 obstáculos e com vento a -110°	95
5.23 Ligação entre Arduino e computador.	96
5.24 Esquema da Ligação entre Arduino e computador.	96
5.25 Simulação para um cenário com 1 obstáculo e com vento a 0°	97
5.26 Simulação para um cenário com 41 obstáculos e com vento a 0°	97

ÍNDICE DE TABELAS

2.1	Legenda da figura 2.1	5
2.2	Comparação dos trabalhos realizados relacionados	33
5.1	Tabela de constantes	82

GLOSSÁRIO

GPS	Global Positioning System i
I2C	Inter-Integrated Circuit i
LoRa	Long Range i
PWM	Pulse Width Modulation i
RF	Radio-Frequência i
UART	Universal Asynchronous Receiver/Transmitter i
ZNN	Zona Não Navegável i

INTRODUÇÃO

1.1 Enquadramento e Motivação

Segundo [Noel, Shreyanka, Kumar, Bm e Akshar 2019], nos últimos anos tem-se observado um aumento de projetos e estudos na área da robótica, nomeadamente na automatização de embarcações não tripuladas, com o objetivo da exploração e navegação dos oceanos [Cruz e Alves 2008]. Recentes estatísticas [EMSA 2020] sobre os incidentes marítimos ocorridos em 2020, mostram que, cerca de 54.4% são causados por colisões (ou algum tipo de contacto), e ainda que 65.8% do número total de acidentes devém-se a ações humanas. Com o avanço tecnológico advém o objetivo de tornar os navios autónomos mais eficientes em termos energéticos, computacionais e aerodinâmicos, sem a necessidade da intervenção humana. Com o cruzamento entre a navegação autónoma e a navegação marítima, as embarcações autónomas começaram a ganhar importância e destaque em diversas tarefas, para além da exploração de vastas áreas do oceano (deixando de ser necessário colocar a vida humana em risco), na redução do número de acidentes ou ainda numa vertente mais competitiva.

Para esta dissertação, o tema é focado na vertente competitiva com veleiros de reduzidas dimensões, onde se pretende que a partir de trabalhos prévios, se desenvolva uma estratégia que permita ao veleiro da NOVA School of Science and Technology, navegar autonomamente em segurança até ao destino, evitando colisões.

De destacar que, nas simulações e resultados apresentados foi considerada uma pequena porção do globo podendo ser aproximada a uma área plana.

1.2 Objetivo

Tendo como referência a competição internacional “World Robotic Sailing Championship” [roboticsailing 2021] o objetivo passa por desenvolver um algoritmo para o veleiro da NOVA School of Science and Technology, onde, através de dados recebidos em tempo real do veleiro (direção do vento, posição atual do veleiro, próximo ponto de chegada) e informações (sob a forma de mensagem) vindas de obstáculos presentes no ambiente

de navegação, realizar o planeamento do caminho até ao destino, evitando colisões com obstáculos.

Considere-se a seguinte figura, onde se encontra representado um possível cenário, com o veleiro a ser colocado num ponto de partida, e através de valores de entrada em tempo real e mensagens enviadas pelos obstáculos é calculado um trajeto (assinalado a tracejado) livre de colisões até ao ponto de chegada (figura 1.1)

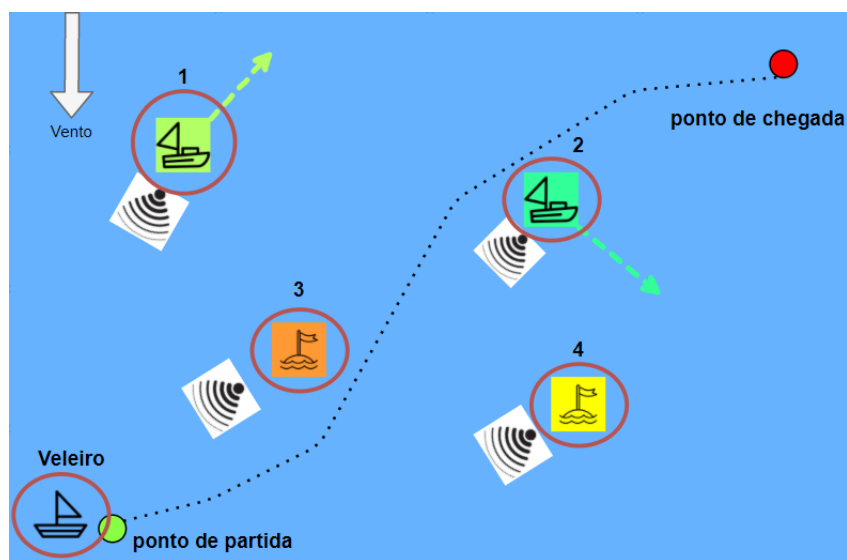


Figura 1.1: Possível cenário dinâmico.

1.3 Contribuição

A presente dissertação tem como principal contribuição o desenvolvimento de um algoritmo baseado em campos potenciais artificiais aplicável na navegação autónoma de veleiros, em particular ao veleiro da NOVA School of Science and Technology, onde a partir de dados recebidos em tempo real (como a direção do vento, as coordenadas da posição atual do veleiro, o próximo ponto de destino e ainda informação sobre obstáculos), é realizado o planeamento de uma rota segura livre de colisões.

1.4 Estrutura da dissertação

A presente dissertação encontra-se dividida em seis capítulos.

O primeiro capítulo engloba os temas de enquadramento e motivação, o objetivo da dissertação, a contribuição e a estrutura da mesma.

No segundo capítulo são abordados conceitos mais teóricos considerados relevantes para o tema. Começando por uma descrição sobre o que é e alguns dos tipos de veleiros existentes, seguidamente uma breve passagem sobre as diferentes manobras e posições

de navegação de um veleiro e ainda um pouco sobre a sua aerodinâmica. É feito um resumo sobre alguns dos algoritmos existentes, e por último, são abordados trabalhos relacionados.

O terceiro capítulo é inicialmente focado nos aspetos mais teóricos do algoritmo de campos potenciais artificiais, seguido dos diferentes campos utilizados para cada um dos elementos de navegação (obstáculos, ponto de chegada), estratégia para lidar com a zona não navegável (ZNN) e evitar colisões em diferentes situações com os diferentes tipos de obstáculo.

No quarto capítulo é descrito de forma mais pormenorizada o sistema proposto, assim como a arquitetura de hardware e software, um fluxograma que descreve o comportamento do algoritmo desenvolvido e ainda, os formatos utilizados das mensagens enviadas pelos diferentes obstáculos.

O quinto capítulo contém os resultados das simulações e testes realizados, assim como uma análise dos mesmos.

Para finalizar, no sexto capítulo são apresentadas as conclusões sobre a dissertação realizada assim como sugestões de trabalhos futuros.

ESTADO DA ARTE

2.1 O que é um veleiro?

Um veleiro é um tipo de embarcação que utiliza a força do vento para navegar. O protótipo do veleiro utilizado na dissertação encontra-se apresentado na figura 2.1, com os diferentes constituintes identificados e explicados na tabela 2.1.



Figura 2.1: Protótipo do veleiro. Imagem adaptada de: [Thundertiger 2021].

Tabela 2.1: Legenda da figura 2.1

Nº	Nome	Descrição
1	Vela Mestre	Vela localizada atrás do mastro principal.
2	Bujarrona	Vela com um formato triangular, envergada no estai da proa.
3	Mastro	Peça cilíndrica alta que sustenta as velas.
4	Retranca	À semelhança do mastro, trata-se também de uma peça cilíndrica de material resistente (madeira, fibra de carbono ou alumínio) que se encontra presa ao mastro principal, e tem a função de ajudar na armação da vela.
5	Leme	Tem como função orientar o veleiro na direção correta.
6	Quilha	A quilha, localizada na parte inferior, é uma das peças fundamentais para o equilíbrio e para o movimento do veleiro, equilibrando com uma força contrária à força de propulsão existente nas velas.
7	Casco	O casco é o “corpo” do veleiro.
8	Estai da popa	Fio de aço inoxidável que ajuda a manter o mastro direito. Encontra-se ligado à popa e à ponta do mastro.
9	Estai da proa	Tem as mesmas funções que o estai da popa, sendo que, a única diferença é que este se encontra ligado à proa.

As diferentes partes de uma embarcação [Centre 2015] podem ser observadas analisando a figura 2.2. A Proa e a Popa, são as partes da frente (da vante) e de trás (da ré) respetivamente. Estando virado para a Proa, do lado direito da embarcação (a partir da linha a tracejado a azul) é Estibordo e o lado esquerdo Bombordo, por último, em ambos os bordos, existe a Amura (parte curva junta à Proa), o Través (parte lateral da embarcação), e a Alheta (parte curva junto à Popa).

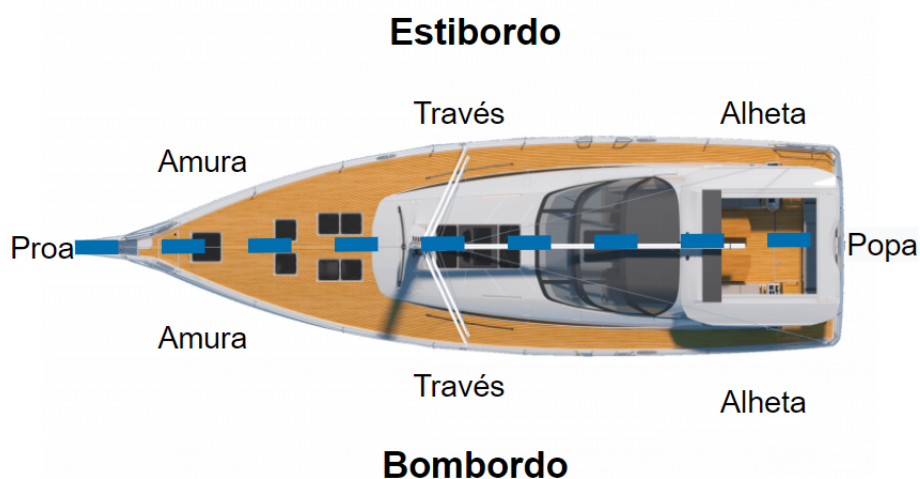


Figura 2.2: Orientação e diferentes partes do veleiro. Imagem: [jeanneau 2021].

2.2 Classificação de veleiros segundo o tipo de casco

Dentro dos veleiros, quando se fala no casco, é possível dividi-los em três tipos, o monocasco, catamarã e trimarã.

2.2.1 Monocasco

Como o próprio nome indica, trata-se de um veleiro com apenas um casco, criado para navegar em águas abertas. Uma das suas grandes vantagens [Vasconcelos 2015][Quant 2009] é a elevada estabilidade e segurança, por outro lado apresenta falta de capacidade em manter um bom desempenho quando exposto a condições de navegação adversas. Em termos de inclinação podem chegar até aos 45° mantendo a navegabilidade. Um exemplo de um veleiro com apenas um casco encontra-se representado na figura 2.3.



Figura 2.3: Exemplo de um veleiro monocasco. Imagem: [SailingEurope 2021].

2.2.2 Catamarã

Neste tipo de veleiro existem dois cascos. O facto de os cascos estarem distanciados um do outro, permite que o catamarã, quando comparado com o veleiro de monocasco, apresente maior estabilidade. Em termos de inclinação, chegam a um máximo de 10°. Note-se ainda que, alguns catamarãs não possuem uma quilha, tornando-os mais leves quando comparados com os veleiros de monocasco [Davis e Holloway 2007]. Um exemplo de um catamarã pode ser observado na figura 2.4.



Figura 2.4: Exemplo de um catamarã. Imagem: [pixabay 2021].

2.2.3 Trimarã

Como o próprio nome indica, este tipo de veleiro possui três cascos, um principal e dois laterais, como se pode observar na figura 2.5. Consegue [Vasconcelos 2015][Quant 2009] atingir velocidades superiores ao catamarã mantendo sempre um elevado nível de estabilidade .



Figura 2.5: Exemplo de um trimarã. Imagem: [SailRaceCrew 2021].

2.3 Posições de navegação

Existem diferentes posições de navegação, que variam consoante a posição do veleiro relativamente à direção do vento (figura 2.6), cada uma com características específicas que permitem o veleiro navegar em diferentes direções e velocidades, consoante o objetivo pretendido. De seguida será feita uma breve descrição de cada uma.

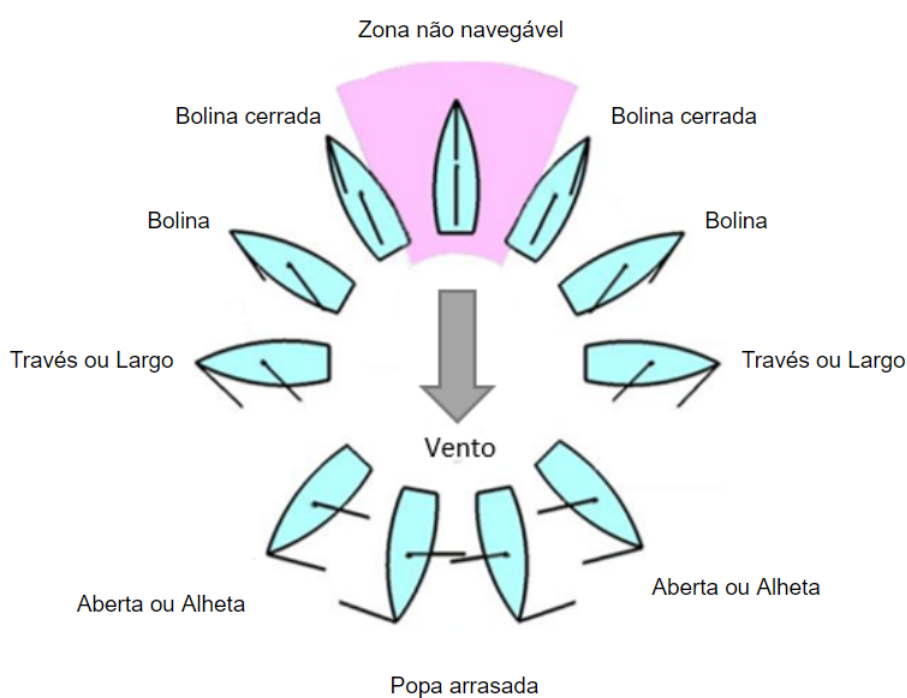


Figura 2.6: Diferentes posições de navegação de um veleiro.
Imagem: [pinterest 2021].

2.3.1 Zona não navegável

Trata-se de uma zona [Universal_yachting 2021] onde os veleiros não conseguem utilizar a força do vento a seu favor para navegar. O facto de esta zona provocar uma redução da velocidade do veleiro leva a que seja utilizada quando se pretende ancorar ou recolher as velas.

A zona não navegável é obtida aproximadamente calculando 45° ¹ para a direita e para a esquerda a partir da direção do vento, como se pode observar na figura 2.7.

¹Este ângulo pode variar consoante as características do veleiro

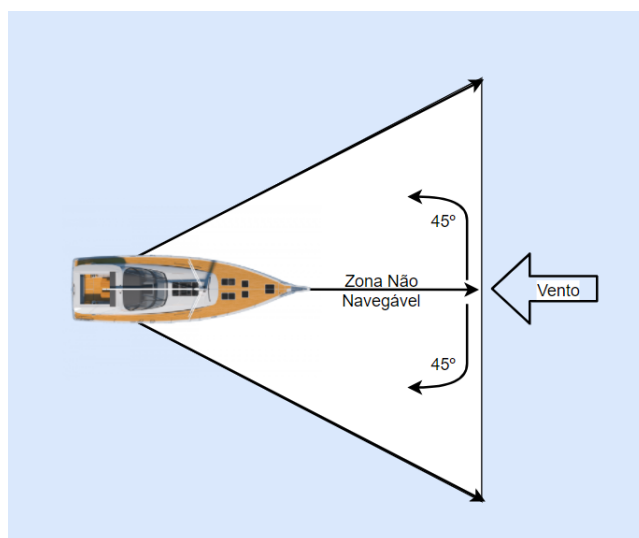


Figura 2.7: Zona não navegável. Imagem: [jeanneau 2021].

2.3.2 Mareação à bolina (e bolina cerrada)

Posição de navegação utilizada quando se pretende navegar na direção contrária ao vento. A bolina cerrada é a navegação no limite da ZNN (veleiro nº2 na figura 2.8). Nestas situações é usual bolinar, onde se muda o bordo sucessivamente, sendo realizado um percurso em “zig-zag”, de modo a atingir o ponto de chegada, como se observa na figura 2.9. Nesta posição, as velas deverão ir caçadas ² ao máximo.

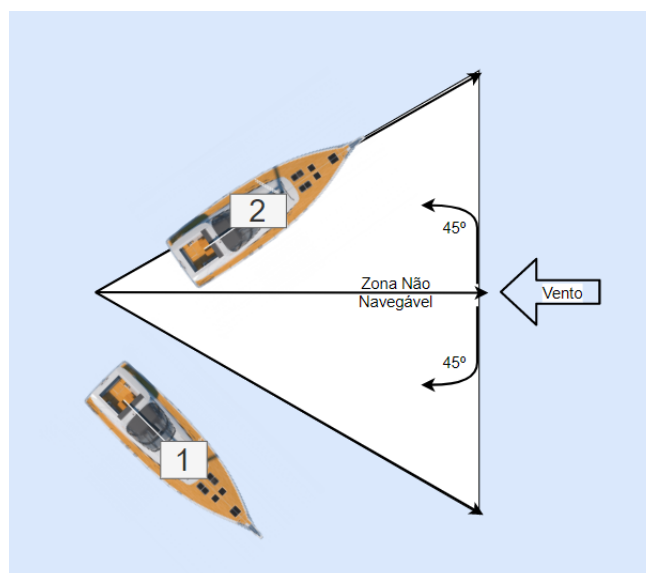


Figura 2.8: Veleiro nº1 - Mareação à bolina. Veleiro nº2 - Mareação à bolina Cerrada. Imagem: [jeanneau 2021].

²Termo náutico usado para indicar que as velas deverão ser esticadas de modo a aproveitar melhor o vento

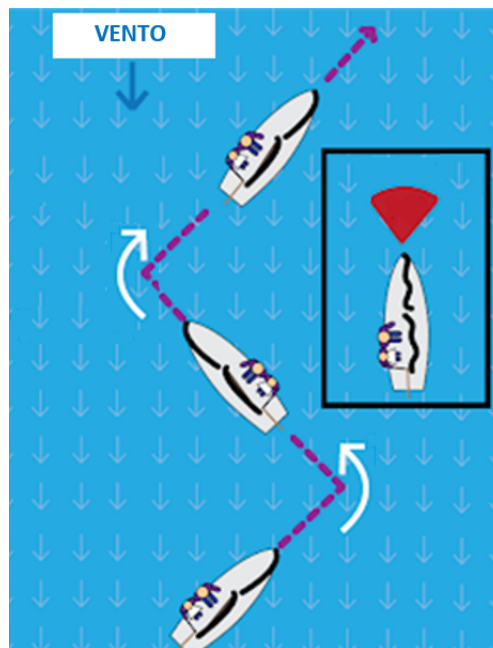


Figura 2.9: Veleiro a bolinar. Imagem: [sailnator 2021].

2.3.3 Mareação pelo través ou largo

Neste tipo de mareação o vento incide na zona lateral do veleiro, ou perpendicularmente (90°) (figura 2.10). Esta posição permite atingir velocidades apreciáveis garantindo ao mesmo tempo o controlo do veleiro.

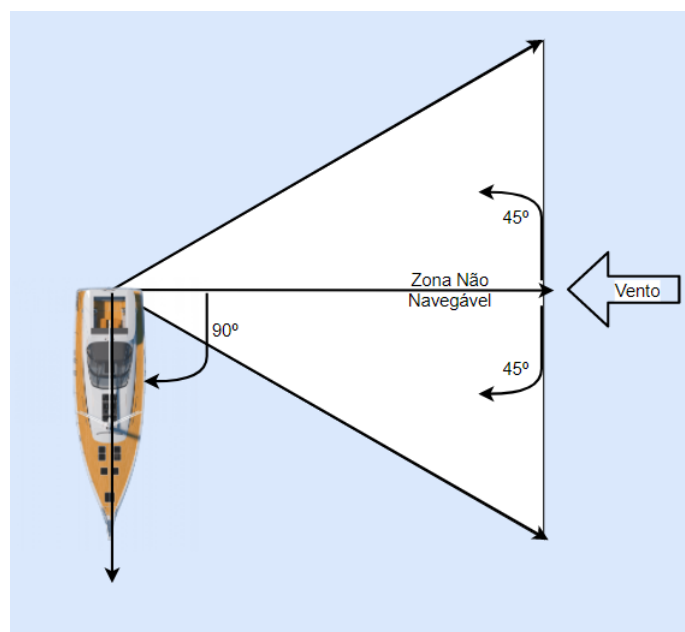


Figura 2.10: Mareação pelo través ou largo. Imagem: [jeanneau 2021].

2.3.4 Mareação aberta ou alheta

Numa situação de mareação aberta, o vento incide sobre a alheta (figura 2.11). A retranca deverá estar posicionada numa linha perpendicular à linha do vento de modo que a vela vá com folga.

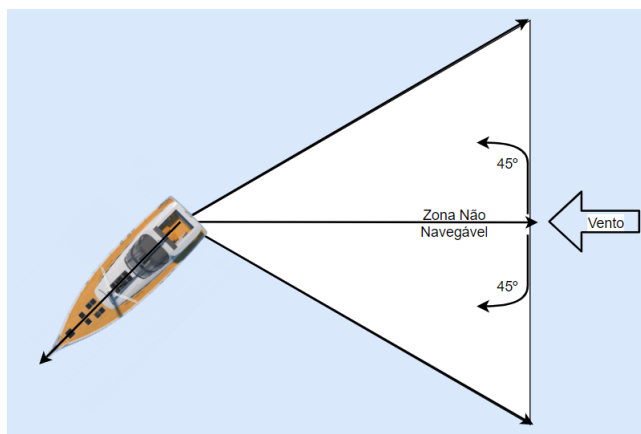


Figura 2.11: Mareação aberta ou alheta. Imagem: [jeanneau 2021].

2.3.5 Mareação à popa arrasada

Nesta última posição, o vento incide diretamente pela popa, fazendo um ângulo de (aproximadamente) 180° em relação à direção do vento (figura 2.12). As velas encontram-se folgadas e torna-se necessário avaliar bem o vento, de modo a evitar situações onde a retranca mude de bordo de forma repentina.

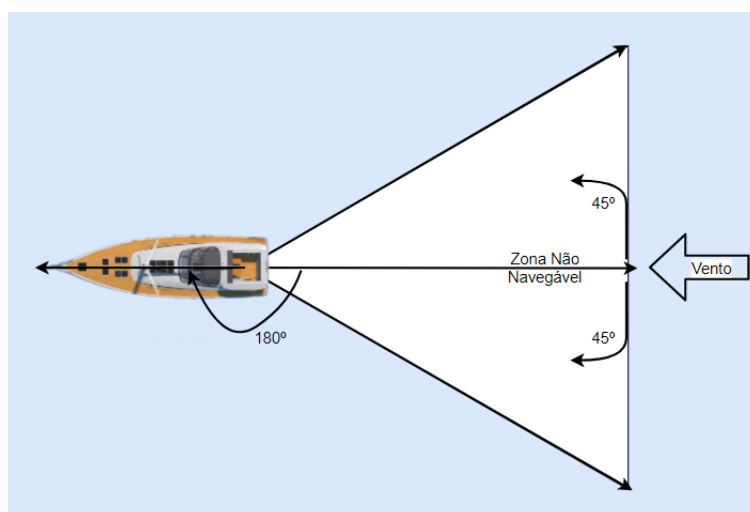


Figura 2.12: Mareação à popa arrasada. Imagem: [jeanneau 2021].

2.4 Aerodinâmica do veleiro

Desde cedo [Mendonça 2013] que o ser humano compreendeu que, colocando velas numa embarcação e aproveitando a força do vento, conseguia fazer com que o mesmo navegasse. Nessa altura, a navegação tratava-se de algo muito rudimentar, estando limitados pela falta de conhecimento.

Nesta secção será feita uma breve passagem por (algumas) das forças que atuam no veleiro.

2.4.1 Tipos de vento e forças existentes num veleiro

Como foi já referido, o vento é um dos principais intervenientes na navegação de um veleiro. Quando se navega [Jansen, Van Deursen e Howe 2012], é possível sentir diferentes “tipos” de vento, nomeadamente:

- **Vento do movimento** - Vento associado ao movimento do veleiro.
- **Vento real** - Vento sentido quando o veleiro se encontra completamente parado.
- **Vento aparente** - Somatório do vento real com o vento do movimento.

Como indicado, o **vento aparente** (\vec{V}_a) é dado pela soma do vetor do **vento associado ao movimento do veleiro** (\vec{V}_m) com o **vetor do vento real** (\vec{V}_r), como é possível verificar na figura 2.13.



Figura 2.13: Representação vetorial dos diferentes ventos sentidos num veleiro. Imagem Adaptada de: [dreamstime 2021].

A partir da figura 2.13, é possível concluir que, com o aumento do vetor do vento associado ao movimento do veleiro (\vec{V}_m), aumentará também o vetor de vento aparente (\vec{V}_a). Um outro exemplo interessante de se analisar é o caso da mareação à popa arrasada, ou seja, quando se navega na direção a favor do vento. Numa situação limite teremos os vetores assinalados na figura 2.14.



Figura 2.14: Vetores sentidos (força do vento) num veleiro numa situação de mareação à popa arrasada. Imagem Adaptada de: [dreamstime 2021].

Para além das forças associadas ao vento aparente (\vec{V}_a), ao movimento do veleiro (\vec{V}_m) e ao vento real (\vec{V}_r), existe ainda uma força a ser exercida na vela (\vec{V}_p) e na quilha (\vec{V}_q), denominada por **Força propulsora e resistiva** [Bisbee e Halloran 1995], que resulta da passagem do vento nas velas e na quilha respetivamente, e com uma direção perpendicular à direção do vento, como se observa na figura 2.15.

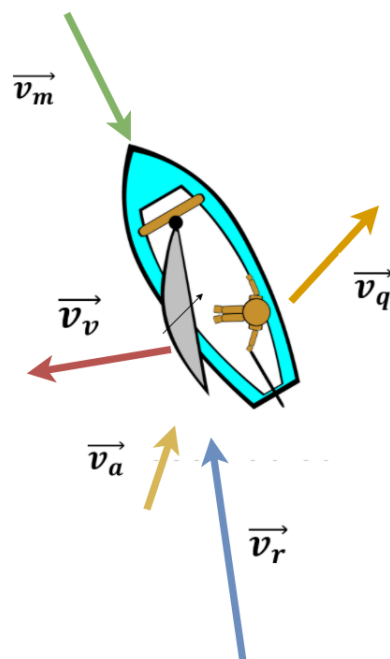


Figura 2.15: Representação vetorial de (algumas) das forças sentidas num veleiro. Imagem Adaptada de: [pexels 2021].

2.4.2 Força de propulsão

O vento aparente (\vec{V}_a) quando em contacto com velas, “divide-se” (figura 2.16), havendo uma parte que se desloca pelo lado de dentro da vela (setas a azul), e outra por fora (setas a laranja).

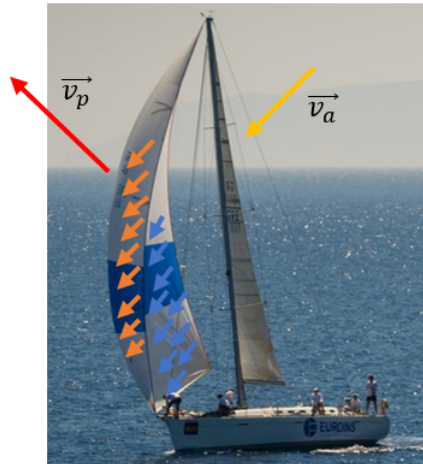


Figura 2.16: Divisão do vento aparente (\vec{V}_a) quando em contacto com a vela e respectiva força resultante (\vec{V}_p). Imagem Adaptada de: [pexels 2021].

A “parte” que se desloca por fora, percorrerá uma distância maior do que a que se desloca por dentro, e quando em contacto com o ar que se movimenta à sua volta, este faz com que a sua velocidade de deslocamento seja superior à velocidade da parte de dentro. Esta diferença de velocidades resulta numa força de propulsão (o raciocínio é análogo para o caso da quilha, mas em vez de ar temos um fluido, água).

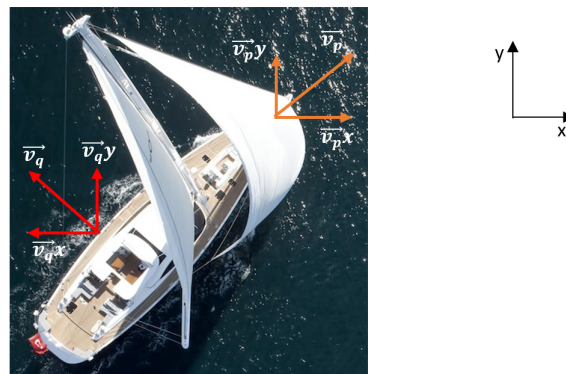


Figura 2.17: Decomposição do vetor da força de propulsão na vela (\vec{V}_p) e da força resistiva na quilha (\vec{V}_q). Imagem Adaptada de: [charterworld 2021].

Analisando agora uma representação de um veleiro com a força propulsora que atua ao nível das velas (\vec{V}_p), e a força resistiva que atua na quilha (\vec{V}_q), decompondo-as segundo o eixo indicado na figura 2.17.

Partindo da figura 2.17 e reduzindo o veleiro ao seu centro de massa, obtemos a figura 2.18.

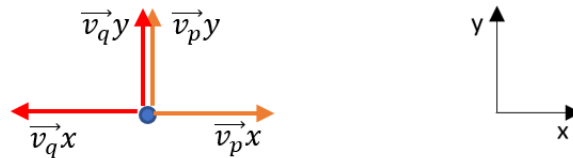


Figura 2.18: Representação das componentes do vetor da força de propulsão na vela (\vec{V}_p) e da força resistiva na quilha (\vec{V}_q) segundo o eixo apresentado.

É possível concluir que [Anderson 2001]:

- As componentes em x (segundo o referencial adotado) anulam-se, pois estamos perante 2 vetores com o **mesmo** módulo, mas com **sentidos opostos**.
- As componentes segundo o eixo y são então responsáveis pelo movimento do veleiro na direção pretendida.

Nota. Existem inúmeras forças que não foram faladas ou mencionadas (como por exemplo as resistivas), pelo que este sub-capítulo teve como principal objetivo dar apenas uma breve e simples explicação da física que permite um veleiro navegar.

2.5 Veleiros autónomos e competições existentes

Um veleiro autónomo, segundo [Stelzer e Jafarmadar 2011], fica caracterizado por ter apenas o vento como fonte de energia para se deslocar, não ser controlado remotamente e com todo o sistema de controlo presente no veleiro. Dentro dos veleiros autónomos podemos destacar aqueles que são totalmente autónomos, querendo isto dizer, que utilizam (por exemplo) energia solar para “alimentar” todo o sistema. Dentro dos veleiros autónomos é possível destacar alguns projetos realizados, como por exemplo, FASt. O FASt [Alves e Cruz 2008] trata-se de um veleiro autónomo (figura 2.19) realizado por um grupo de docentes e estudantes de Engenharia Electrotécnica da Faculdade de Engenharia da Universidade do Porto (FEUP), com cerca de 2.5 metros, tendo participado em diversas competições, chegando a ganhar em 2012 a World Robotic Sailing Championship and Internacional Robotic Sailing [roboticsailing 2021].



Figura 2.19: Veleiro Autónomo FASt. Imagem: [Alves e Cruz 2008].

Em relação ao hardware Alves e Cruz 2008, o sistema de computação é baseado numa FPGA de baixo consumo energético. Possui diversos sensores necessários à navegação tais como: receptor de GPS, um anemómetro e ainda sensores da temperatura e velocidade da água, entre outro.

Existem inúmeras competições de robótica dedicadas aos veleiros autónomos. Seguidamente serão apresentadas algumas dessas competições.

2.5.1 World Robotic Sailing Championship and Internacional Robotic Sailing

Trata-se da maior competição mundial de veleiros autónomos com comprimentos até quatro metros [roboticsailing 2021]. Até à data houve doze edições, sendo que duas delas foram realizadas em Portugal (2009 e 2016). Antes e/ou depois dos dias da competição, é realizada uma conferência científica onde são discutidos os recentes desenvolvimentos e estudos feitos na área. Nesta competição, dividida em 4 desafios: corrida de frotas, zonas para evitar colisões, exploração de uma área em conjunto e ainda um desafio onde é necessário identificar e perseguir um determinado obstáculo. testando diferentes habilidades em variados desafios, pondo à prova a robustez do sistema do veleiro.

2.5.2 Microtransat Challenge

A Microtransat [microtransat 2021] [Colin Sauze 2021] trata-se de uma competição destinada a duas classes de veleiros, a que apenas utiliza o vento como fonte de propulsão e a classe que utiliza outras formas de propulsão para além do vento. O desafio passa por realizar a travessia entre a Europa e a América do Norte (ou vice-versa) no menor tempo possível.

2.5.3 International Robotic Sailing Regatta

A International Robotic Sailing Regatta (SailBot) [sailbot 2021] trata-se de uma competição semelhante à “World Robotic Sailing Championship and Internacional Robotic Sailing”, onde veleiros (até dois metros) de navegação autónoma ou totalmente autónomos competem numa série de desafios, como por exemplo numa corrida onde é permitido o controlo remoto com o objetivo de comprovar a robustez do sistema de controlo, e ainda uma corrida de navegação autónoma. Todas os desafios poderão ser realizados com recurso ao controlo manual, havendo no entanto uma penalização de pontos. Destinado a equipas de Universidades, colégios e escolas secundárias. A primeira edição realizou-se em 2006 (Kingston, Canada), e a última em 2019 (Worcester, USA).

2.6 Algoritmos de planeamento de rota existentes

Quando a rota a seguir é calculada, no caso dos veleiros autónomos, torna-se de extrema importância ter em conta as condições relativamente ao ambiente de navegação [Jing et al. 2020], como por exemplo, a direção do vento ou a existência de correntes, pois são fatores que irão influenciar a velocidade e o rumo escolhido pelo veleiro, e consequentemente a rota escolhida. Para além de uma rota que permita que o veleiro chegue ao ponto de chegada, é necessário que esta seja a mais eficiente.

2.6.1 Algoritmos de pesquisa em grafos

Neste tipo de algoritmos é feita uma discretização do mapa em pontos com um peso associado. Trata-se de um exemplo de algoritmo muito utilizado na área do planeamento de caminho [Daniel, Nash, Koenig e Felner 2010] pois os grafos são estruturas de dados relativamente simples de processar, permitindo obter bons resultados com o uso de algoritmos pouco complexos. Outra vantagem é o facto de se poder otimizar a precisão do caminho obtido aumentando a resolução da rede. Depois de identificado o ponto de partida, chegada e os obstáculos, estes são ligados a partir de uma característica (normalmente vértices) ou pontos aleatórios, no caso do algoritmo de caminhos probabilísticos [Maria Medeiros Alves 2017].

O algoritmo proposto por Dijkstra [Dijkstra 1959] em 1959 é considerado um dos primeiros e mais importantes algoritmos nesta área [De e Guglieri 2012], permitindo encontrar o caminho mais curto num grafo. De uma forma geral, o algoritmo começa por procurar os vizinhos do ponto atual, colocando numa fila de prioridade consoante o custo associado a cada nó, visitando cada um apenas uma vez. Ao chegar ao nó final, é feita uma análise à fila de prioridade com todos os nós que se utilizaram para chegar até ele, obtendo o caminho.

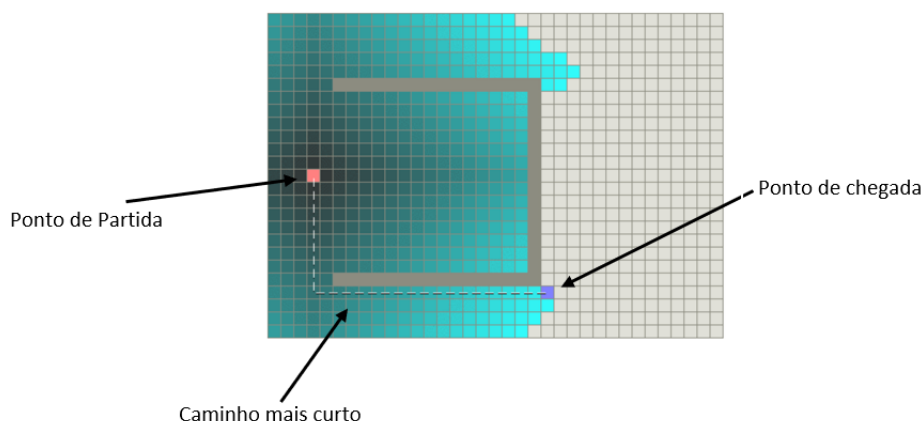


Figura 2.20: Exemplo do algoritmo Dijkstra. Imagem: [stanford 2021].

No exemplo da figura 2.20 verifica-se que todos os pontos com uma cor azulada representam os nós analisados pelo algoritmo.

O algoritmo Procura do Melhor Primeiro (Best-First-Search) [Drozdov, Kim e Lazebnik 2011] funciona de maneira semelhante ao algoritmo Dijkstra, mas em vez de avaliar o custo do percurso, escolhe o nó que estiver mais próximo do ponto de chegada, reduzindo o tempo de procura.

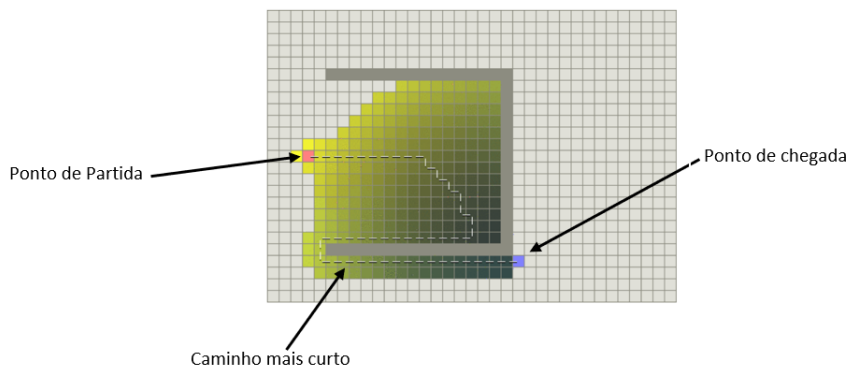


Figura 2.21: Exemplo do algoritmo procura do melhor primeiro.
Imagem: [stanford 2021].

Uma desvantagem que este algoritmo apresenta, é o facto da pesquisa ser feita em direção ao ponto de chegada de uma forma “cega”, podendo por vezes não ser o trajeto correto ou o mais curto, como se pode verificar na figura 2.21.

O algoritmo A* [Rosen 1961][stanford 2020] resulta de uma combinação do algoritmo procura do melhor primeiro com o algoritmo Dijkstra, onde, para se escolher o nó é tido em conta o custo do trajeto e a distância a que esse nó se encontra do ponto de chegada. Na figura 2.22, é feita uma comparação entre os três algoritmos.

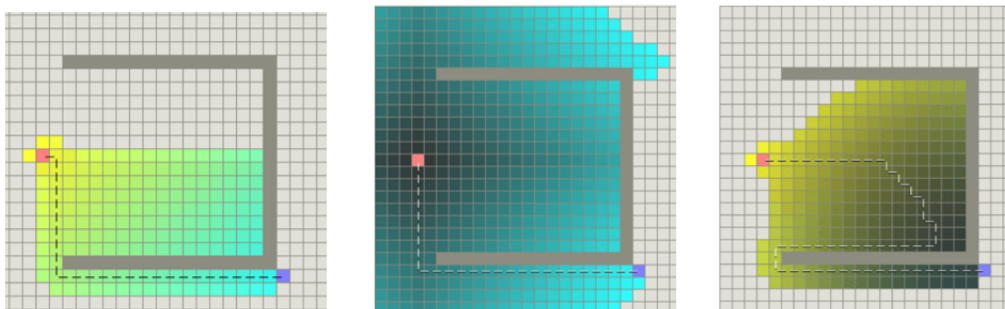


Figura 2.22: Comparação dos 3 algoritmos. A* na figura mais à esquerda, no meio Dijkstra e na direita Procura do Melhor Primeiro. Imagem: [stanford 2021].

Em [Saoud e M.-d. Hua s.d.] é usada uma variação do algoritmo de Dijkstra, onde o trajeto é calculado num mapa de grafos a partir do algoritmo probabilístico [Maria Medeiros Alves 2017] (Probabilistic Road Map), onde são gerados pontos aleatórios no mapa, distribuídos e interligados. Para o cálculo do trajeto, para além do custo, também o tempo necessário para percorrer cada um dos segmentos que compõe o trajeto é tido em conta.

Em [Erckens, Büsser, Pradalier e Siegwart 2010], utilizou-se o algoritmo A* modificado, onde cada ponto do caminho era escolhido consoante o seu “custo”, determinado por parâmetros como: o tempo até chegar a esse mesmo nó (distância/velocidade), o tempo a que se encontra do ponto de chegada ou as manobras que seriam necessárias (virar de bordo ou cambar). Atendendo à figura 2.23, no resultado da esquerda, verifica-se que os segmentos de reta que compõe o trajeto (vermelho a tracejado) apresentam uma relação de múltiplos de 45° ($0^\circ, 45^\circ, 90^\circ$, etc.), sendo por vezes gerado um trajeto que não representa na realidade o trajeto mais curto ou eficiente. Para resolver este problema foi desenvolvido o algoritmo Theta* [Daniel, Nash, Koenig e Felner 2010], que encontra o melhor trajeto sem as restrições dos ângulos que existem no algoritmo A*, como se pode verificar no resultado da direita.

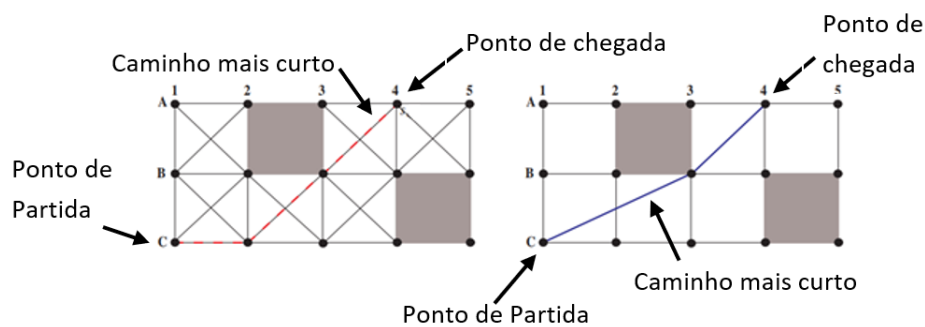


Figura 2.23: Comparação entre o algoritmo A* (à esquerda), e Theta* (à direita). Imagem: [Daniel, Nash, Koenig e Felner 2010].

Um dos problemas do algoritmo Dijkstra [Liu, Cao e Li 2021] é a fraca adaptabilidade a ambientes dinâmicos, o que leva a um constante replaneamento de todo o caminho sempre que um obstáculo destes é encontrado, ou o mapa sofre alguma alteração, visto que estes algoritmos se baseiam em pré-processamento. O algoritmo A* Dinâmico [Likhachev, Ferguson, Gordon, Stentz e Thrun 2005], trata-se de uma evolução do algoritmo A*, adaptado para ambiente dinâmicos, onde o custo (peso) associado a cada nó sofre alterações ao longo do tempo. Outro exemplo de algoritmo utilizado em ambientes dinâmicos é o caso de ADA*(Anytime Dynamic A*) [Silva, Siebra e Nascimento 2016], que planeia e replaneia um caminho (nem sempre o mais otimizado) com base no tempo que tem disponível, reutilizando informação de pesquisas prévias, contudo, apresenta um consumo computacional elevado.

2.6.2 Campo Potencial Artificial

Proposto por *Khatib* em 1985 [Khatib 1985], trata-se de um método baseado nos modelos físicos de campos potenciais, onde o robô se encontra sob o efeito de dois tipos de forças: uma atrativa (que atrai o robô na direção do ponto de chegada), e uma repulsiva (que “afasta” o robô dos obstáculos). Na caso dos veleiros autônomos, em [Plumet, Pêtrès, Gas e Ieng 2014] ou [Petres, Romero-Ramirez e Plumet 2011], utilizou-se esta estratégia, com algumas forças extra: uma para simular as zonas não navegáveis e outra para representar o “custo” de uma manobra (virar de bordo ou cambar).

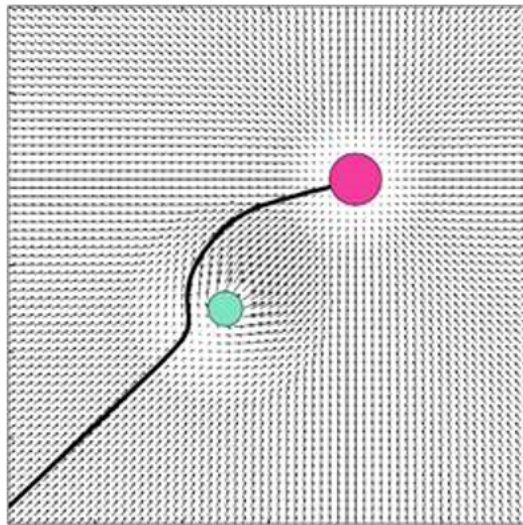


Figura 2.24: Exemplo de planeamento do caminho utilizando o método baseado em campos potenciais artificiais. Imagem: [mcgill 2021].

A linha a preto representa o trajeto seguido por um robô até ao ponto de chegada (ponto rosa), responsável por exercer uma força atrativa sobre o robô, e o ponto verde, trata-se de um obstáculo que irá exercer uma força repulsiva no robô, como se pode observar na figura 2.24. A principal desvantagem deste algoritmo é a existência dos **mínimos locais** [Park e Lee 2003], que são pontos que ficam caracterizados por terem uma força atrativa e repulsiva igual. Para além de ser um algoritmo [Plumet, Saoud e M. D. Hua 2013] rápido e fiável, pode ser implementado em microcontroladores mantendo um consumo energético baixo.

2.6.3 Algoritmo baseado em colónias de formigas

Segundo [Reshamwala 2013], trata-se de um algoritmo que pretende replicar o comportamento das colónias de formigas quando descobrem o caminho mais curto desde o local da colónia até à comida. Este caminho [Maria Medeiros Alves 2017] é descoberto à custa da uma substância chamada feromona.

Cada formiga começa por percorrer um caminho aleatoriamente, quando chegam à comida (C) voltam para o seu ninho (N), e durante este processo, são deixados vestígios de feromona, acontecendo o mesmo nos caminhos alternativos, no entanto, no caminho mais curto o vestígio da substância será reforçado, pois as formigas percorrerão este caminho mais rapidamente. Com uma maior presença de feromona maior será a probabilidade de as próximas formigas passarem por esse mesmo caminho. Este processo repete-se até que o vestígio de feromona nos caminhos mais longos se evapore, ficando com o caminho mais curto (figura 2.25).

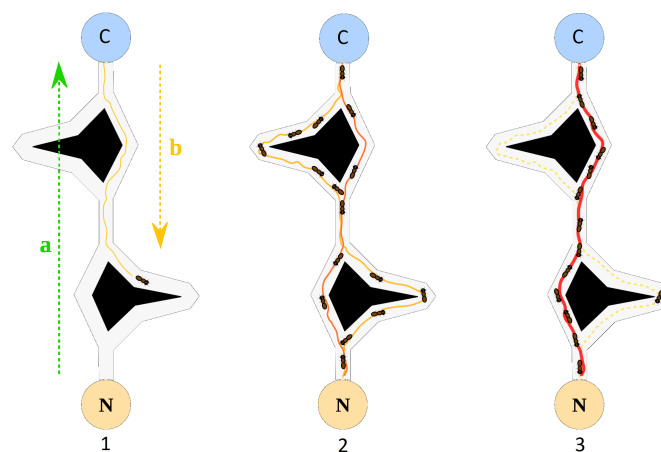


Figura 2.25: Exemplo do algoritmo baseado em colônias de formigas. Imagem: [Liu, Cao e Li 2021].

2.6.4 Conclusões

Depois de apresentados alguns dos algoritmos existentes na área do planeamento de caminho, decidiu-se optar por utilizar o algoritmo baseado em campos potenciais artificiais, por se tratar de um algoritmo fiável, rápido e capaz de calcular um caminho em tempo real em ambientes dinâmicos [Plumet, Saoud e M. D. Hua 2013]. Para além do que foi referido, pode ser implementado em microcontroladores mantendo um consumo energético baixo.

Tendo como objetivo desenvolver um sistema robusto capaz de lidar com ambientes dinâmicos, conclui-se que é necessário haver uma rápida adaptação, o que não seria possível com algoritmos baseados na ideia de Dijkstra como A^* ou Θ^* , onde seria necessário recalcular todo o caminho.

2.7 Trabalhos relacionados com o algoritmo baseado em campos potenciais artificiais

Nesta secção serão brevemente apresentadas as principais contribuições de um conjunto de publicações consideradas relevantes, onde foi utilizada uma abordagem baseada em campos potenciais artificiais. Os trabalhos analisados são os seguintes:

- “COLREGS-Constrained Real-time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields” - [Lyu e Yin 2019]
- “A Balance-Artificial Potential Field Method for Autonomus Surface Vessel Navigation in Unstructured Riverine Environment” - [Hong e Arshad 2015]
- “A Reactive COLREGs-Compliant Navigation Strategy for Autonomous Maritime Navigation” - [Naeem, Henrique e Hu 2016]
- “Toward an Autonomous Sailing Boat” - [Plumet, Pêtrès, Gas e Ieng 2014]
- “Reactive path planning for autonomous sailboat” - [Petres, Romero-Ramirez e Plumet 2011]

2.7.1 COLREGS-Constrained Real-time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields

Neste projeto foi desenvolvido um algoritmo para barcos autónomos [Lyu e Yin 2019] presentes num ambiente de navegação dinâmico, ou seja, obstáculos estáticos e móveis estão presentes, tendo em conta algumas das regras de navegação (COLREGS). Para determinar o tipo de situação em que a embarcação se encontra, o espaço é dividido de acordo com a figura 2.26.

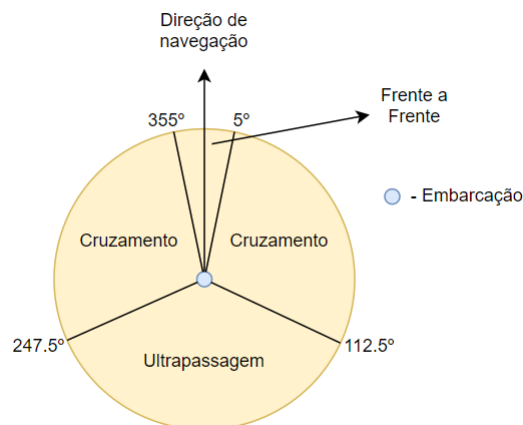


Figura 2.26: Ilustração dos diferentes cenários de encontro.
Referência: [Lyu e Yin 2019].

Para uma embarcação cujo ângulo de incidência se encontre entre 5° e 112.5° ou entre 247.5° até 355° encontra-se numa situação de cruzamento de rotas, se o ângulo estiver entre 112.5° e 247.5° trata-se de uma situação de ultrapassagem, e por último, se o ângulo estiver entre 355° e 5° uma situação frente a frente (figura 2.26). Desta forma, sabendo a localização de uma outra embarcação nas proximidades, torna-se possível avaliar a situação e agir de forma a cumprir as regras de navegação.

Para a detecção de colisões, a cada obstáculo (estático ou móveis) é associado um círculo de segurança à sua volta, cujo raio depende da velocidade com que o obstáculo se movimenta, o espaço de navegação ou condições de visibilidade (por ex. nevoeiro). Caso a direção pretendida pelo barco cruze este círculo, passa a haver risco de colisão.

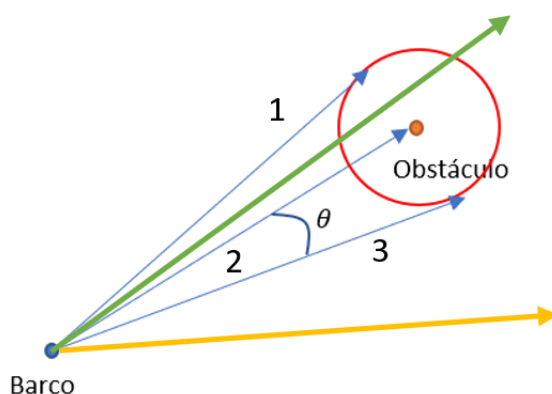


Figura 2.27: Estratégia utilizada para evitar colisões.

Atendendo à figura 2.27, caso o barco (ponto azul) navegue na direção apresentada pela seta verde verifica-se que esta cruza o círculo de segurança (círculo a vermelho) do obstáculo (ponto laranja), havendo risco de colisão. Por outro lado, se se considerar que a direção pretendida é a da seta a amarelo, conclui-se que esta não intersecta o círculo de segurança, não havendo risco de colisão. O ângulo que define se existe risco de colisão nesta fase é θ , definido entre as setas 2-3 e 2-1.

De modo a definir diferentes graus de colisão, a distância de segurança à volta do barco é dividida consoante a proximidade do obstáculo e o campo potencial repulsivo associado, conforme representado na figura 2.28.

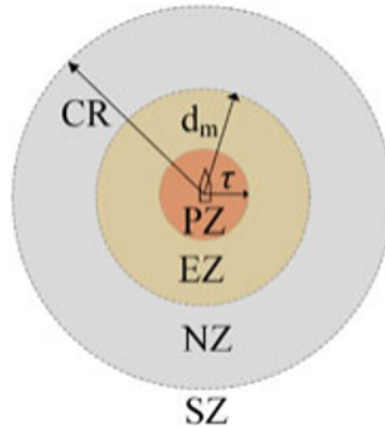


Figura 2.28: Estratégia para evitar colisões. Imagem: [Lyu e Yin 2019].

SZ (Safe Zone) - não existe risco de colisão e nenhum campo potencial repulsivo.

NZ (Negotiation Zone) - zona com um potencial repulsivo para obstáculos, se a distância ao obstáculo $\geq d_m$ e houver interseção entre a direção a seguir e o círculo associado ao obstáculo.

EZ (Emergency Zone) - zona com potencial repulsivo para situações de emergência se a distância_ao_obstáculo $\leq d_m$.

PZ (Prohibited zone) – zona de acesso proibida com um potencial repulsivo elevado.

De modo a obedecer às regras de navegação, as equações que descrevem a intensidade, força e direção associada a cada situação são feitas de modo a “forçar” a direção que se pretende a seguir. Em termos de resultados, foram realizadas com sucesso simulações em ambientes com obstáculos estáticos e móveis, cumprindo as regras de navegação e evitando colisões.

2.7.2 A Balance Artificial Potential Field Method for Autonomus Surface Vessel Navigation in Unstructured Riverine Environment

Projeto realizado com o objetivo de desenvolver um controlador para um barco navegar ao longo de um rio com uma velocidade constante, mantendo-se no centro do mesmo, evitando obstáculos estáticos que surjam ao longo do caminho.

Para evitar os obstáculos durante o trajeto, foi-lhes atribuído uma força repulsiva, dada pela equação 2.1.

$$F_{rep} = \begin{cases} K_r * \left(\frac{1}{(d(X_r, X_o) - d_o)^2} - \frac{1}{(d_m - d_o)^2} \right) & d \leq d_m \\ 0 & d > d_m \end{cases} \quad (2.1)$$

Onde $d(X_r, X_o)$ representa a distância euclidiana entre a embarcação a motor e o obstáculo, d_o é a distância mínima de segurança para evitar uma colisão, d_m representa

a distância a partir da qual os obstáculos começam a exercer influência no barco (raio da margem de segurança) e K_r representa uma constante associada ao campo potencial repulsivo. Para manter o barco no centro do rio, é utilizado uma força atrativa que o “puxa” para o centro, dada pela equação 2.2.

$$F_{at}(X_R) = K_a * d(X_R, X_C) \quad (2.2)$$

Onde $d(X_r, X_C)$ representa a distância entre o barco e o centro do rio, e K_a representa uma constante associada ao campo potencial atrativo. Para obter a direção a seguir pelo barco (θ) utiliza-se a equação 2.3.

$$\theta = \theta_{at} + \theta_{rep} = K_{at} * (D_E - D_D) + K_{rep} * d(X_R, X_O)^{-2} \quad (2.3)$$

Onde D_E e D_D representam a distância entre o barco e a margem esquerda e direita respetivamente. θ_{at} (equação 2.4) representa o ângulo da força atrativa.

$$\theta_{at} = K_{at} * (D_E - D_D) \quad (2.4)$$

Assim como θ_{rep} representa o ângulo da força repulsiva, com uma equação semelhante à equação 2.4, substituindo K_{at} por K_{rep} .

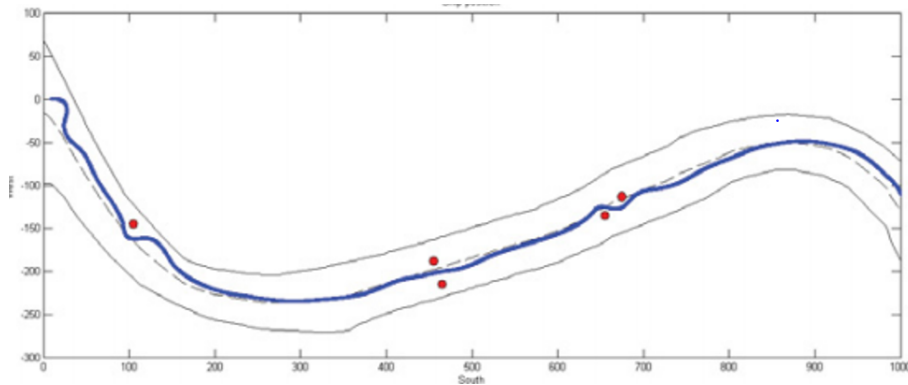


Figura 2.29: Resultado da simulação com base num rio real.
Imagem: [Hong e Arshad 2015].

Na simulação efetuada, verifica-se que o traço a azul representa o trajeto seguido pelo barco, o tracejado indica o centro do rio e os pontos a vermelho os diferentes obstáculos encontrados ao longo do caminho, como se observa na figura 2.29. Um ponto importante a notar neste projeto, é o facto de o algoritmo utilizado se tratar de um algoritmo híbrido, pois é responsável por evitar obstáculos e manter o barco no centro do rio.

No final, é possível concluir que o método apresentado permite que o barco se mantenha perto do centro do rio, evitando ao mesmo tempo os obstáculos estáticos com sucesso.

2.7.3 A Reactive COLREGs-Compliant Navigation Strategy for Autonomous Maritime Navigation

Neste projeto a estratégia é semelhante ao do trabalho [Lyu e Yin 2019], com o desenvolvimento de um algoritmo reativo para um barco, cumprindo algumas regras de navegação (ultrapassagem, cruzamento frontal e cruzamento) em cenários com obstáculos estáticos e móveis.

À semelhança de [Hong e Arshad 2015], é definida uma área (círculo) centrada no barco, com raio D_o , que representa a distância a partir da qual é necessário ter em conta o efeito de obstáculos (figura 2.30).

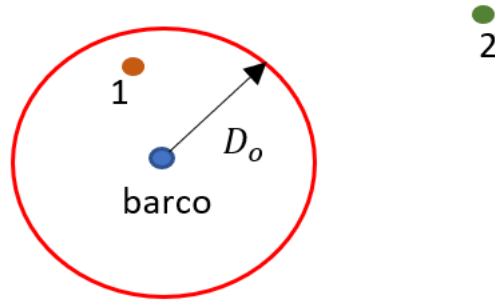


Figura 2.30: Área de efeito com um raio de D_o centrada no barco (ponto azul).

Na situação da figura 2.30, apenas o obstáculo n°1 (ponto laranja) seria tido em conta para o planeamento do trajeto. O obstáculo n°2 (ponto verde) não teria influência, pois encontra-se fora da área de efeito definida pelo círculo vermelho. Os potenciais atrativo e repulsivo são dados pelas equações 2.5 e 2.6 respetivamente.

$$U(q)_{atractivo} = \frac{1}{2} \cdot K_{at} \cdot d_G^2(q, q_{chegada}) \quad (2.5)$$

Na equação 2.5, $d_G^2(q, q_{chegada})$ corresponde ao quadrado da distância euclidiana entre a posição do barco (q) e o ponto de chegada ($q_{chegada}$) e K_{at} representa uma constante atrativa.

$$U(q)_{rep} = \begin{cases} \frac{1}{2} \cdot K_{ob} \cdot \frac{1}{d(q, q_{obs})} & d \leq D_o \\ 0 & d \geq D_o \end{cases} \quad (2.6)$$

Na equação 2.6, $d(q, q_{obs})$ reflete o quadrado da distância euclidiana entre a posição do barco (q) e o obstáculo (q_{obs}), K_{ob} uma constante repulsiva e D_o representa o raio do círculo de segurança.

Relativamente à constante K_{ob} , esta varia de acordo com o tipo de obstáculo (estático, móveis ou linhas costeiras), podendo ser ajustado dependendo se se quer um desvio com mais ou menos margem de segurança.

Em relação às linhas costeiras, a abordagem utilizada define obstáculos virtuais nos limites da mesma e não como um “bloco” de obstáculos, de forma a evitar que haja uma acumulação de potencial negativo muito elevada (K_{margem}), sobrepondo-se a qualquer força atrativa existente sobre o barco.

De modo a respeitar as regras de navegação, a estratégia utilizada passa por atribuir obstáculos virtuais às zonas proibidas, formando uma linha de obstáculos sem espaçamento entre eles, para evitar o problema dos mínimos locais (onde potencial atrativo + potencial repulsivo = 0).

Após as simulações, concluiu-se que o método apresentado representa uma solução capaz de lidar com obstáculos estáticos e móveis simultaneamente.

Foram encontrados alguns cenários onde o barco ficava preso num mínimo local, e ainda foram sentidas algumas dificuldades no facto dos obstáculos virtuais criados não garantirem o impedimento do barco seguir nessa trajetória, consequência do ajuste dos parâmetros $K_{ob}, K_{at}, K_{margem}$.

2.7.4 Toward an Autonomous Sailing Boat

O projeto [Plumet, Pêtrès, Gas e Ieng 2014] teve como objetivo desenvolver um algoritmo reativo para um veleiro autónomo. Como foi visto em [Lyu e Yin 2019] e [Naeem, Henrique e Hu 2016], os algoritmos e as estratégias propostas, foram feitos para um barco com motor, eliminando desta forma quaisquer problemas associados ao vento e correntes marítimas. Embora neste artigo não sejam abordadas as regras de navegação e considerados apenas obstáculos estáticos, é abordada a questão da influência do vento na estratégia de navegação.

Tendo em consideração a componente do vento no âmbito dos campos potenciais artificiais, o potencial resultante (U) é dado pela equação 2.7.

$$U = U_{chegada} + \sum_{i=0}^n U_{rep}^{(i)} + U_{vento} \quad (2.7)$$

Onde $U_{chegada}$ corresponde ao potencial atrativo associado ao ponto de chegada, dado pela equação 2.8.

$$U_{chegada} = G_{chegada} \cdot d_{chegada} \quad (2.8)$$

Onde $G_{chegada}$ representa uma constante associada ao campo potencial atrativo, $d_{chegada}$ indica a distância euclidiana entre a posição do veleiro e o ponto de chegada, e $\sum_{i=0}^n U_{rep}^{(i)}$ corresponde ao somatório dos potenciais repulsivos associados a cada um dos n obstáculos existentes, potencial este definido pela equação 2.9.

$$U_o^{(j)} = \begin{cases} G_{rep} \cdot \left(\frac{1}{d_o^{(j)}} - \frac{1}{d_{inf}} \right) & d_o^{(j)} \leq d_{inf} \\ 0 & d_o^{(j)} > d_{inf} \end{cases} \quad (2.9)$$

À semelhança do potencial atrativo, existe também um fator de peso constante repulsivo G_{rep} , d_o representa a distância euclidiana entre o veleiro e o obstáculo e d_{inf} refere-se à distância de influência do obstáculo. Por último, temos um potencial associado ao vento U_{vento} , definido pela equação 2.10.

$$U_{vento} = \begin{cases} P_{znn}, & \phi \in [zona proibida] \\ P_h + G_v \cdot \left(\frac{V - V_{max}}{V_{max}} \right) & \phi \notin [zona proibida] \end{cases} \quad (2.10)$$

No primeiro ramo, P_{znn} representa o potencial associado às zonas não navegáveis (representadas a cinzento na figura 2.31). No segundo ramo, P_h trata-se de uma variável, utilizada para controlar o número de manobras efetuadas (virar o bordo ou cambar). Caso ϕ (ângulo que corresponde à próxima direção a seguir) e φ (direção atual) estejam no mesmo semiplano, a variável P_h terá um valor nulo, caso contrário terá um valor diferente de 0. V representa a velocidade atual do veleiro, V_{max} é a velocidade máxima expectável (segundo o diagrama polar).

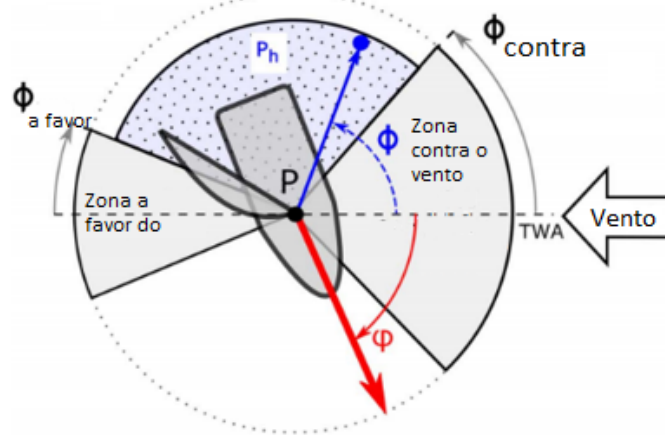


Figura 2.31: Esquema das zonas de vento a favor e contra. Nas zonas com pontos a azul P_h tem um valor diferente de 0.

Imagem: [Plumet, Pêtrès, Gas e Ieng 2014].

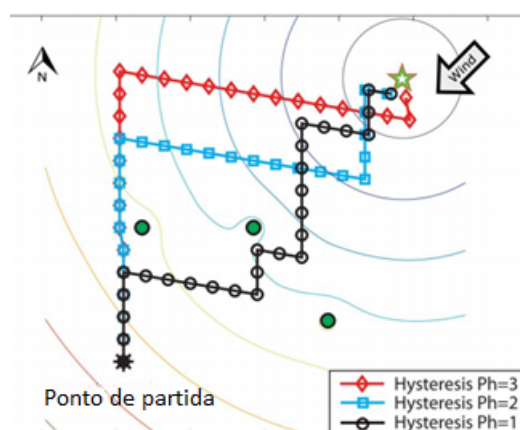


Figura 2.32: Simulações com diferentes parâmetros P_h . Imagem: [Plumet, Pêtrès, Gas e Ieng 2014].

As simulações foram realizadas num cenário com vento constante e obstáculos (pontos verdes), fazendo variar o valor de P_h (figura 2.32). Conclui-se que, quanto maior o valor de P_h , menos manobras são efetuadas. Foi feito ainda um teste em condições reais com os resultados ilustrados figura 2.33.



Figura 2.33: Simulação em condições reais com o trajeto seguido a amarelo. Imagem: [Plumet, Pêtrès, Gas e Ieng 2014].

O veleiro foi inicialmente conduzido do ponto A até ao ponto B por controlo remoto, sendo posteriormente definido o primeiro ponto de chegada (Wp1). Ao chegar ao ponto Wp1, definiu-se outro ponto de chegada, Wp2. A linha a amarelo indica o percurso efetuado pelo veleiro, verificando-se que a simulação foi realizada com sucesso embora sem a presença de qualquer obstáculo (figura 2.33).

2.7.5 Reactive path planning for autonomous sailboat

Projeto [Petres, Romero-Ramirez e Plumet 2011] desenvolvido por três dos autores de [Plumet, Pêtrès, Gas e Ieng 2014], sendo possível observar um raciocínio muito semelhante, de notar apenas algumas diferenças nas equações que descrevem os diferentes potenciais, sendo ainda abordado de que forma as forças aerodinâmicas e hidrodinâmicas influenciam os resultados. Tendo como objetivo o desenvolvimento de um algoritmo para um veleiro, torna-se necessário abordar de que forma as forças aerodinâmicas e hidrodinâmicas que atuam nas velas e na quilha respectivamente, influenciam a navegação, de modo a preparar o algoritmo o mais próximo das condições reais. A força que atua ao nível das velas (F_V) e na quilha (F_Q) são descritas pela equação 2.11.

$$\begin{cases} F_V = \frac{1}{2} \cdot \rho_1 \cdot S_1 \cdot V^2 \cdot C_d \\ F_Q = \frac{1}{2} \cdot \rho_2 \cdot S_2 \cdot V^2 \cdot C_l \end{cases} \quad (2.11)$$

A densidade do ar, representada por ρ_1 ou da água representada por ρ_2 , S_1 representa a área da vela e S_2 a área da quilha, V a velocidade do vento aparente sentido na vela (ou velocidade relativa da água para a quilha), e $C_{d/l}$ os coeficientes de arrasto e levantamento.

Para descrever o potencial atrativo associado ao ponto de chegada, é utilizada a equação 2.12.

$$P_{atr} = G_{atr} \cdot dist(P, P_{chegada}) \quad (2.12)$$

Com G_{atr} a corresponder a um fator de peso constante e $dist(P, P_{chegada})$ à distância euclidiana entre a posição do veleiro e o ponto de chegada. A equação 2.13 traduz o potencial repulsivo associado aos obstáculos.

$$P_{rep} = \frac{G_{rep}}{dist(P, P_{obs})} \quad (2.13)$$

Onde G_{rep} representa um fator de peso constante e $dist(P, P_{obs})$ a distância euclidiana entre as posições do veleiro e do obstáculo.

Relativamente ao potencial associado ao vento o raciocínio é igual ao de [Plumet, Pêtrès, Gas e Ieng 2014], assim como para evitar que o veleiro realize manobras (como cambiar ou virar o bordo) com muita frequência. Foram simulados três ambientes de navegação.

1º - Ambiente com vento constante e obstáculos

Verifica-se que o veleiro chega ao ponto de chegada definido (estrela vermelha), evitando os obstáculos, como ilustrado na figura 2.34. São apresentados dois caminhos obtidos, um troço azul e um troço preto, com diferentes direções de vento. Concluindo que, para este tipo de simulações o algoritmo apresenta resultados corretos.

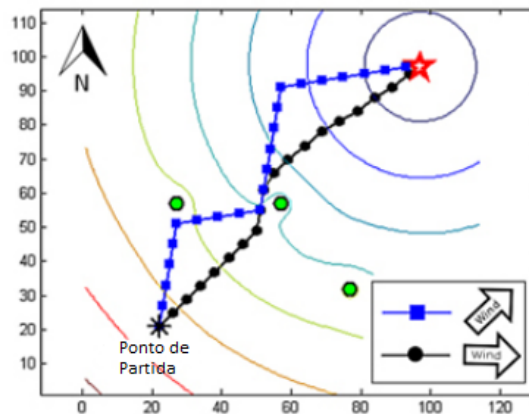


Figura 2.34: Trajetórias obtidas por simulação com ventos constantes vindos de Oeste (linha com bola a preto) e vindos do Sudoeste (linha com quadrado a azul). Imagem: [Petres, Romero-Ramirez e Plumet 2011].

2º - Ambiente com ventos dinâmicos e obstáculos

Nesta simulação, o algoritmo é testado em ambientes de navegação marítima com ventos mais realistas, como se pode observar pelas setas azuis na figura 2.35. Mais uma vez, verifica-se que chega ao destino com sucesso, evitando os obstáculos (pontos verdes).

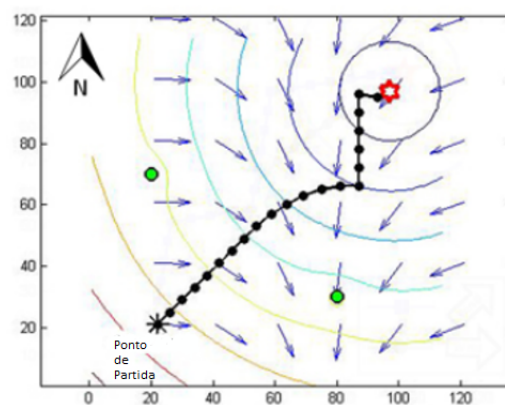


Figura 2.35: Trajetória obtida por simulação com ventos dinâmicos. Imagem: [Petres, Romero-Ramirez e Plumet 2011].

3º - Utilização de um modelo do veleiro mais realista

Para esta simulação, são tidos em conta as dinâmicas e características do próprio veleiro, como a massa, o torque, velocidade angular etc. sendo feita uma comparação entre os resultados simulados com e sem o modelo do veleiro mais realista na figura 2.36.

2.7. TRABALHOS RELACIONADOS COM O ALGORITMO BASEADO EM CAMPOS POTENCIAIS ARTIFICIAIS

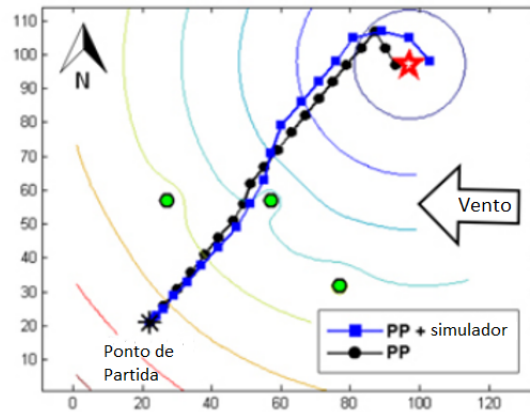


Figura 2.36: Trajetórias obtidas por simulação utilizando apenas o algoritmo (linha com bola a preto) e o algoritmo com o simulador das forças aerodinâmicas e hidrodinâmicas (linha com quadrado a azul). Imagem de: [Petres, Romero-Ramirez e Plumet 2011].

Em relação ao resultado da simulação, verifica-se que, tanto utilizando o algoritmo com um modelo mais realista do veleiro (traço azul) como o modelo sem as dinâmicas incluídas (traço preto), ambos chegam ao destino com sucesso evitando os obstáculos, a diferença reside principalmente na parte final do trajeto, onde é necessário realizar uma manobra em direção ao ponto de chegada, onde se verifica que o traço a preto realiza tal manobra, mas no trajeto a traço a azul tal não acontece por não ser uma manobra de imediata execução quando se têm em conta as dinâmicas do veleiro.

2.7.6 Conclusões

Através da tabela 2.2 é possível concluir que, dos trabalhos apresentados, nenhum deles cumpre na totalidade os objetivos propostos para esta dissertação.

Tabela 2.2: Comparação dos trabalhos realizados relacionados

Referência	Robô	Obst.* Estáticos/ Móveis	Testes Sim.**/ Reais	COLREGS
[Lyu e Yin 2019]	Barco	Sim/Sim	Sim/Não	Sim
[Hong e Arshad 2015]	Barco	Sim/Não	Sim/Não	Não
[Naeem, Henrique e Hu 2016]	Barco	Sim/Sim	Sim/Não	Sim
[Plumet, Pêtrès, Gas e Ieng 2014]	Veleiro	Sim/Não	Sim/Sim	Não
[Petres, Romero-Ramirez e Plumet 2011]	Veleiro	Sim/Não	Sim/Não	Não

* Obstáculos ** Simulações

Nos casos de [Lyu e Yin 2019] e [Naeem, Henrique e Hu 2016], embora a estratégia desenvolvida tenha em conta obstáculos estáticos, móveis e algumas regras de navegação

(COLREGS), foi utilizada uma embarcação a motor e não um veleiro, pelo que toda a questão da navegação dependente do vento deixa de ser relevante.

No caso de [Hong e Arshad 2015], pela mesma razão referida anteriormente.

Em trabalhos como [Plumet, Pêtrès, Gas e Ieng 2014] e [Petres, Romero-Ramirez e Plumet 2011] são utilizados veleiros, onde são definidos campos potenciais associados ao vento. Embora sejam apresentadas simulações do planeamento de rota para cenários com obstáculos estáticos presentes e resultados de simulação em ambiente real, é fornecida pouca informação sobre a estratégia adotada para obter tais resultados.

ALGORITMO DE CAMPOS POTENCIAIS ARTIFICIAIS

Um dos principais desafios na realização desta dissertação passa por decidir qual o algoritmo que melhor se enquadra nas condições de navegação apresentadas, nomeadamente, o facto de se tratar de um **ambiente dinâmico**, onde tanto as condições como o ambiente de navegação estarão em constante mudança. A abordagem utilizada passa por desenvolver um algoritmo responsável pelo planeamento de rota e evitar obstáculos **em tempo real**, com base no conceito de campos potenciais artificiais.

Proposto por *Oussama Khatib* em 1985 [Khatib 1985], tem como característica o facto de a execução e o planeamento da trajetória ocorrerem simultaneamente. Neste tipo de algoritmo, a ideia principal passa por replicar os campos potenciais (como por exemplo elétricos, magnéticos ou gravitacionais) e as forças que eles exercem noutras partículas que “entrem” na região do campo. Neste caso, o veleiro é reduzido a uma partícula que se move num campo potencial, campo este que é descrito por funções matemáticas reais e diferenciáveis.

O veleiro poderá estar sob o efeito de dois tipos de campos potenciais [Choset 2006] [Dolgov, Thrun, Montemerlo e Diebel 2010], um **potencial atrativo** (U_{atr}), normalmente associado a um ponto de chegada, pois estará a “atrair” o veleiro na sua direção, e associado a obstáculos estará um **potencial repulsivo**, (U_{rep}) que afastará o veleiro. O potencial resultante (U_x) em cada instante, é então dado pela soma do potencial atrativo e repulsivo (associado a cada um dos obstáculos), como indicada na equação 3.1.

$$U_x = U_{atr} + \sum_{i=0}^n U_{rep}^{(i)} \quad (3.1)$$

Uma possível representação das forças exercidas pelo campo potencial atrativo no ponto de chegada (setas verdes) e pelo campo potencial repulsivo (setas vermelhas) no obstáculo, obtendo o caminho a amarelo, é possível observar-se na figura 3.1.

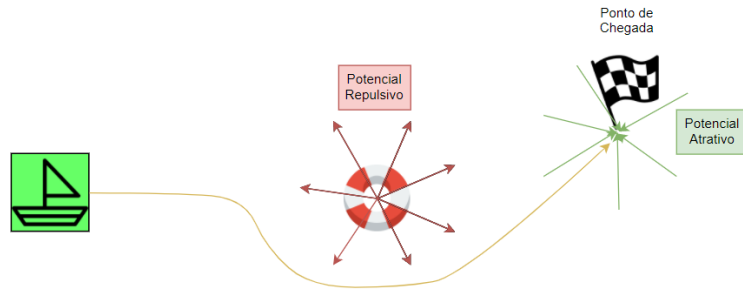


Figura 3.1: Campo potencial atrativo (verde) no ponto de chegada e repulsivo (vermelho) no obstáculo.

A direção de navegação a seguir pelo veleiro [Khatib 1985] em cada instante é dada por um vetor, que resulta do somatório de todas as forças (atrativas e repulsivas) em cada ponto. O deslocamento do veleiro é dado pela magnitude do vetor, e, sendo o espaço de estudo um ambiente de duas dimensões, o vetor de deslocamento poderá ser decomposto em duas componentes, latitude (x) e longitude (y).

Os campos potenciais serão descritos por funções $f(x,y)$, que se traduzem em superfícies no espaço. Para analisar a taxa de variação (derivada), esta terá que ser tratada para cada direção independentemente, utilizando as derivadas parciais com respeito a x (latitude) e a y (longitude).

Tratando-se de um operador diferencial, pode-se afirmar que as propriedades indicadas pelas equações 3.2, 3.3, 3.4 e 3.5 se verificam.

$$\frac{d(f + g)}{dx} = \frac{df}{dx} + \frac{dg}{dx} \quad (3.2)$$

$$\frac{d(f - g)}{dx} = \frac{df}{dx} - \frac{dg}{dx} \quad (3.3)$$

$$\frac{d(f * g)}{dx} = f * \frac{dg}{dx} + \frac{df}{dx} * g \quad (3.4)$$

$$\frac{d(f/g)}{dx} = [g * \frac{df}{dx} - f * \frac{dg}{dx}] / g^2 \quad (3.5)$$

Considere-se agora uma função multi-variável $f(x,y,...)$, onde o **gradiente**, denotado por ∇f , representa um vetor composto pelas derivadas parciais da função f .

$$\nabla f[x, y, \dots] = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \dots \end{bmatrix} \quad (3.6)$$

O gradiente é composto pelas derivadas parciais de uma função, ou seja, pela taxa de variação instantânea em cada uma das variáveis para cada ponto (equação 3.6).

Uma das propriedades do **gradiente** é o facto deste apontar na direção de **máxima variação**, como se pode verificar na figura 3.2, onde no ponto a vermelho, aponta na direção de máxima variação do campo potencial apresentado.

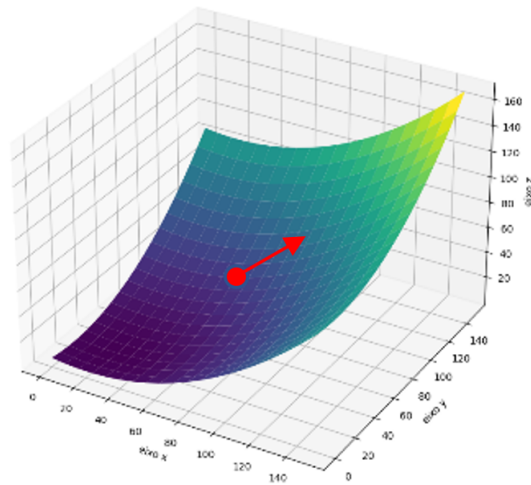


Figura 3.2: Exemplo do vetor gradiente no ponto a vermelho.

Considerando agora o cenário da figura 3.3, onde o ponto amarelo representa o ponto de chegada com veleiro na posição do ponto vermelho. O gradiente (seta a vermelho) aponta no sentido contrário do ponto de chegada, pois este representa o ponto onde o potencial total é zero, logo, a força que é exercida no veleiro terá de ser igual ao simétrico do gradiente (seta a verde), desta forma, obtém-se um vetor que aponta no sentido onde a variação é menor, ou seja, na direção de um mínimo do campo, neste caso, do ponto de chegada (ponto amarelo), logo: **Força** = $-\nabla f$.

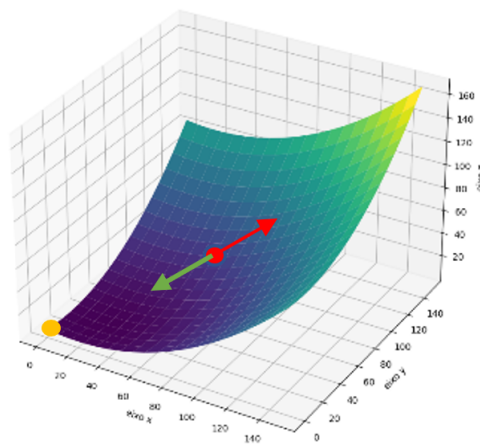


Figura 3.3: Exemplo do vetor gradiente no ponto a vermelho, o simétrico do gradiente (seta a verde) e o ponto de chegada (ponto amarelo).

3.1 Problema dos Mínimos Locais

Uma das vantagens dos algoritmos baseados em campos potenciais artificiais [Plumet, Saoud e M. D. Hua 2013] é o facto de serem relativamente simples de implementar, exigirem pouco poder computacional e funcionarem com obstáculos estáticos e móveis. Idealmente, os campos potenciais teriam apenas **um mínimo global**, que representaria o ponto de chegada, contudo, existem diversos cenários que originam **mínimos locais** [Park e Lee 2003] [Azmi e Ito 2020] (figura 3.4), ou seja, pontos onde a força atrativa e repulsiva se igualam.

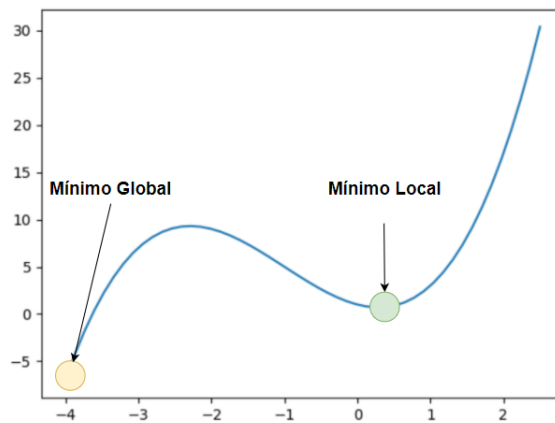


Figura 3.4: Ilustração de um mínimo global e local.

O veleiro é inicialmente atraído pelo campo potencial associado ao ponto de chegada, seguindo o trajeto indicado pela seta no cenário da figura 3.5. Quando o veleiro chega à zona de influência (zona delimitada por linhas a vermelho) do obstáculo (área a amarelo), este começa a exercer uma força repulsiva no veleiro com direção contrária à da força atrativa, igualando-se (ponto 1), originando um mínimo local, resultando numa paragem do veleiro, pois o somatório de forças é 0.

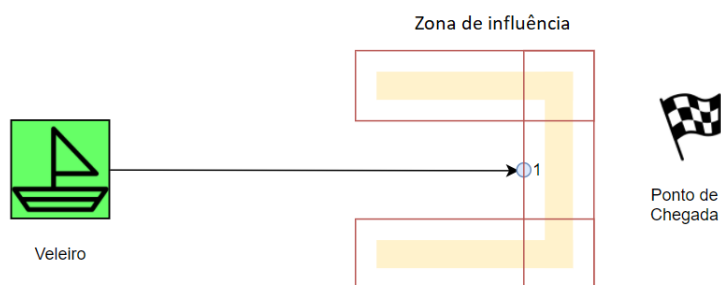


Figura 3.5: Cenário com um mínimo local devido à presença de um obstáculo côncavo.

Um outro exemplo onde este problema poderia existir, é o caso da passagem pelo meio de dois obstáculos muito próximos um do outro, como se encontra representado na figura 3.6. Para além dos cenários apresentados, existem inúmeros outros que poderão dar origem ao problema dos mínimos locais, sendo uma situação muito difícil de prever.



Figura 3.6: Cenário com um mínimo local devido à presença de dois obstáculos muito próximos.

Diminuindo a distância de influência do obstáculo, poderia ser uma possível solução para o cenário da figura 3.6.

Em [Matoui, Boussaid e Abdelkrim 2016], são propostas soluções para este problema, como por exemplo, associar o caminho encontrado pelos campos potenciais artificiais a uma função aleatória, que é utilizada sempre que o robô fica preso num mínimo local.

3.2 Funções dos campos potenciais

Nesta secção, serão apresentadas as funções utilizadas para descrever cada um dos campos potenciais utilizados.

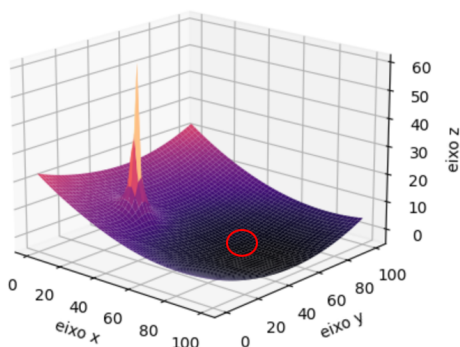


Figura 3.7: Campo potencial global. Ponto de chegada (mínimo global) representado pelo círculo a vermelho.

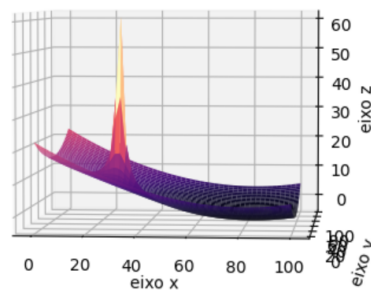


Figura 3.8: Vista lateral do campo potencial global.

Atendendo ao cenário da figura 3.1, onde o ponto de chegada representa um mínimo global (círculo vermelho), é possível observar o campo potencial global, resultante da

soma dos campos potenciais que são criados, na figura 3.7. Por outro lado, associado a um pico de potencial repulsivo temos um obstáculo, representado por uma boia.

3.2.1 Campo Potencial Atrativo

O campo potencial atrativo terá que ser definido de forma que:

- o gradiente “puxe” o veleiro para o ponto de chegada
- seja contínuo e diferenciável em todos os pontos do domínio
- seja positivo, $U(q) > 0 \forall q \neq q_{chegada}$, onde $U(q_{chegada}) = 0$.

Tendo em conta as condições indicadas, conclui-se que uma das opções, é o **campo potencial cônico**, sendo linear e dependente da distância euclidiana entre o veleiro e o ponto de chegada, descrito pela equação 3.7 e representado na figura 3.9.

$$U_{x,y} = \sqrt{(x - x_{chegada})^2 + (y - y_{chegada})^2} \quad (3.7)$$

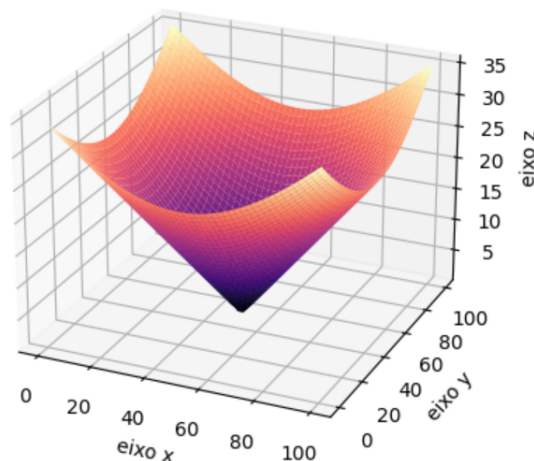


Figura 3.9: Campo potencial cônico.

Analisando o campo potencial da figura 3.9, verifica-se que no ponto de chegada ($q_{chegada}$: $x = 50$, $y = 50$) temos $U(q_{chegada}) = 0$.

Considerando agora o mesmo campo potencial mas num referencial de duas dimensões, e a respetiva força associada (simétrico do gradiente), obtém-se o resultado da figura 3.10.

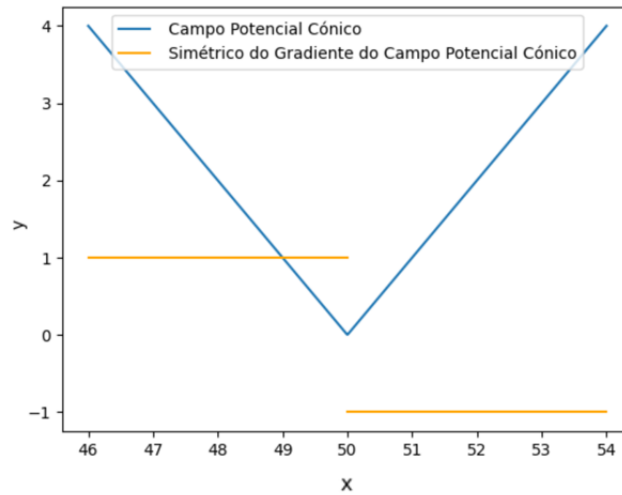


Figura 3.10: Campo potencial cónico num referencial de duas dimensões (azul) e o respetivo simétrico do gradiente (laranja).

Um dos problemas associado a um campo potencial cónico é o facto de o simétrico do gradiente no ponto de chegada ser descontínuo. Como se pode observar na figura 3.10, o gradiente no ponto $x=50$ não está definido, havendo uma descontinuidade, para os restantes pontos não existe qualquer problema e conclui-se que a força atrativa é igual em todos os pontos, com magnitude de um.

Outra opção que não apresenta qualquer tipo de descontinuidade no ponto de chegada é o campo parabólico, definido pela equação 3.8 e representado na figura 3.11 .

$$U_{x,y} = (x_{chegada} - x)^2 + (y_{chegada} - y)^2 \quad (3.8)$$

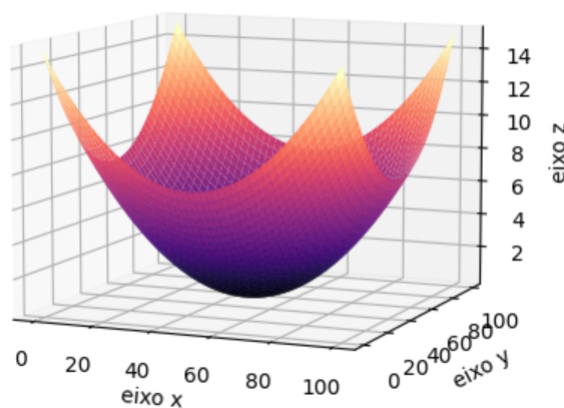


Figura 3.11: Campo potencial parabólico.

É apresentado seguidamente um “corte” lateral do campo potencial parabólico a duas dimensões, e o simétrico do gradiente associado na figura 3.12.

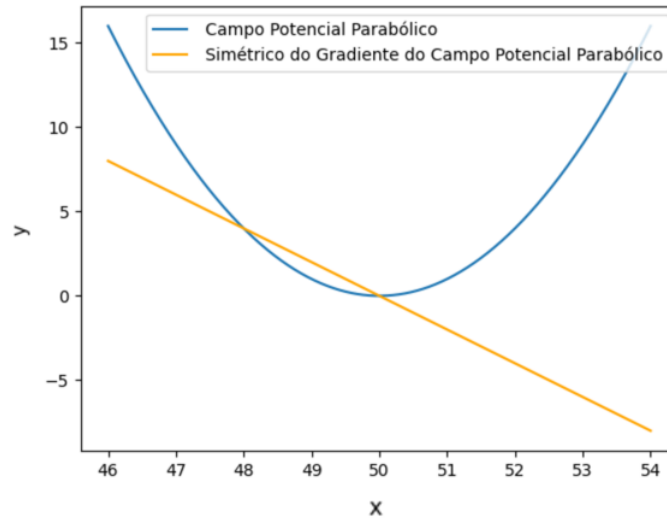


Figura 3.12: Campo potencial parabólico num referencial de duas dimensões (azul) e o respetivo simétrico do gradiente (laranja).

Verifica-se que o campo potencial possui apenas um mínimo global no ponto de chegada (neste caso $x = 50$), e que o simétrico do gradiente é contínuo em todo o domínio, não apresentando o problema de descontinuidade do campo potencial cónico. Por outro lado, este cresce linearmente com o aumento da distância ao ponto de chegada para infinito (figura 3.12). Para grandes distâncias ao ponto de chegada, obtém-se uma força atrativa muito elevada, correndo o risco de esta se sobrepor a qualquer outra força que seja exercida, sendo um problema para os casos em que o veleiro se encontra distante do ponto de chegada e encontra algum tipo de obstáculo.

De modo a tirar proveito das vantagens de cada um dos campos apresentados, o campo potencial atrativo fica então definido como uma junção de dois campos, onde, para evitar o problema das forças atrativas muito elevadas e a descontinuidade do simétrico do gradiente no ponto de chegada, para distâncias inferiores a δ utiliza-se o campo potencial parabólico, e para distâncias superiores ou iguais a δ utiliza-se o campo potencial cónico. No final, obtém-se o campo potencial definido pela equação 3.9.

$$U_{atrativo} = \begin{cases} \text{Campo_Potencial_Parabolico} & \text{dist_chegada} < \delta \\ \text{Campo_Potencial_Conico} & \text{dist_chegada} \geq \delta \end{cases} \quad (3.9)$$

Considerando a equação 3.10 e as definições de cada um dos campos potenciais, obtemos a equação 3.11 e o respetivo gradiente na equação 3.12.

$$dist_chegada = \sqrt{(x - x_{chegada})^2 + (y - y_{chegada})^2} \quad (3.10)$$

$$U_{atractivo} = \begin{cases} \epsilon * dist_chegada^2 & dist_chegada < \delta \\ \epsilon * dist_chegada & dist_chegada \geq \delta \end{cases} \quad (3.11)$$

Como em [Plumet, Pêtrès, Gas e Ieng 2014], adicionou-se uma constante, ϵ , associada ao potencial atrativo. Nesta parte, e de modo a simplificar o cálculo da força atrativa foi considerado $\epsilon = 0.5$.

$$F_{atractiva} = \begin{cases} F_{atractiva_Parabolico} & dist_chegada < \delta \\ F_{atractiva_Conico} & dist_chegada \geq \delta \end{cases} \quad (3.12)$$

$$F_{atractiva_Parabolico} = -\nabla U_{Parabolico}(x, y) = \begin{bmatrix} x_{chegada} - x \\ y_{chegada} - y \end{bmatrix} \quad (3.13)$$

$$F_{atractiva_Conico} = -\nabla U_{Conico}(x, y) = \begin{bmatrix} 0.5 * \frac{x_{chegada} - x}{\sqrt{(x - x_{chegada})^2 + (y - y_{chegada})^2}} \\ 0.5 * \frac{y_{chegada} - y}{\sqrt{(x - x_{chegada})^2 + (y - y_{chegada})^2}} \end{bmatrix} \quad (3.14)$$

Um corte lateral do campo potencial atrativo final, encontra-se ilustrado na figura 3.13, e o respetivo gradiente na figura 3.14.

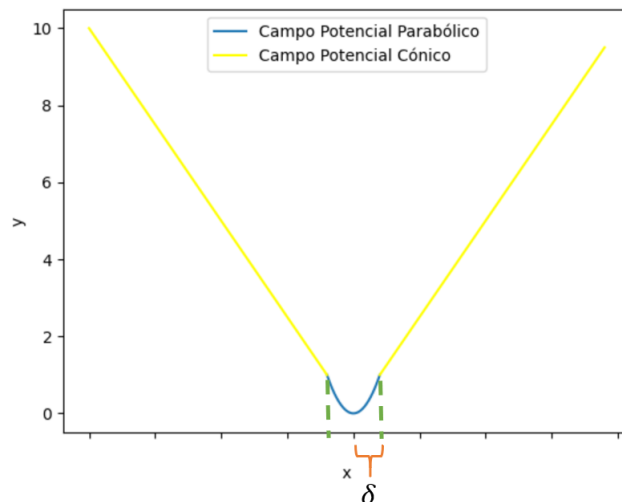


Figura 3.13: Campo potencial atrativo

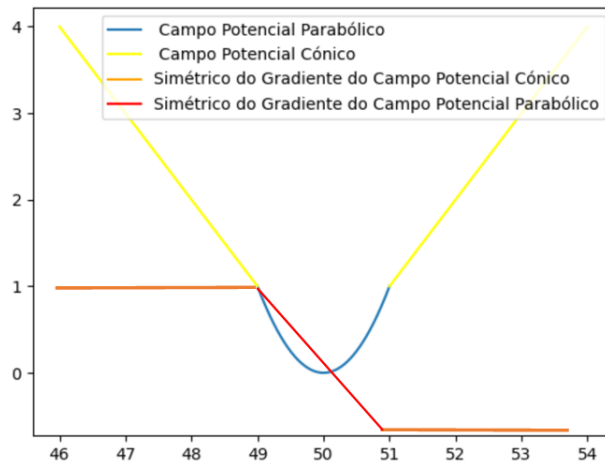


Figura 3.14: Campo potencial atrativo e o gradiente associado

Analisando o gradiente na figura 3.14 verifica-se que, para uma distância_chegada (distância do veleiro ao ponto de chegada) $\in [\text{ponto_chegada}+\delta, \text{ponto_chegada}-\delta]$ o campo potencial utilizado é o parabólico e a força atrativa decresce linearmente com distância ao ponto de chegada, sendo zero nesse mesmo ponto.

Por outro lado, se distância_chegada $\notin [\text{ponto_chegada}+\delta, \text{ponto_chegada}-\delta]$ o campo potencial a utilizar é o cônico e a força atrativa é constante e com magnitude de um.

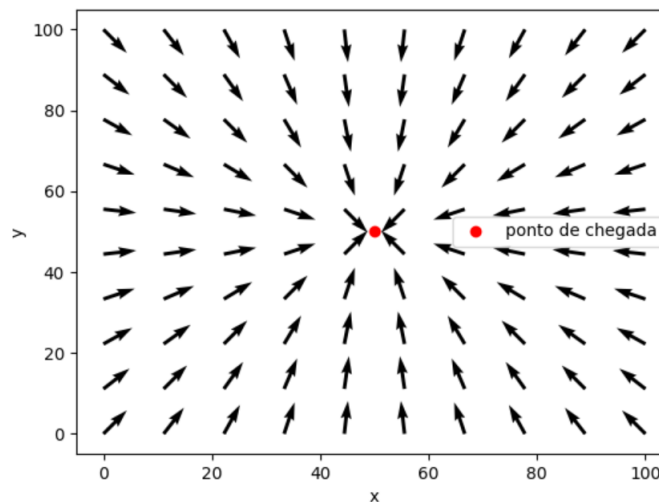


Figura 3.15: Gradiente do campo potencial atrativo

As forças existentes em cada ponto do espaço definido, assim como a sua direção, dadas pelo gradiente do campo potencial atrativo, encontram-se representadas na figura 3.15. Neste caso, para distâncias superiores ou iguais a um determinado δ , a magnitude da força que o campo exerce no veleiro é igual em todos os pontos, apontando sempre para o ponto de chegada, neste caso, de coordenadas $x=50$ e $y= 50$.

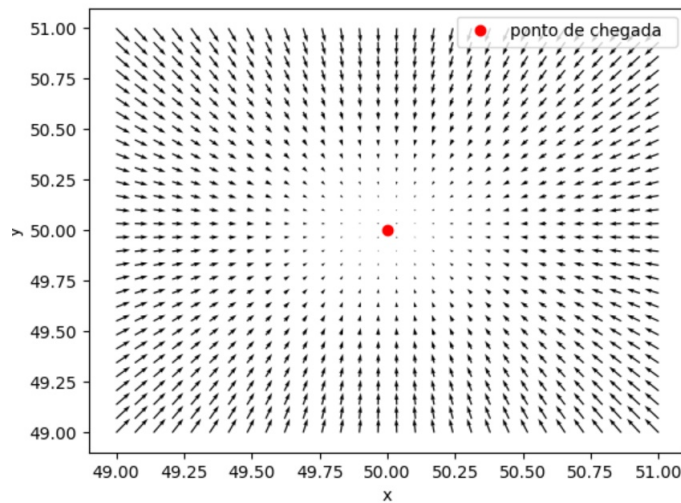


Figura 3.16: Gradiente parabólico associado ao campo potencial atrativo para distâncias inferiores a δ .

Para distâncias inferiores a δ ($\delta = 1$ neste caso) obtêm-se as forças representadas na figura 3.16, onde se verifica uma diminuição linear com a diminuição da distância ao ponto de chegada, apontando sempre para o destino pretendido como esperado.

3.2.2 Campo Potencial Repulsivo

No caso do campo potencial repulsivo este terá que ser definido de forma a que:

- o gradiente repele o veleiro dos obstáculos
- raio de influência seja limitado
- evite colisões a todo o custo.

Contrariamente ao campo potencial atrativo, este terá que exercer uma força que “afaste” o veleiro dos obstáculos [Khatib 1985] [Azmi e Ito 2020], que será mais acentuada quanto mais próximo o veleiro estiver do obstáculo, tendendo para infinito, querendo isto dizer que dependerá da distância do veleiro ao obstáculo. O raio de influência é um fator importante a ter em conta, pois não queremos que um obstáculo que não esteja nas proximidades do veleiro tenha qualquer tipo de influência no trajeto. Numa primeira abordagem, começou-se por definir o campo potencial repulsivo como indicado na equação 3.15.

$$U_{repulsivo} = \begin{cases} \frac{1}{2} * \eta * \left(\frac{1}{dist(veleiro,obstaculo)}\right)^2 & dist(veleiro,obstaculo) \leq \rho \\ 0 & dist(veleiro,obstaculo) > \rho \end{cases} \quad (3.15)$$

Considerando a equação 3.16.

$$dist(veleiro, obstaculo) = \sqrt{(x - x_{obstaculo})^2 + (y - y_{obstaculo})^2} \quad (3.16)$$

Nesta primeira definição, e à semelhança do campo potencial atrativo, η representa uma constante positiva que permite escalar o campo potencial e ρ representa distância a partir da qual o obstáculo exerce algum tipo de influência.

O único problema na definição do campo potencial repulsivo é o facto de não haver uma transição suave entre os diferentes ramos que definem o campo [Azmi e Ito 2020]. Para resolver esse problema, subtrai-se o valor do campo na zona de transição ($\frac{1}{\rho}$) aos ramos que define o campo, como indicado na equação 3.17 e o respetivo gradiente na equação 3.18.

$$U_{repulsivo} = \begin{cases} \frac{1}{2} * \eta * \left(\frac{1}{dist(veleiro, obstaculo)} - \frac{1}{\rho} \right)^2 & dist(veleiro, obstaculo) \leq \rho \\ 0 & dist(veleiro, obstaculo) > \rho \end{cases} \quad (3.17)$$

$$F_{repulsiva}(x, y) = -\nabla U_{repulsivo}(x, y) = \begin{bmatrix} \eta * \left(\frac{1}{dist(veleiro, obstaculo)} - \frac{1}{\rho} \right) * \left(\frac{1}{dist(veleiro, obstaculo)^2} \right) * \frac{x - x_{obst}}{dist(veleiro, obstaculo)} \\ \eta * \left(\frac{1}{dist(veleiro, obstaculo)} - \frac{1}{\rho} \right) * \left(\frac{1}{dist(veleiro, obstaculo)^2} \right) * \frac{y - y_{obst}}{dist(veleiro, obstaculo)} \end{bmatrix} \quad (3.18)$$

Desta forma, obtém-se um campo potencial repulsivo que tem o valor de zero para uma distância ρ ao obstáculo, e que tende para infinito quanto mais próximo o veleiro estiver.

É possível observar o campo potencial gerado por um obstáculo na posição $x=30, y=30$ na figura 3.17.

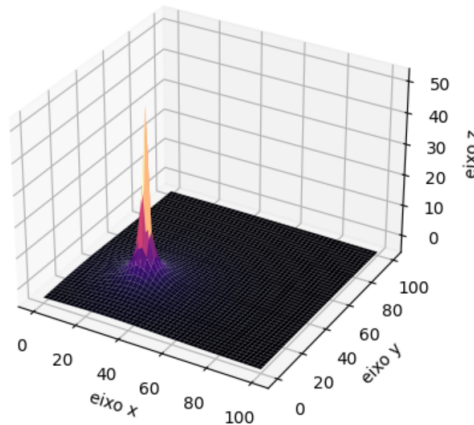


Figura 3.17: Campo potencial repulsivo gerado por um obstáculo.

O campo potencial repulsivo num gráfico de duas dimensões, assim como o simétrico do gradiente associado, pode ser observado na figura 3.18. Note-se que foi considerada uma distância de influência (ρ) igual a 1 e um fator de escalamento (η) igual a 2.

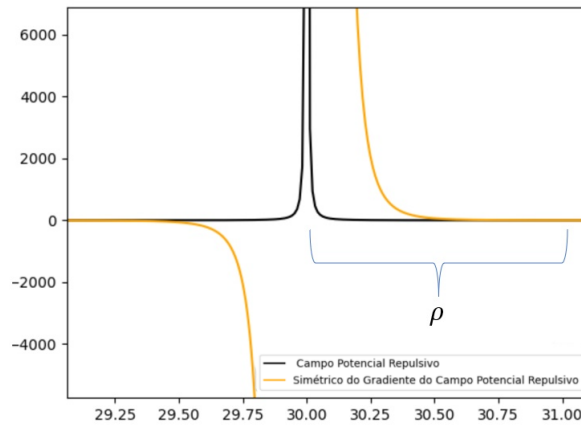


Figura 3.18: Campo potencial repulsivo gerado por um obstáculo e o respetivo simétrico do gradiente num referencial a duas dimensões.

Verifica-se que a força (simétrico do gradiente) tem o valor de 0 para distâncias superiores a 1 (valor atribuído a ρ) e que, quanto mais próximo da posição do obstáculo ($x=30$) maior é a força repulsiva exercida tendendo para infinito (figura 3.18). Foi igualmente feita uma representação dos vetores da força repulsiva na figura 3.19, onde se verifica que as forças apontam para fora do obstáculo e exercem influência até à distância estabelecida (neste caso $\rho = 10$).

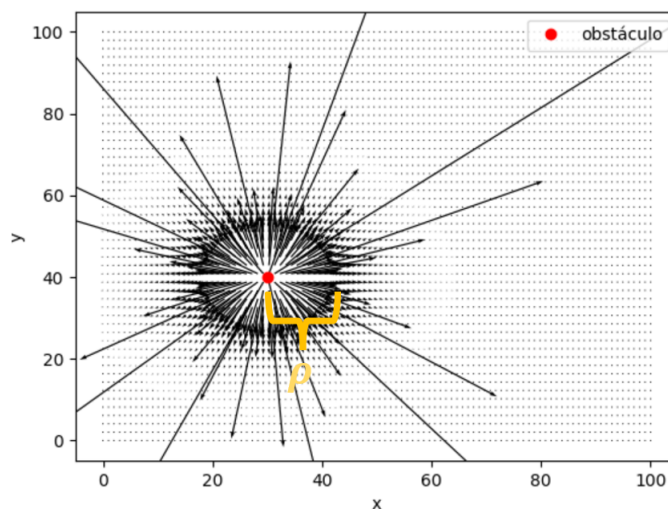


Figura 3.19: Simétrico do gradiente de um campo potencial repulsivo.

3.2.3 Campo Potencial do Vento

O vento é um dos principais intervenientes na navegação de um veleiro, para além de permitir o seu deslocamento, cria as ZNN.

Durante o processo do desenvolvimento do potencial associado ao vento, foram encontrados vários problemas, sendo necessárias diversas estratégias até se chegar à solução final. Será feita uma breve passagem pelas diferentes abordagens e o porquê de terem falhado.

Como foi já abordado, existem diversas posições de navegação que variam de acordo com o ângulo que a vela faz com o vento, permitindo atingir diferentes velocidades para as cada uma das diferentes posições. Os **diagramas polares** [N. Website 2018] são instrumentos que permitem prever o comportamento de um veleiro para diferentes ângulos e velocidades de vento. De notar que as velocidades que o veleiro atinge dependem de inúmeras características, como o tipo de veleiro, tipo de casco, material, tamanho das velas, tipo de velas etc, querendo isto dizer, que o diagrama polar é específico e único para cada tipo de veleiro.

O diagrama polar do veleiro utilizado em [Petres, Romero-Ramirez e Plumet 2011], encontra-se representado na figura 3.20. Cada curva colorida representa a velocidade¹ expectável do veleiro para diferentes velocidades de vento em diferentes ângulos. Por exemplo, com um vento de 5 nós (curva rosa) e com o veleiro a 120° relativamente à direção do vento, seria de esperar que o veleiro navegasse a uma velocidade de 2 nós (para mais informações sobre diagramas polares: [N. Website 2018]).

Para ângulos entre os 315° e os 45° (zona a cinzento), a velocidade do veleiro é 0 por se tratar de uma ZNN.

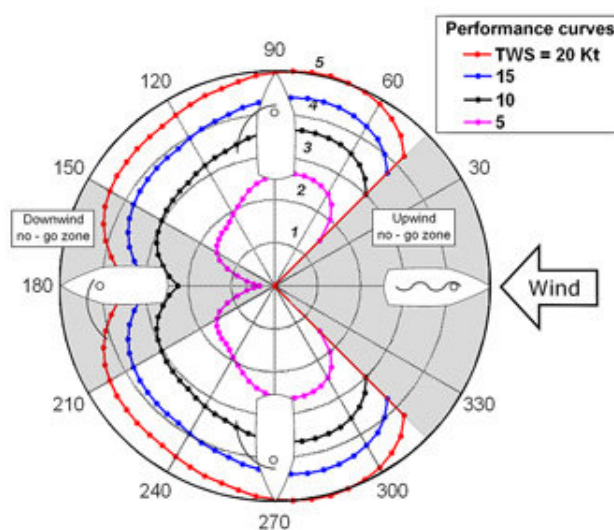


Figura 3.20: Exemplo de Diagrama polar. Referência:[Petres, Romero-Ramirez e Plumet 2011]

¹Apresentada com a unidade **nós**, onde: 1 nó = 1.8520km/h

No caso desta dissertação, foi admitido um diagrama polar ideal (figura 3.21), querendo isto dizer que, para uma **velocidade de vento** - V o veleiro navegará a uma velocidade constante V , limitada pelo vento, para qualquer que seja o ângulo em que o vento incida. Cada curva representa então a velocidade do veleiro para uma determinada velocidade de vento num determinado ângulo, que será não nula para todos os ângulos exceto na ZNN (região a vermelho).

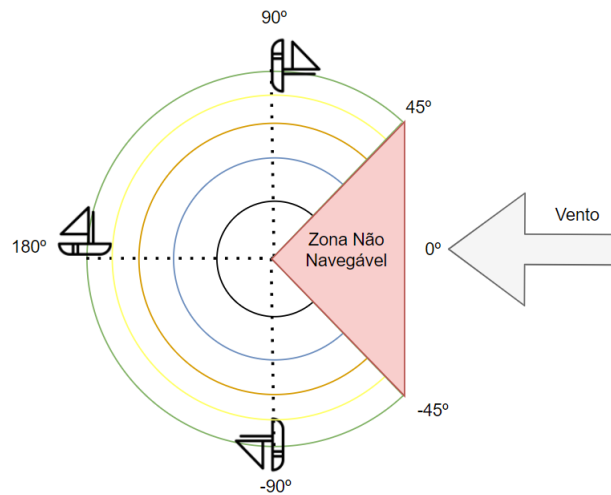


Figura 3.21: Diagrama polar utilizado.

Tendo em conta o diagrama polar referido, a estratégia a utilizar passa por atribuir um potencial repulsivo constante e elevado à ZNN, garantindo que o veleiro não navega dentro dessa mesma zona.

Para a definição dos ângulos foi utilizado a função trigonométrica **arco tangente 2** ou **atan2** [Ukil, Shah e Deck 2011], representada na figura 3.22.

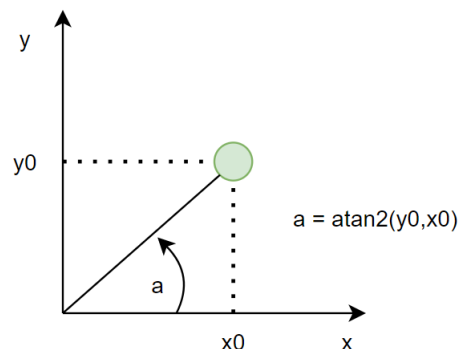


Figura 3.22: Definição gráfica do atan2.

Na figura 3.23 encontra-se uma representação do **atan2** em coordenadas polares.

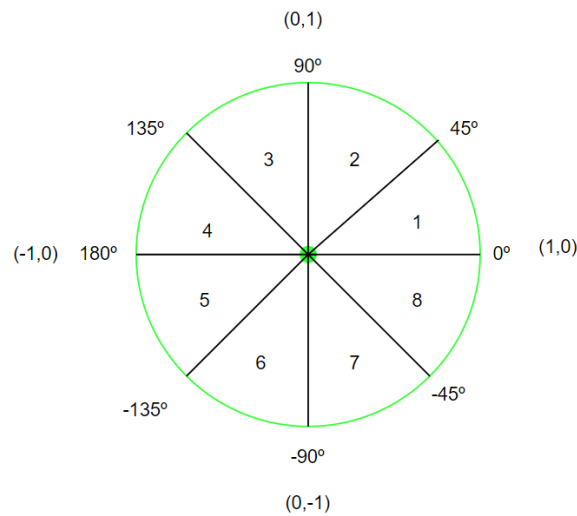


Figura 3.23: Atan2 em coordenadas polares.

Considerando o vento com uma direção de 0° , facilmente se calculam os limites da ZNN (limite inferior = direção do vento - 45° ; limite superior = direção do vento + 45°). Tendo em conta a definição de ângulos adotada, é necessário um cuidado especial para **ventos com direções pertencentes ao intervalo $]135^\circ, 180^\circ[$ U $]-180^\circ, -135^\circ[$** (setores 4 e 5 da figura 3.23), pois os limites da ZNN são calculados de maneira diferente. Com a utilização do atan2 e sabendo o próximo ponto de chegada do veleiro (latitude e longitude), facilmente se calcula o ângulo de direção pretendido, conseguindo avaliar se este se encontra dentro da ZNN ou não. De destacar, que o ponto onde o veleiro se encontra será sempre considerado como a origem do referencial, sendo feita uma translação da posição do veleiro e do próximo ponto de chegada para o ponto de coordenadas latitude = 0 e longitude = 0 do referencial, como se pode observar na figura 3.24.

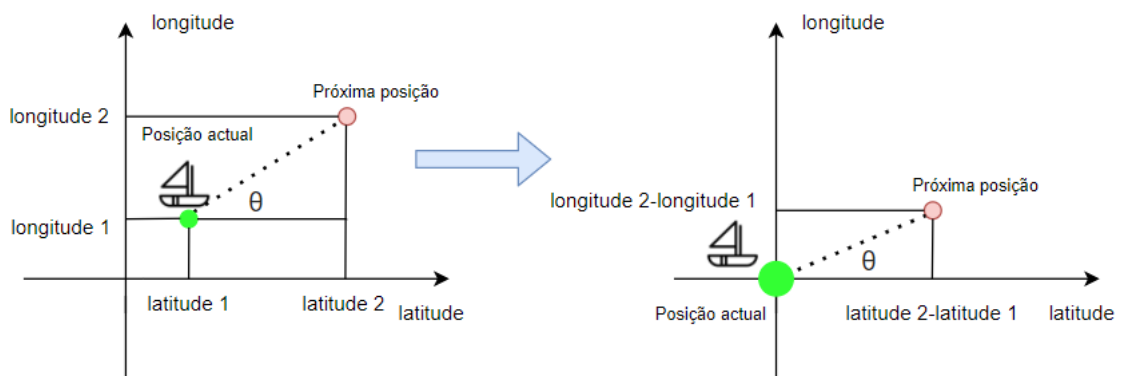


Figura 3.24: Cálculo do ângulo da direção a seguir.

O facto de se assumir um diagrama polar ideal, significa que apenas a ZNN terá um potencial repulsivo associado. Este potencial terá que exercer uma força no veleiro de

modo a impedi-lo de entrar nesta zona, garantindo que este boline no limite da mesma até que uma mudança de bordo seja possível de realizar.

Numa primeira abordagem optou-se por associar um campo potencial cónico (figura 3.9), onde o veleiro estaria sempre no centro do campo, mas após uma observação mais atenta do gráfico do gradiente obtido (figura 3.25), concluiu-se que não iria funcionar pois qualquer que fosse o ângulo do veleiro em relação à ZNN, a força repulsiva exercida pelo vento seria sempre no sentido contrário à força atrativa, pelo que o veleiro não se deslocaria.

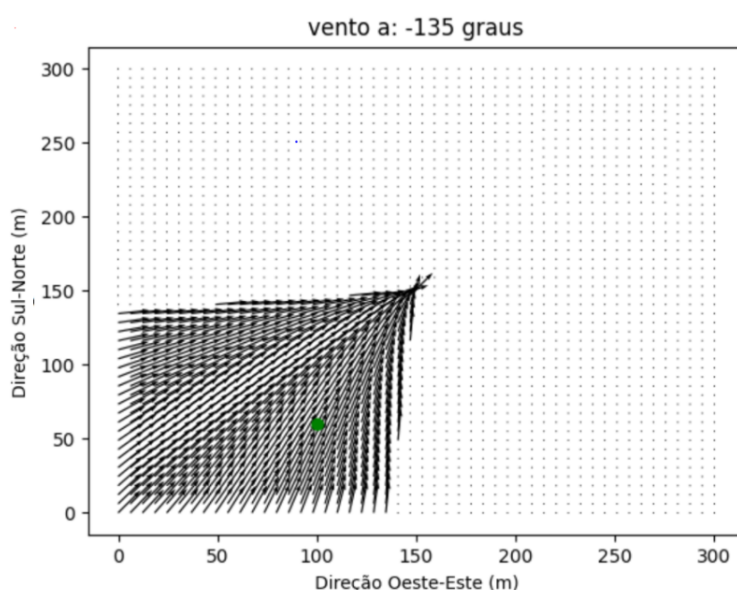


Figura 3.25: Gradiente repulsivo para um campo potencial cónico associado à ZNN.

Concluindo que o problema estaria na orientação da força repulsiva, optou-se por definir a direção do gradiente dentro da ZNN, em vez de definir o campo potencial. Começou-se por delinear um gradiente que apontasse para fora da ZNN, onde, na parte superior da ZNN o gradiente aponta para a direção do vento + 90° e na parte inferior para a direção do vento - 90°. como ilustrado na figura 3.26. No caso particular da figura 3.26 a parte superior aponta para 90° e a parte inferior para -90°, pois o ângulo do vento encontra-se nos 0°.

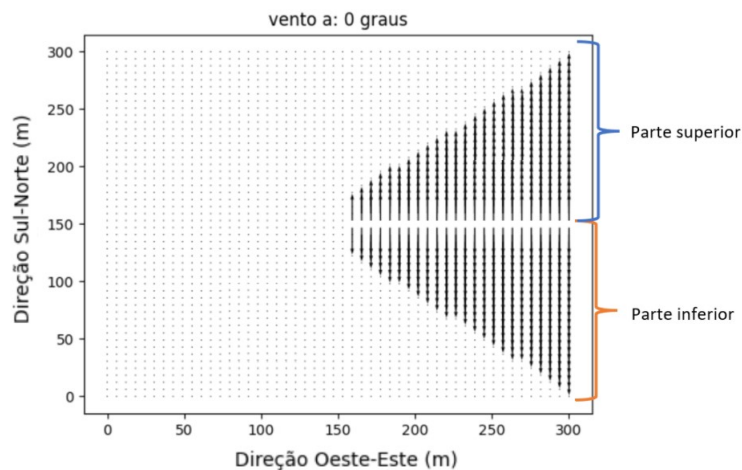


Figura 3.26: Gradiente repulsivo associado à ZNN e as diferentes partes consideradas.

O primeiro problema surge quando o ângulo entre o veleiro e o ponto de chegada tem um valor muito próximo do ângulo do vento, o que faz com que seja exercido um potencial da zona superior e da zona inferior alternadamente, onde, juntamente com o potencial atrativo origina sucessivas mudanças de bordo, obtendo um caminho aos “zig-zags” (figura 3.27).

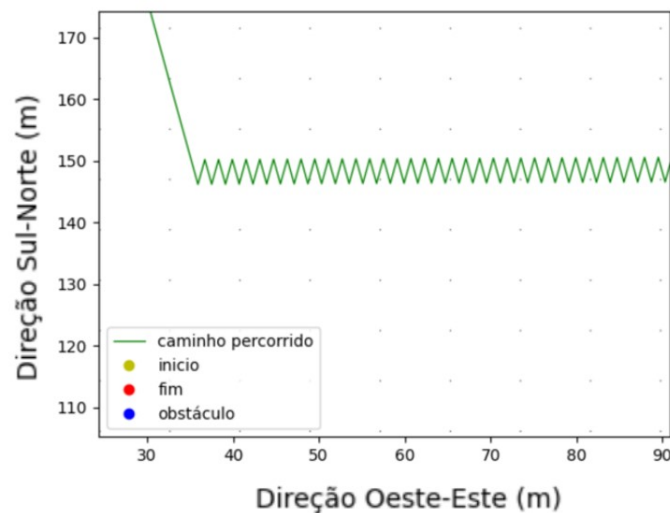


Figura 3.27: Caminho obtido com sucessivas mudanças de bordo para direções de navegação próximas da direção do vento.

Para resolver o problema apresentado na figura 3.27, foi criada uma variável que indica qual o último potencial exercido no veleiro, desta forma, para ângulos próximos da direção do vento é tido em conta o valor da variável evitando as sucessivas mudanças de bordo.

À custa da soma da força atrativa (que atrai o veleiro na direção do ponto de chegada

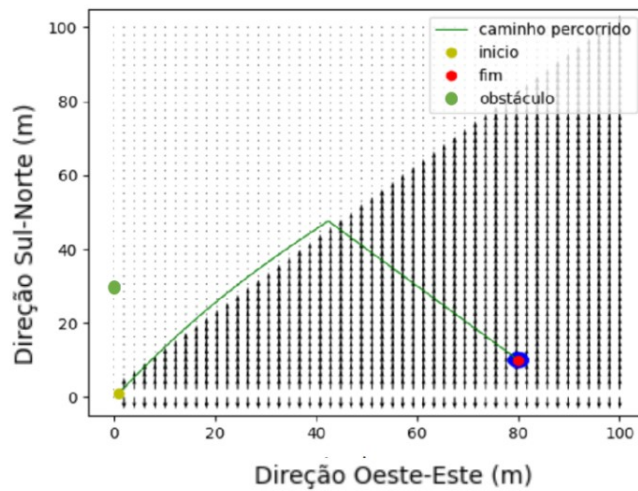


Figura 3.28: Simulação de navegação na ZNN.

com a força repulsiva representada pelas setas na ZNN, é obtido o caminho da figura 3.28. O círculo azul junto ao ponto de chegada, representa a área a partir da qual se considera que o veleiro chegou ao destino pretendido. Um dos problemas associados ao facto de atribuir um potencial repulsivo constante ao vento, é a criação de caminhos pouco otimizados, que se afastam cada vez mais do caminho pretendido, à medida que o ponto de chegada se aproxima dos limites da ZNN, como se pode observar no cenário da figura 3.29, com uma comparação entre o caminho obtido (verde) e o caminho desejado (laranja).

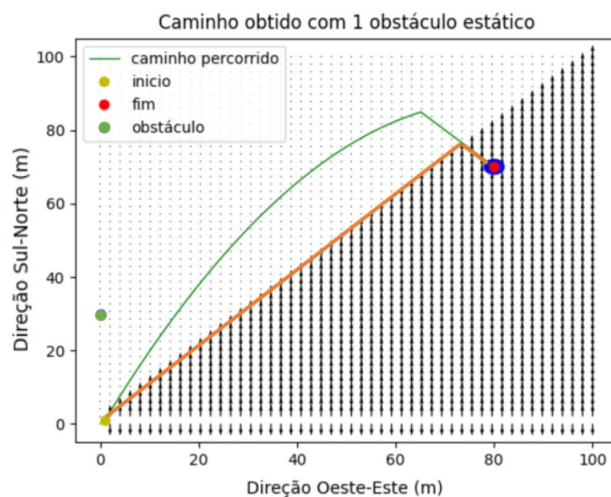


Figura 3.29: Comparação entre o caminho obtido (verde) e o pretendido (laranja).

Após uma tentativa (falhada) de desenvolver um método para variar a força repulsiva dependendo da posição do veleiro relativamente ao ponto de chegada, a abordagem foi novamente alterada chegando à solução final.

A estratégia passa por “esquecer” a existência de um potencial negativo para evitar obter os caminhos “curvos” (trajeto a verde na figura 3.29), para isso, se a próxima direção a seguir estiver dentro da ZNN, o **potencial atrativo** é calculado sobre a reta que limita essa mesma zona. Este processo mantém-se até que a direção pretendida já não se encontre dentro da ZNN, pelo que o ângulo máximo de mudança de bordo é limitado pelo ângulo superior e ângulo inferior da ZNN, desta forma, o número de manobras é reduzido para apenas uma, obtendo um trajeto com dois troços quando se pretende navegar na direção contrária ao vento sem a existência de quaisquer obstáculos. A figura 3.30 representa o raciocínio descrito.

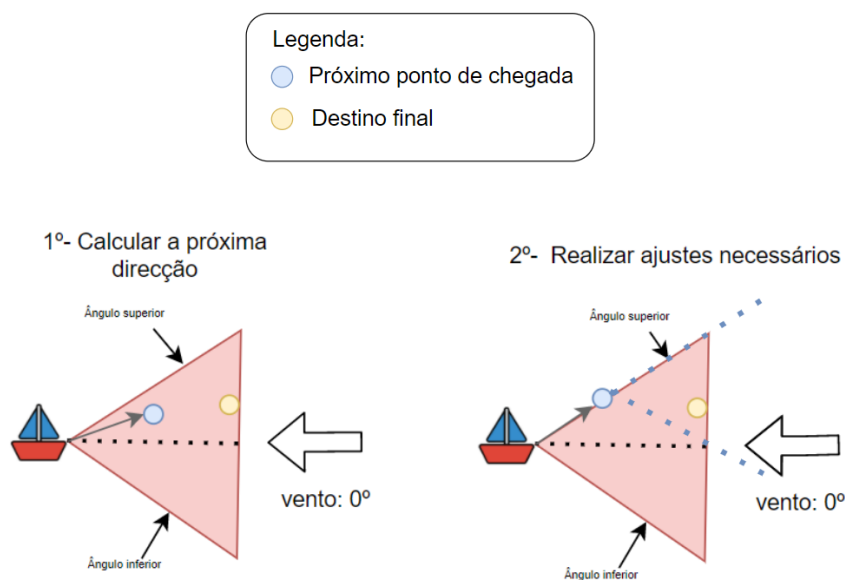


Figura 3.30: Estratégia utilizada para navegar numa ZNN.

No cenário apresentado, é considerado um vento com direção de 0°, criando uma ZNN (representada pela região triangular a vermelho) onde a parte superior é limitada por um ângulo de 45°, e a parte inferior por -45°. Neste caso, o ponto de destino final (círculo a amarelo), encontra-se dentro da ZNN, havendo a necessidade de uma navegação à bolina cerrada. No 1º passo, o algoritmo começa por determinar o próximo ponto de chegada (ponto azul) e a direção (ângulo) entre a posição atual do veleiro e esse mesmo ponto, caso essa direção se encontre dentro da ZNN é verificado se se localiza na parte superior ou inferior (figura 3.26), neste caso encontra-se na zona superior. No 2º passo, e sabendo a zona do próximo ponto, é calculada a equação da recta ² (equação 3.19) que limita a zona superior e que passa no ponto atual do veleiro. A ZNN é dinâmica, querendo isto dizer

²O formato da equação variará dependendo do próximo ângulo de navegação

que esta "move-se" com o veleiros. Na imagem 3.30, as retas a tracejado a azul representam a ZNN no próximo ponto de chegada.

$$\text{Longitude} = \text{Latitude} * m + b \quad (3.19)$$

Sabendo a equação da reta é atribuído um deslocamento na latitude e calculada a longitude correspondente, realizando ajustes nas forças a serem aplicadas no veleiro, de modo que a próximo ponto (ponto azul) se encontre sobre o limite da zona superior da ZNN. Este é um processo que se repete (1º e 2º passo) até que a próxima direção a seguir não se encontre dentro da ZNN, sendo o trajeto calculado normalmente a partir daí.

Nas figuras seguintes observam-se alguns dos resultados obtidos com esta estratégia para diferentes direções de vento (figuras 3.31 e 3.32. Para efeitos de simulação, foi considerado um vento constante sempre com a mesma direção.

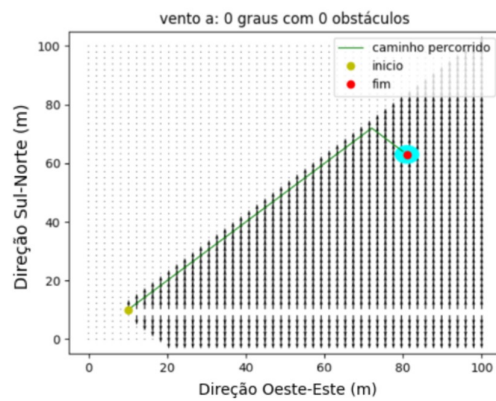


Figura 3.31: Simulação para um cenário com 0 obstáculos e com vento a 0º.

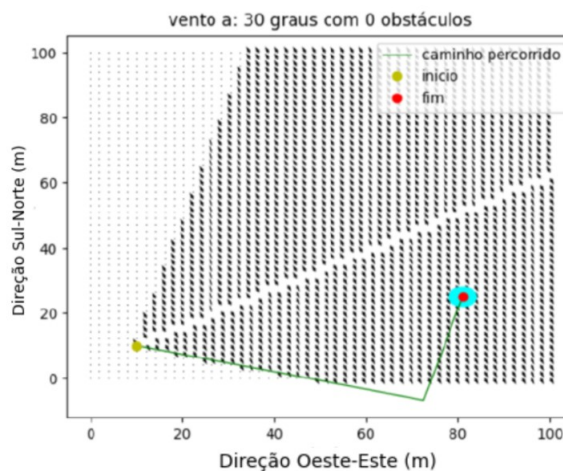


Figura 3.32: Simulação para um cenário com 0 obstáculos e com vento a 30º.

Para os casos em que o vento assume um valor de 45° , -45° , 135° ou -135° é necessário um tratamento de exceção. Considere-se o caso em que o vento tem uma direção de 45° (figura 3.33), obtém-se um ângulo superior a limitar a ZNN de 90° , o que representa um declive infinito, não sendo possível calcular a equação da reta. Para estes casos, é considerado um deslocamento no eixo da longitude e 0 para a latitude.

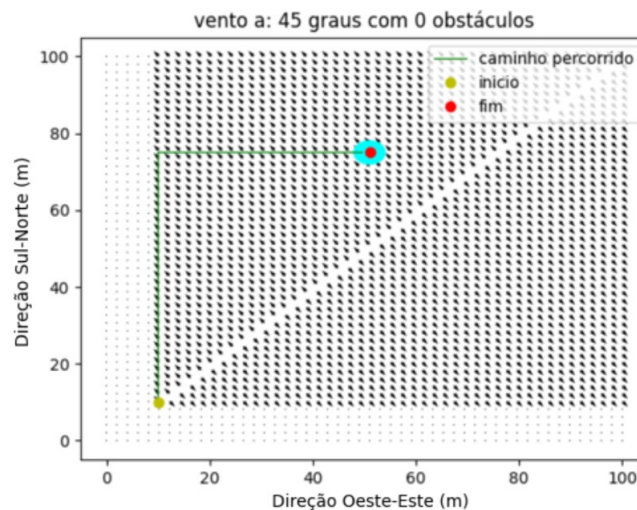


Figura 3.33: Simulação para um cenário com 0 obstáculos e com vento a 45° .

Como foi já referido, ventos com direções pertencentes ao intervalo $]135^\circ, 180^\circ[$ U $]-180^\circ, -135^\circ[$, é necessário uma função específica para tratar desses casos. Considere-se o cenário em que o vento se encontra na direção dos -140° , obtendo um ângulo superior de -95° ($-140^\circ + 45^\circ$), e um ângulo inferior de 175° ($-140^\circ - 45^\circ$), devido à zona de transição de -180° para 180° . Obtém-se uma zona inferior dividida em duas partes, a 1ª dos 175° aos 180° e a 2ª dos -180° aos -140° . Esta zona de transição dá origem a problemas, nomeadamente quando é verificado em que zona se encontra o ponto de chegada, pois o ângulo limite da zona superior é inferior ao ângulo limite da zona inferior. Para o mesmo cenário, tendo um ponto de chegada na zona superior (entre -140° e -95°), o potencial será aplicado enquanto a próxima direção for superior ao ângulo inferior, ora, como o próximo ângulo de navegação é negativo, esta nunca será superior ao ângulo inferior (175°).

3.2.4 Aglomerados de obstáculos estáticos

Nesta seção pretende-se demonstrar o raciocínio tido para lidar com conjuntos de obstáculos estáticos. Será abordada a questão em específico para o caso dos pontões, sendo que a estratégia é semelhante para outros tipos de obstáculos estáticos, como por exemplo a linha da costa, ou massas rochosas. Um pontão [Rutledge 2011] trata-se de uma estrutura estreita que protege a linha da costa de correntes e marés. É construída a partir de terra e estendida para a água (figura 3.34).



Figura 3.34: Exemplo de um pontão de pedra. Referência:[P. D. Website 2015]

Tendo como referência o trabalho [Naeem, Henrique e Hu 2016], considerou-se o pontão como uma estrutura retangular, sendo decomposto em obstáculos estáticos virtuais a partir dos seus limites, como se pode observar na figura 3.35.

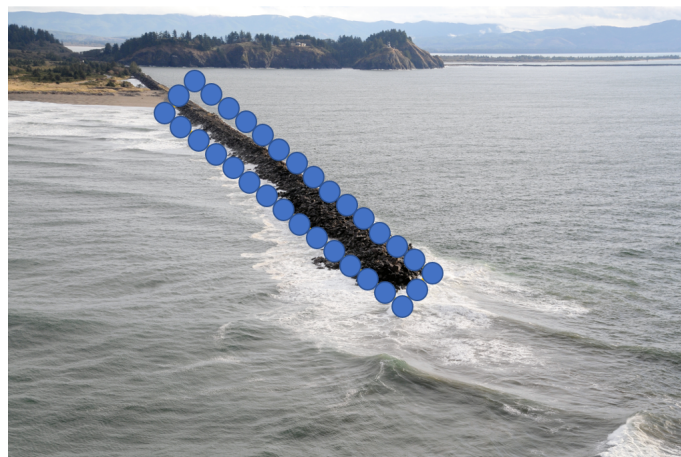


Figura 3.35: Exemplo de um pontão de rochas. Referência:[P. D. Website 2015]

Um ponto importante a ter em atenção, é a separação e o raio de influência dos obstáculos virtuais que são atribuídos.

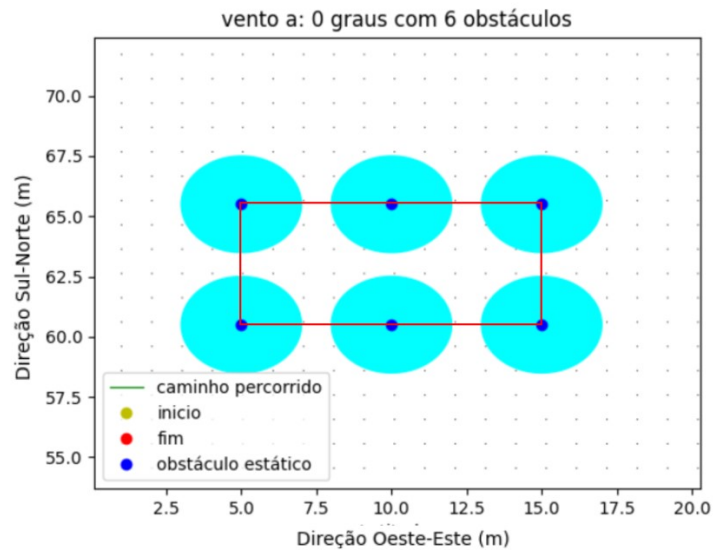


Figura 3.36: Exemplo de um pontão decomposto em obstáculos estáticos virtuais.

Observando a figura 3.36, verifica-se que existem espaçamentos entre os diferentes obstáculos, a solução para resolver este problema passa por aumentar o raio de influência de cada obstáculo ou diminuir o espaçamento entre eles. Optou-se por diminuir o espaçamento entre os diferentes obstáculos, pois esta abordagem não ocupa espaço desnecessariamente, e permite utilizar a mesma definição de obstáculo utilizada para representar obstáculos móveis como as boias/boias de sinalização. É possível analisar uma comparação entre as duas estratégias na figura 3.37.

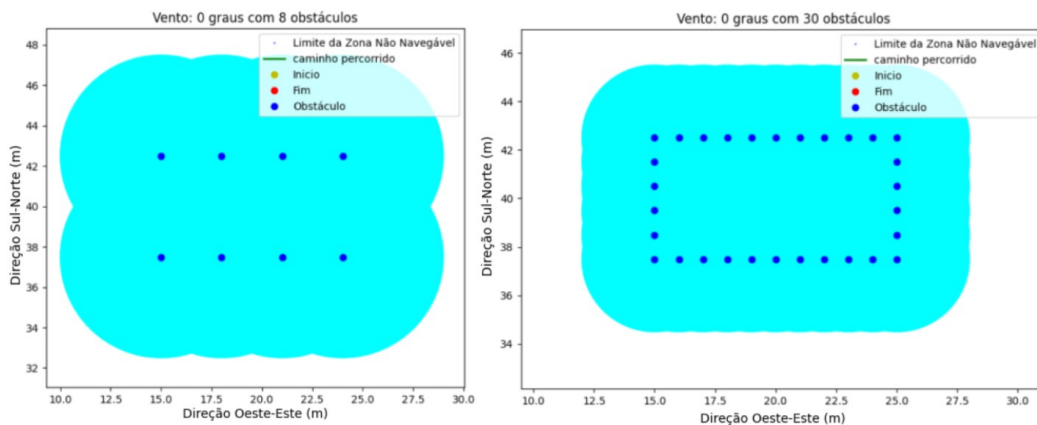


Figura 3.37: Comparação entre as duas estratégias. Aumento do raio de influência do obstáculo (lado esquerdo), e diminuição da distância entre os diferentes obstáculos (lado direito).

Depois de criados, os obstáculos são adicionados à lista de obstáculos presentes no ambiente de navegação. Para evitar colisões com aglomerados de obstáculos, é necessário apenas considerar as forças repulsivas de cada um deles (figura 3.38).

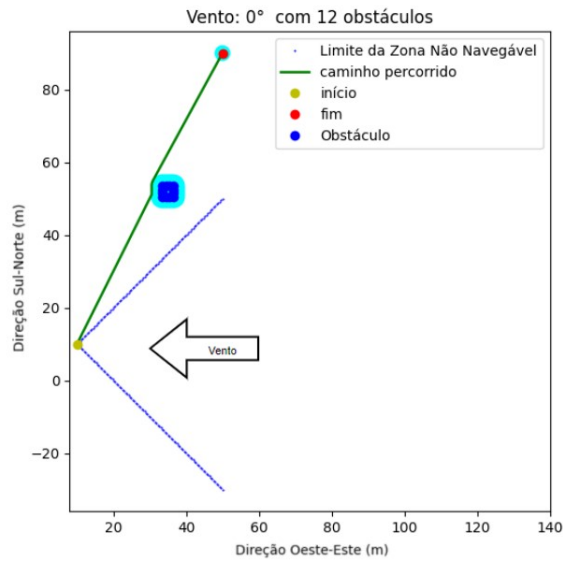


Figura 3.38: Exemplo de navegação com um pontão composto por 12 obstáculos e vento a 0°.

3.2.5 Obstáculos dentro da zona não navegável

Um dos objetivos desta dissertação passa por conseguir obter uma navegação autónoma evitando obstáculos³. Quando estes obstáculos se encontram dentro da ZNN torna-se necessário especial atenção, para que o veleiro ao contornar os mesmos obstáculos realize um trajeto correto e realista.

Tendo por base os resultados apresentados em [Plumet, Pêtrès, Gas e Ieng 2014], e o facto de a estratégia da navegação numa ZNN privilegiar apenas uma mudança de bordo decidiu-se dividir ZNN em duas zonas, como indicada na figura 3.39.

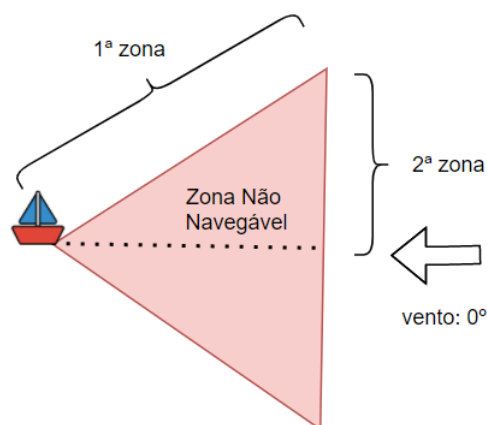


Figura 3.39: Zonas consideradas dentro da ZNN

³Como o algoritmo tem uma visão estática dos obstáculos, as estratégias apresentadas consideram os obstáculos estáticos e móveis

A estratégia utilizada para evitar obstáculos na 1ª zona encontra-se na figura 3.40.

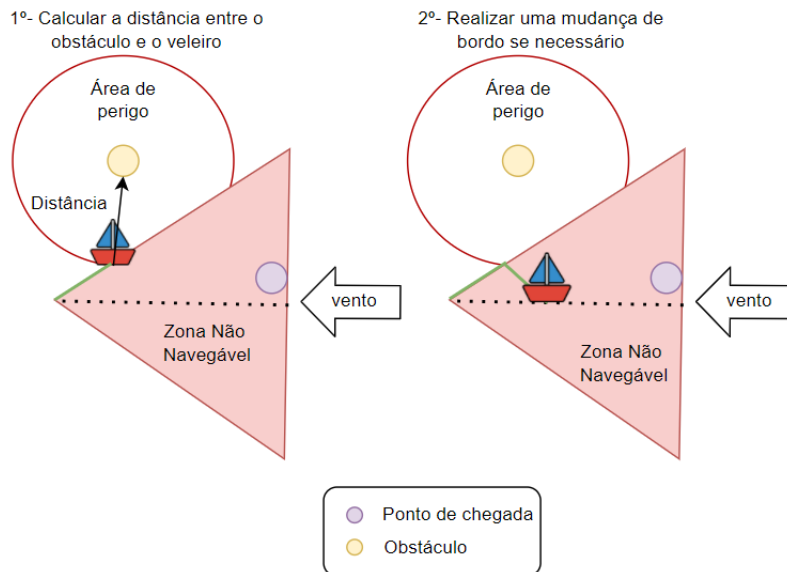


Figura 3.40: Estratégia de navegação com obstáculos dentro da ZNN para a 1ª zona.

Em cada iteração do algoritmo é calculada a distância do veleiro a cada um dos obstáculos previamente identificados. Caso essa distância seja inferior ao raio da "Área de perigo" (na prática corresponde a uma distância superior à de influência do obstáculo) é feita uma mudança de bordo.

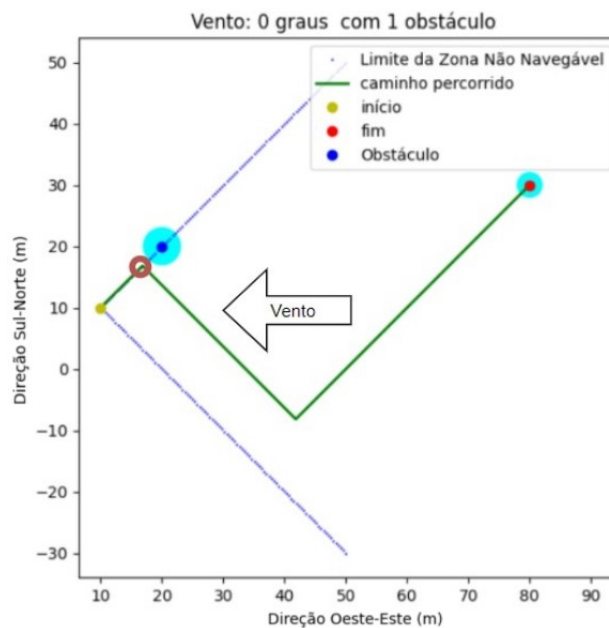


Figura 3.41: Simulação de navegação na ZNN com 1 obstáculo e vento a 0°.

Quando o veleiro se encontra próximo do obstáculo, realiza uma mudança de bordo assinalada na simulação apresentada na figura 3.41, pelo círculo vermelho.

No resultado da figura 3.42, verifica-se que existe um obstáculo estático, mas como este não se encontra nas proximidades do veleiro, ou seja, como a distância do veleiro ao obstáculo é superior ao raio da área de influência do obstáculo, não existe risco de colisão, obtendo o trajeto indicado.

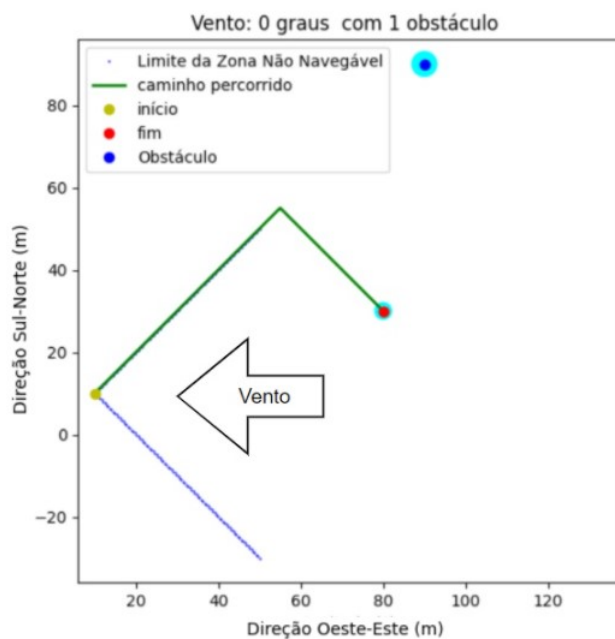


Figura 3.42: Simulação de navegação na ZNN com 1 obstáculo e vento a 0°.

Uma das limitações detetada na estratégia proposta, ocorre em alguns cenários específicos, onde o obstáculo e o ponto de chegada se encontram em “linha”, onde é detetado perigo de colisão com o obstáculo e é realizada uma mudança de bordo, mas como o ponto de chegada se encontra por detrás do obstáculo, o trajeto obtido passa “por cima” do mesmo, como se observa na figura 3.43.

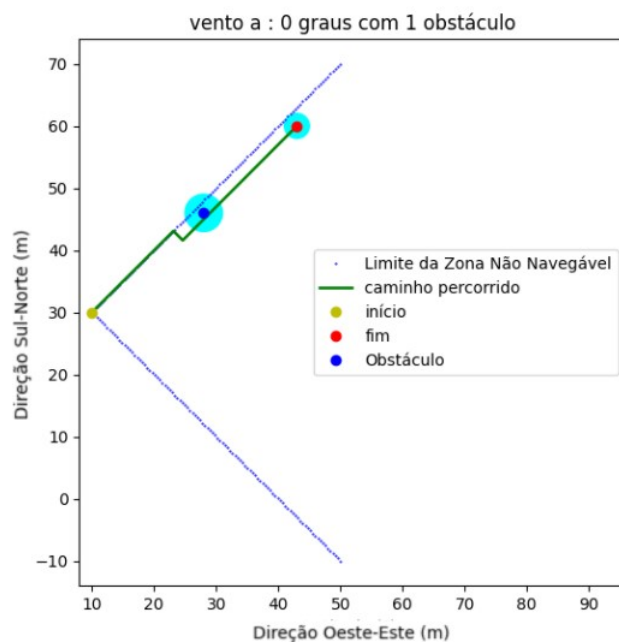


Figura 3.43: Limitação da estratégia utilizada para obstáculos na 1ª zona.

Considerando agora um obstáculo estático na 2ª zona, o que corresponde ao mesmo estar no trajeto do veleiro, depois deste realizar uma mudança de bordo em direção ao ponto de chegada, como indicada na figura 3.44.

A estratégia utilizada para evitar colisões neste tipo de cenários passa por uma abordagem de previsão.

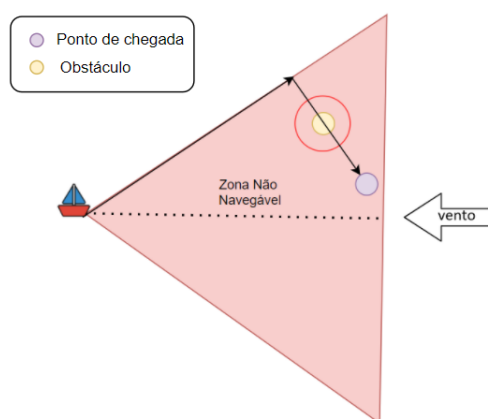


Figura 3.44: Representação de um obstáculo na 2ª zona.

No caso da figura 3.44, o veleiro realiza o 1º trecho ao longo do limite da zona superior da ZNN. Quando o ângulo entre a posição do veleiro e o ponto de chegada é inferior ao ângulo que limita a zona inferior, é realizada uma mudança de bordo. Sabendo o ângulo

em que a mudança de bordo será efetuada, torna-se então possível de prever se existe algum obstáculo nesse trecho, e nesse caso, a mudança de bordo é feita com uma maior antecedência.

O 1º passo passa por compreender quando e qual a zona onde existe risco de colisão. Como se pode observar na figura 3.45, a zona de perigo de colisão estaria localizada ao longo do trajeto que o veleiro navegaria quando realizasse uma mudança de bordo em direção ao obstáculo.

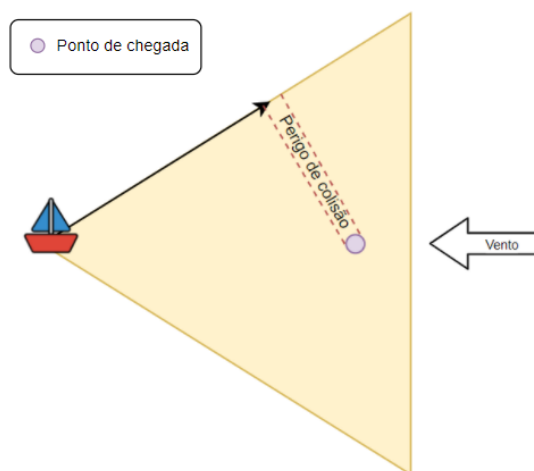


Figura 3.45: Área de possível colisão com obstáculos na 2ª zona.

No cenário da figura 3.46 encontram-se representados 3 obstáculos e o ponto de chegada (ponto roxo), a linha a tracejado preta representa o trajeto que o veleiro seguiria numa situação normal depois de realizar uma mudança de bordo em direção ao ponto de chegada. Através da figura conclui-se que, no limite, existe risco de colisão se a distância entre o centro do obstáculo e a reta a tracejado preta for **inferior ou igual** ao raio da área de efeito do obstáculo. No caso do obstáculo verde, a distância entre o seu centro e a reta a tracejado é **inferior ou igual** ao seu raio, pois está a “tocar” na reta, logo, existe perigo de colisão, o mesmo para o obstáculo amarelo, onde se verifica de uma forma clara que este se sobrepõe à reta. No caso do obstáculo azul, como a distância entre o seu centro e a reta é **superior** ao seu raio (distância de efeito) não existe perigo de colisão. Como na prática o veleiro não pode ser reduzido ao seu centro de massa, a verificação da distância é feita com uma margem.

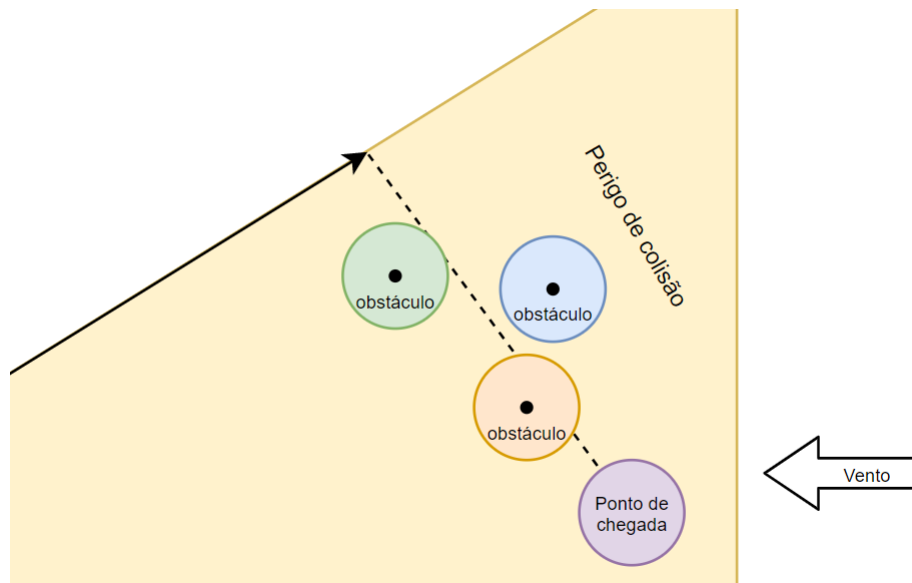


Figura 3.46: Cenário de possível colisão com obstáculos na 2ª zona.

Para o cálculo da distância perpendicular do obstáculo à reta, começa-se por determinar a equação que passa no ponto de chegada (reta a tracejado na figura 3.46) a partir do ponto de chegada e o ângulo de mudança de bordo. Sabendo a equação da reta e a localização do obstáculo, calcula-se distância o obstáculo à reta através da equação 3.20 [Wikipedia 2021], onde a equação da reta é dada por $a \cdot x + b \cdot y + c = 0$ e a Latitude e Longitude representam as coordenadas do veleiro.

$$dist = \frac{|a \cdot Latitude + b \cdot Longitude + c|}{\sqrt{a^2 + b^2}} \quad (3.20)$$

Antes de se realizar o cálculo da distância, é necessário garantir que o obstáculo se encontra dentro da ZNN. Isto é facilmente conseguido sabendo os ângulos que limitam as zonas superiores e inferiores, o ângulo entre o veleiro e os obstáculos e o ângulo entre o veleiro e o ponto de chegada.

Um aspeto relevante a ter em conta é o facto de, quando veleiro se encontra ainda longe do local da mudança de bordo, estas verificações não são necessárias, como tal, o cálculo das distâncias apenas é feito quando o veleiro se encontra a uma distância mínima do ponto da mudança de bordo. Terá que ser suficiente, para que, no caso limite de existência de colisão, haja tempo e espaço para uma mudança de bordo antecipada. O cálculo da distância perpendicular do veleiro até à reta que passa no ponto de chegada é feito da mesma forma já apresentada para o cálculo da distância de um ponto a uma reta.

A estratégia descrita encontra-se representada na figura 3.47.

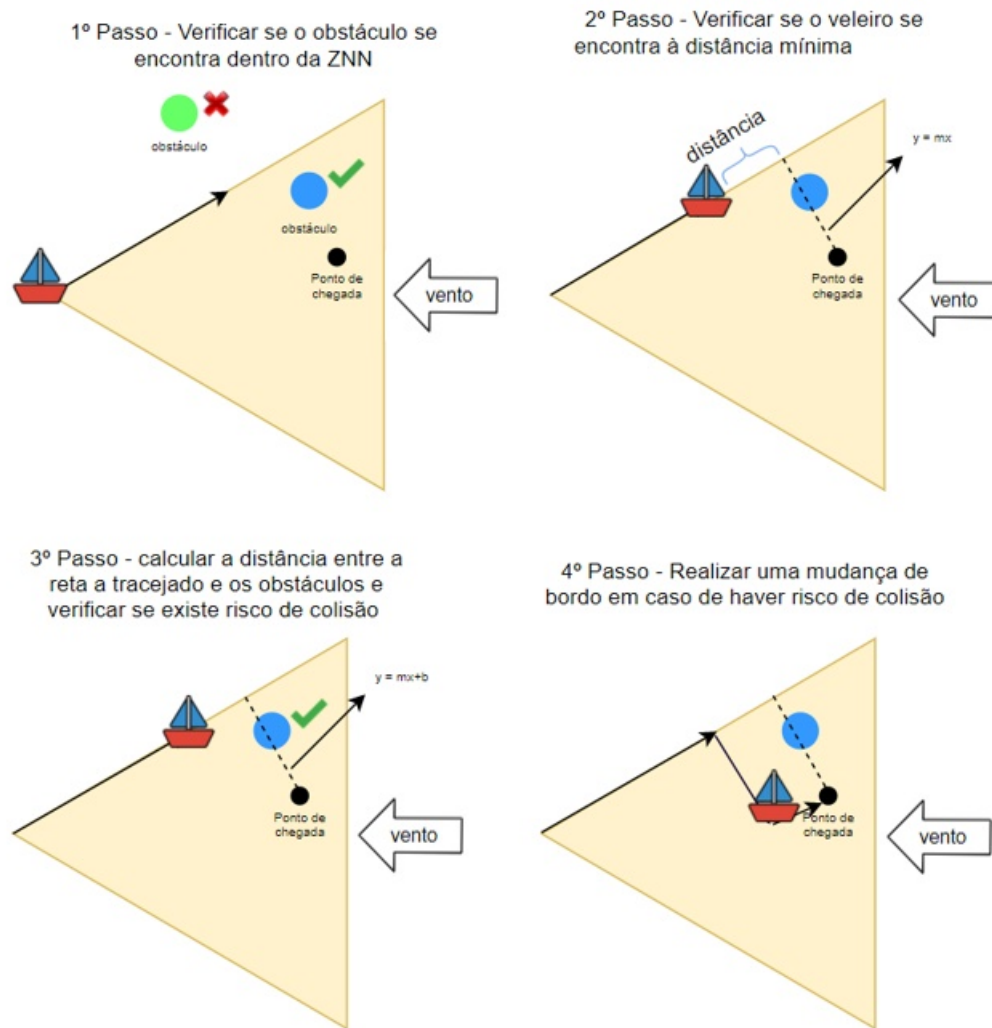


Figura 3.47: Estratégia de navegação com obstáculos dentro da ZNN na 2ª zona.

Mais uma vez, estando a trabalhar com o cálculo de declives, existem casos especiais, como é o caso dos 90° onde o declive é infinito. De modo a facilitar, para os casos em que algum dos módulos dos ângulos que limita a ZNN for igual a 90° , a mudança de bordo é feita logo no início. Isto acontece para ventos de: $45^\circ, -45^\circ, 135^\circ$ e -135° .

Uma das limitações desta estratégia é facto de as distâncias calculadas serem sempre valores positivos, não permitindo perceber se um obstáculo se encontra de um lado ou do outro da reta, levando a resultados pouco otimizados, como é o caso da figura 3.48, onde se verifica a existência do risco de colisão, mas devido ao facto de não se saber a posição do obstáculo relativamente à reta a tracejado, a mudança de bordo é feita com demasiada antecedência.

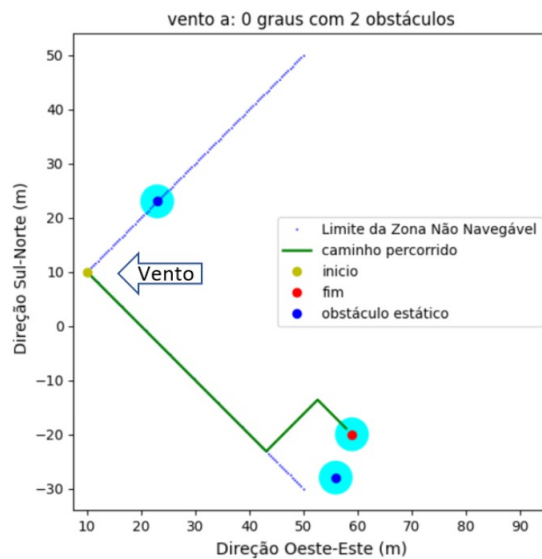


Figura 3.48: Limitação da estratégia utilizada para obstáculos na 2ª zona

3.2.6 Incurrir em Zonas não Navegáveis

Considere-se agora o cenário onde o veleiro se encontra a navegar **fora** da ZNN em direção ao ponto de chegada, e, ao encontrar um obstáculo o trajeto do veleiro poderá incorrer na ZNN. Para evitar uma colisão com o obstáculo, o veleiro necessita de navegar dentro da ZNN, como tal, realiza duas mudanças de bordo assinaladas pelos círculos a vermelho, como assinalado na figura 3.49,

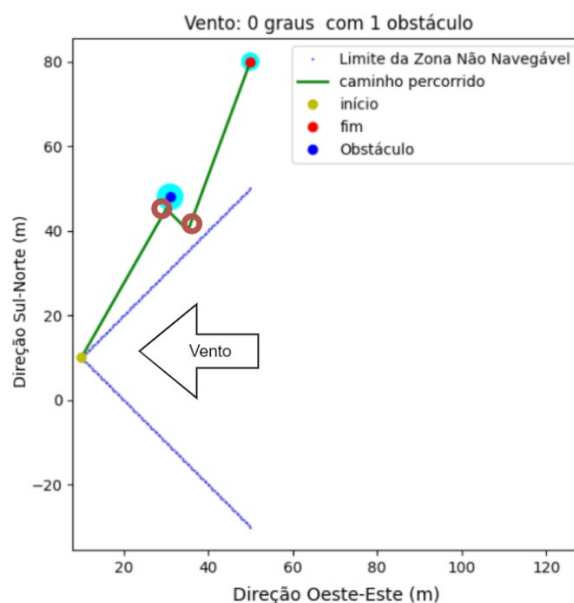


Figura 3.49: Simulação de navegação com 1 obstáculo e vento a 0°.

Como se pode observar na figura 3.49, o trajeto obtido resulta de um planeamento da rota feito ponto a ponto, característico de um método baseado em campos potenciais artificiais. No caso do planeamento do trajeto ser feito globalmente, o veleiro iria contornar o pontão pelo lado esquerdo.

Para o caso do obstáculo ser um pontão, começou-se por utilizar a mesma estratégia, obtendo o resultado da figura 3.50, onde se verificam sucessivas mudanças de bordo, devido mais uma vez à existência de uma grande concentração de obstáculos, muito próximos uns dos outros.

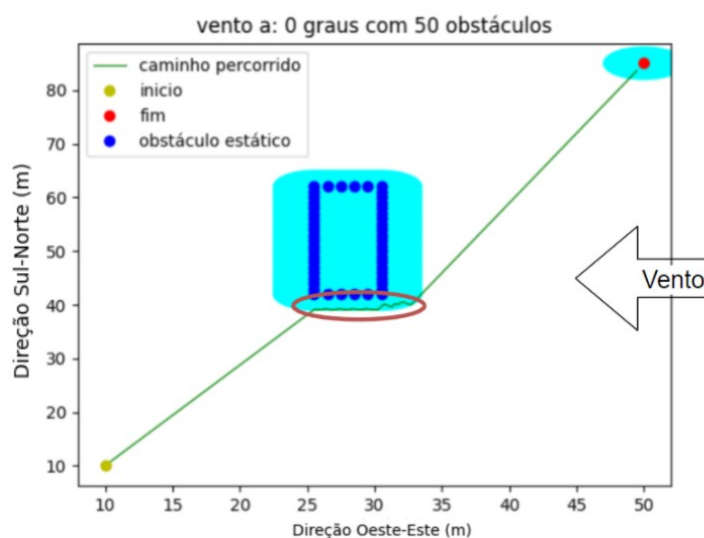


Figura 3.50: Simulação de navegação com 56 obstáculos e vento a 0° .

Para resolver este tipo de situações, quando é realizada a 1ª mudança de bordo é guardado o **index** (posição na lista de obstáculos) na matriz do obstáculo responsável pela mesma. Apenas quando o veleiro estiver a uma distância considerável desse mesmo obstáculo é que uma nova mudança de bordo poderá ser feita, desta forma, é possível “controlar” o número de mudanças de bordo, como se pode verificar na figura 3.51.

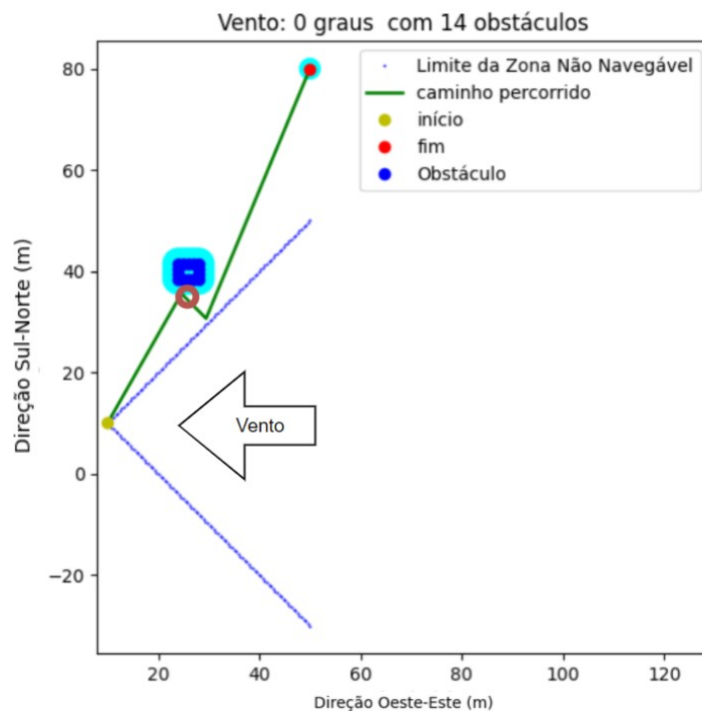


Figura 3.51: Simulação de navegação com 14 obstáculos e vento a 0° .

Na simulação do cenário da figura 3.51, com um obstáculo estático composto por um total de 14 obstáculos, obstáculos estes que podem ser observados na matriz na figura 3.52.

A matriz final de obstáculos estáticos contém cinco campos. O ID é um dos elementos da matriz pois é o que permite identificar um obstáculo e atualizar a sua posição se necessário. No cenário simulado existe apenas um pontão (exemplo), pelo que todos os obstáculos que o compõe têm o mesmo ID. Outro aspeto importante a notar é o facto de apenas o 1º obstáculo possuir o campo “Manobras” com o valor 1, estando coerente com o resultado da simulação obtida (figura 3.51), pois é realizada apenas **uma mudança de bordo devido a um obstáculo**, neste caso, o obstáculo com **Latitude** de 24 e uma **Longitude** de 38.5.

3.2. FUNÇÕES DOS CAMPOS POTENCIAIS

Tipo	Latitude	Longitude	Manobras	ID
0	24	38.5	1	1
0	24	41.5	0	1
0	25	38.5	0	1
0	25	41.5	0	1
0	26	38.5	0	1
0	26	41.5	0	1
0	27	38.5	0	1
0	27	41.5	0	1
0	28	38.5	0	1
0	28	41.5	0	1
0	24	39.5	0	1
0	28	39.5	0	1
0	24	40.5	0	1
0	28	40.5	0	1

Figura 3.52: Matriz com a lista de obstáculos obtida para o cenário da figura 3.51.

No caso desta dissertação, foi abordado com mais detalhe o caso do pontão como aglomerado de obstáculos, no entanto, a estratégia apresentada é válida para qualquer obstáculo de grandes dimensões como a linha da costa, ou um navio/embarcação que apresentem um contorno diferente, desde que, esses mesmos obstáculos sejam passíveis de ser identificados e caracterizados através de uma mensagem com os diferentes parâmetros.

SISTEMA PROPOSTO

4.1 Arquitetura do sistema

Neste capítulo será feita uma descrição dos diferentes elementos do sistema, com o objetivo de enquadrar o algoritmo desenvolvido em cada um deles.

4.1.1 Arquitetura de Hardware

Não tendo sido feita uma montagem completa de um sistema de controlo passível de testes e simulações de navegação, a arquitetura de hardware do sistema, tendo como referência alguns dos trabalhos desenvolvidos anteriormente como [Anjos 2020], [Gomes, Costa, Moutinho e Mota 2015] ou [Gomes, Santos, Pereira e Costa 2015], encontra-se representada na figura 4.1.

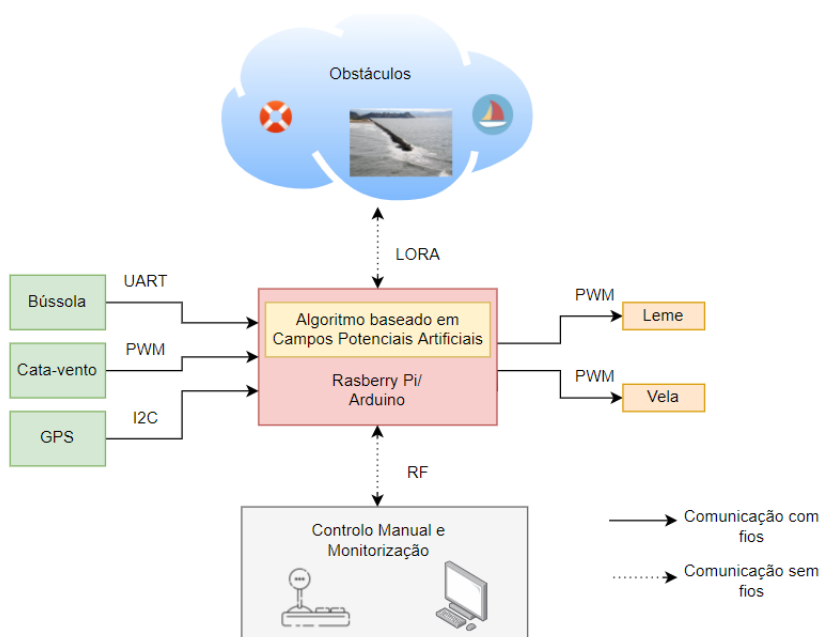


Figura 4.1: Interligações dos subsistemas do arquitetura de hardware e os protocolos utilizados.

Embutido no veleiro encontra-se um microprocessador Arduino ou uma Raspberry pi com o código do controlador, diversos sensores e atuadores, um módulo responsável por comunicação LORA e um de comunicação RF.

O algoritmo desenvolvido nesta dissertação inclui-se no módulo do controlador, sendo integrado diretamente na Raspberry pi ou no Arduino. O algoritmo utiliza diversos parâmetros relativos ao ambiente de navegação como entrada, tais como:

- **Direção do vento** - valor lido através do sensor cata-vento.
- **Posição atual do veleiro** - valor lido pelo GPS.
- **Próximo ponto de destino** - enviado pelo “Bloco de Controlo e Monitorização” através do módulo RF, ou definidos previamente numa lista de pontos a visitar (como no caso de uma regata).
- **Informação sobre os obstáculos presentes** - enviada pelos próprios obstáculos para o controlador, utilizando LORA.

Será com base nos valores de entrada referidos que o algoritmo irá calcular o novo rumo de navegação otimizado, sendo posteriormente feitos os ajustes necessários na vela e no leme com base no novo rumo.

4.1.2 Arquitetura de Software

O diagrama de blocos de software, encontra-se representado na figura 4.2. O bloco azul representa um módulo de perceção ou de aquisição de dados internos em tempo real, como a posição da vela, do leme, posição e inclinação do veleiro (com recurso a GPS), direção do vento, obstáculos presentes.

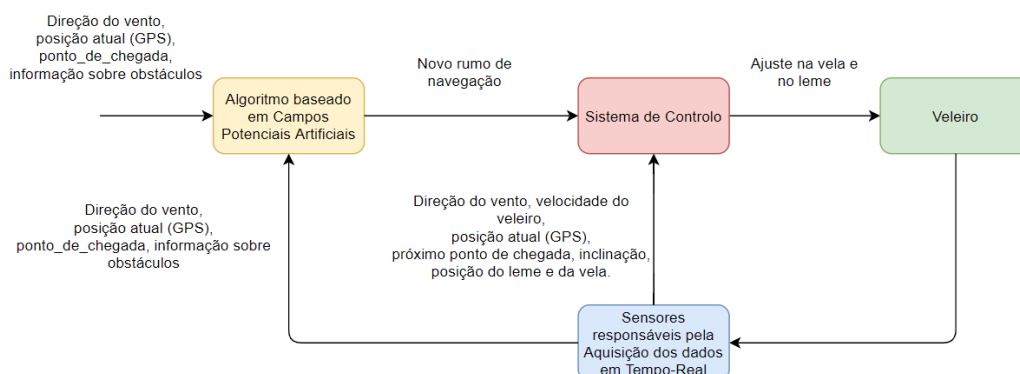


Figura 4.2: Diagrama de blocos do software.

O bloco amarelo representa o algoritmo desenvolvido baseado em campos potenciais artificiais, e tem como principal objetivo calcular o novo rumo de navegação com base na

direção do vento, próximo ponto de chegada, informação sobre os obstáculos presentes e posição do veleiro, valores estes que são enviados pelo módulo de percepção (bloco azul).

O novo rumo de navegação é enviado para o sistema de controlo. É com base no rumo recebido e nos dados enviados pelo módulo de percepção, que o sistema de baixo nível realiza os devidos ajustes no leme e na vela do veleiro. Este processo repete-se até que o veleiro chegue ao ponto de chegada.

Analisando agora com mais detalhe a estrutura do controlador (bloco vermelho da figura 4.2), e tendo por base a estrutura apresentada em [Gomes, Costa, Moutinho e Mota 2015], verifica-se que esta se encontra dividida em três níveis, como se pode observar na figura 4.3.

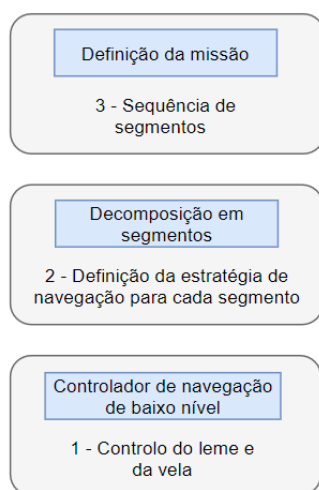


Figura 4.3: Estrutura hierárquica do controlador.
Referência:[Gomes, Costa, Moutinho e Mota 2015].

Cada nível tem diferentes funções na estratégia de controlo [Gomes, Costa, Moutinho e Mota 2015]:

- **Nível 1 – Controlador de navegação de baixo nível:** representa o controlo de baixo nível do leme e da vela.
- **Nível 2 – Decomposição em segmentos:** dividindo o trajeto total num conjunto de segmentos, neste nível é definida a estratégia de navegação mais adequada para cada um dos segmentos, tendo em conta fatores como o ponto de chegada, a direção do vento e a posição de obstáculos existentes.
- **Nível 3 – Definição de missão:** nível correspondente à especificação das características do modo de regata com a definição das várias etapas.

Atendendo à definição de cada um dos níveis, é possível concluir que o algoritmo desenvolvido se enquadra no nível 2 da estrutura hierárquica do controlador, onde através

de valores de entrada como a direção do vento, coordenadas da posição atual, do ponto de chegada e informação sobre obstáculos, é calculado o novo rumo de navegação otimizado para cada um dos segmentos que compõe o trajeto total.

4.1.3 Fluxograma do Sistema

Após definição de um ponto de chegada, todo o caminho desde a posição inicial até ao destino é dividido em troços/segmentos, criando assim um conjunto de pontos de chegada, onde o último elemento desse mesmo conjunto representa o ponto final definido. Seguidamente, é feita uma inicialização global das variáveis constantes do algoritmo. Cada um dos subpontos de chegada, como visto no subcapítulo anterior, é enviado para o controlador e posteriormente para o algoritmo, que é responsável por garantir que o veleiro percorre cada um dos segmentos até chegar ao destino pretendido, como se observa no fluxograma apresentado na figura 4.4.

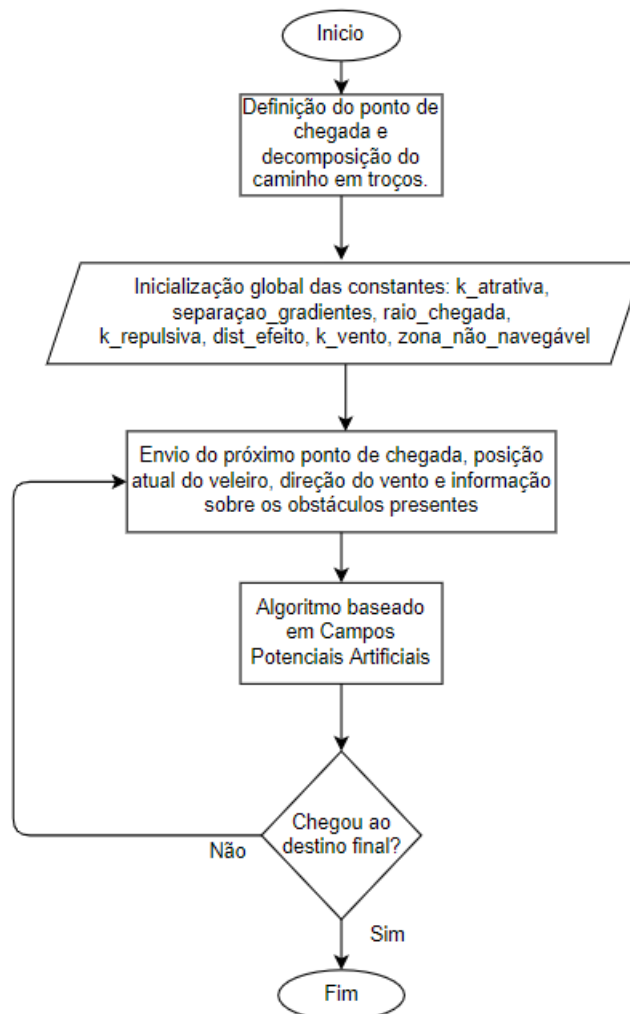


Figura 4.4: Fluxograma global do sistema.

Analisando agora o fluxograma referente ao bloco “Algoritmo baseado em Campos Potenciais Artificiais” da figura 4.4.

São primeiramente recebidos e interpretados os parâmetros de entrada do algoritmo já referidos, como a posição atual do veleiro, o valor da direção do vento lido, o ponto de chegada do segmento em questão e informação sobre os obstáculos presentes. Depois de calculadas as forças (atrativas e repulsivas) é determinado o próximo rumo de navegação a seguir, e caso esse rumo se encontre dentro da ZNN torna-se necessário adotar outra estratégia para o cálculo das mesmas forças, agora dentro de uma ZNN. O próximo ângulo de navegação é posteriormente enviado para o controlador de modo a serem feitos os devidos ajustes na vela e no leme (figura 4.5).

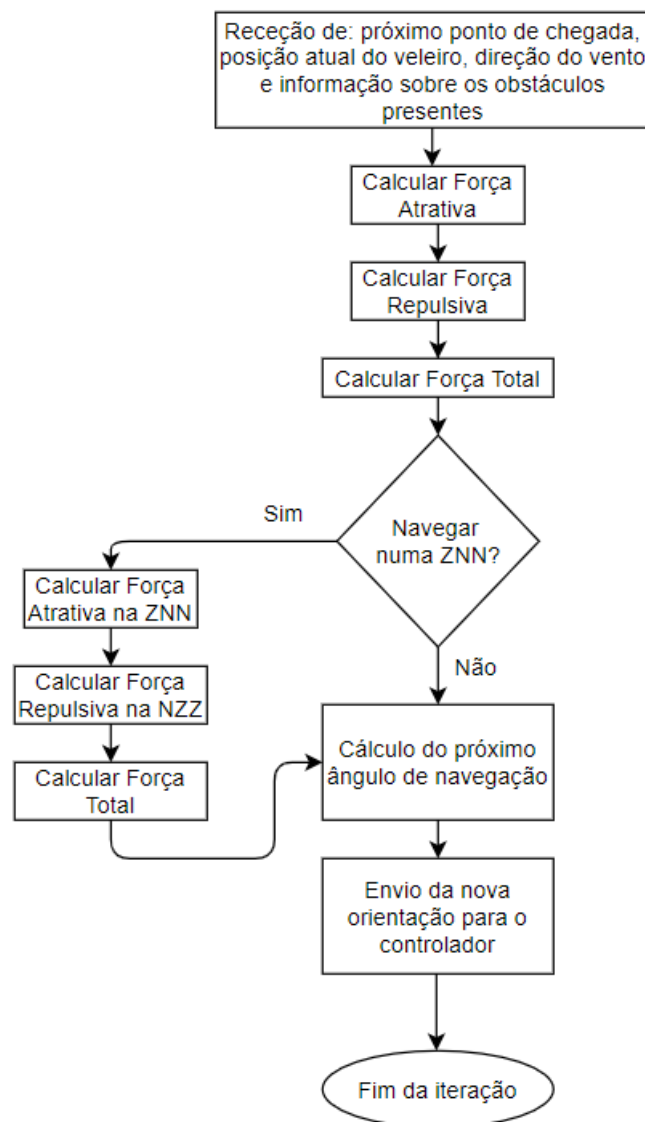


Figura 4.5: Fluxograma do algoritmo baseado em campos potenciais artificiais

4.2 Formato das mensagens

Nesta dissertação foram considerados obstáculos estáticos e móveis. As mensagens enviadas pelos obstáculos permitirão que o algoritmo os identifique e localize corretamente.

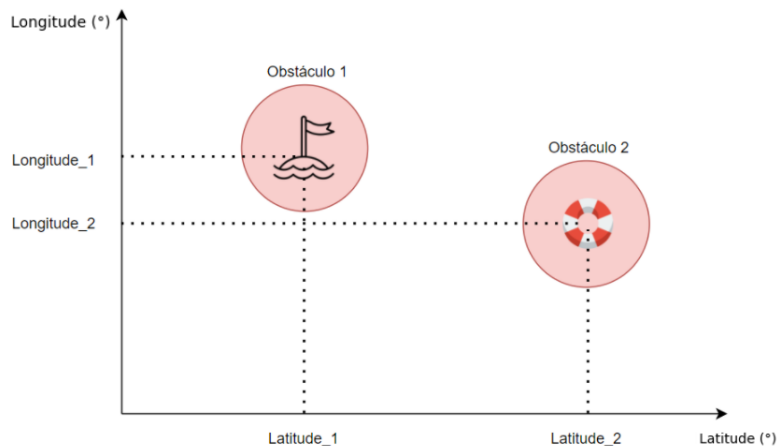


Figura 4.6: Representação de obstáculos.

O círculo a vermelho tem como finalidade limitar a distância a partir da qual o obstáculo poderá exercer algum tipo de influência no veleiro, e as coordenadas do obstáculo representam o seu centro de massa, como se pode observar na figura 4.6.

Considerou-se um obstáculo estático algo que não varia a sua posição ao longo do tempo, como é o caso dos pontões ou de uma estrutura fixa que seja um impedimento à navegação, que representam aglomerados de obstáculos estáticos. Por outro lado, considerou-se como um obstáculo móvel algo que varia a sua posição ao longo do tempo, como é o caso de outras embarcações ou boias/boias de sinalização. No caso das embarcações não existe qualquer tipo de estratégia de previsão da rota, apenas uma atualização da posição do obstáculo ao longo do tempo. Relativamente às boias/boias de sinalização, embora não variem muito a sua posição (quando comparado com uma embarcação), estando presentes num ambiente marítimo onde existem fatores externos como as marés ou o vento, conclui-se que a sua posição também variará.

As mensagens enviadas poderão então ter dois formatos, dependendo do tipo de obstáculo em questão:

- **Formato 1:** Obstáculos móveis como por exemplo outras embarcações ou boias/boias de sinalização.
- **Formato 2:** Obstáculos estáticos, como por exemplo os pontões ou linhas costeiras.

Embora os formatos de mensagem sejam muito semelhantes, torna-se necessário distingui-los, de forma que o algoritmo consiga interpretar o tipo de obstáculo em questão e aplicar a estratégia adequada.

4.2.0.1 Formato 1

Uma mensagem enviada por um obstáculo móvel é composta por quatro campos, apresentados seguidamente.

- **Tipo:** Campo que permite ao algoritmo identificar se a mensagem provem de um obstáculo móvel (Tipo = 0) ou de um obstáculo estático (Tipo = 1).
- **Latitude:** Representa a latitude do centro geométrico do obstáculo.
- **Longitude:** Representa a longitude do centro geométrico do obstáculo.
- **ID¹:** Campo que tem como objetivo identificar o obstáculo em questão.

Uma mensagem com o formato 1, enviada por um obstáculo móvel (uma boia neste caso), pode ser observada no cenário da figura 4.7. Como foi referido, este formato é composto por quatro campos: o 1º campo tem o valor 0 por se tratar de um obstáculo móvel, o 2º campo com o valor de 38.69409342° que representa a latitude do obstáculo, o 3º campo com o valor de -9.20657113° que indica a longitude do obstáculo, e o 4º campo, que representa o ID do obstáculo e tem o valor 1.

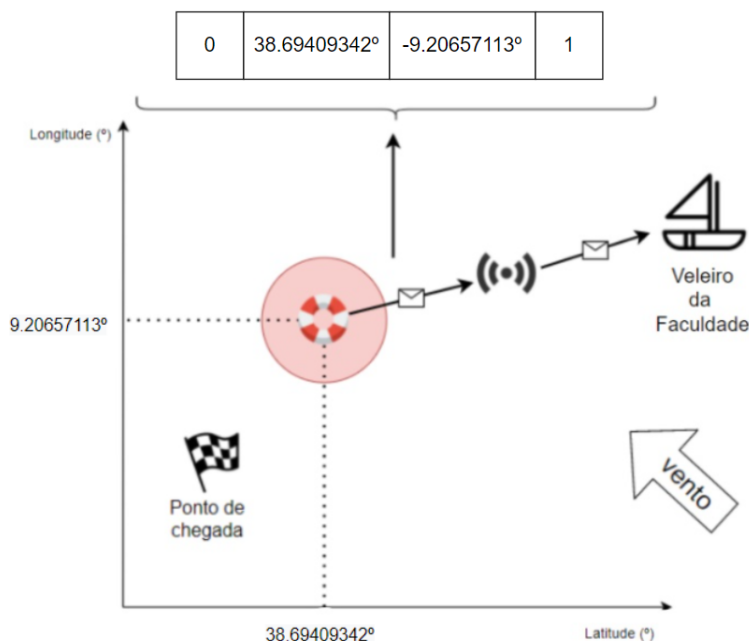


Figura 4.7: Formato da mensagem enviada por um obstáculo móvel.

Nos casos dos obstáculos móveis, após um certo intervalo de tempo é enviada outra mensagem com a sua nova posição, onde, após uma verificação do seu ID na lista dos

¹Para este campo, foi atribuído um valor inteiro a cada obstáculo para efeitos de simulação, não representando uma estratégia bem suportada, pelo que se sugere que seja repensada a abordagem para este campo

obstáculos, e caso já tivesse enviado alguma mensagem previamente, a sua posição seria atualizada, caso contrário seria adicionado à lista de obstáculos, como se pode ver na figura 4.8, onde sempre que o obstáculo móvel envia uma mensagem, o algoritmo embutido no sistema de controlo do veleiro atualiza a sua posição.

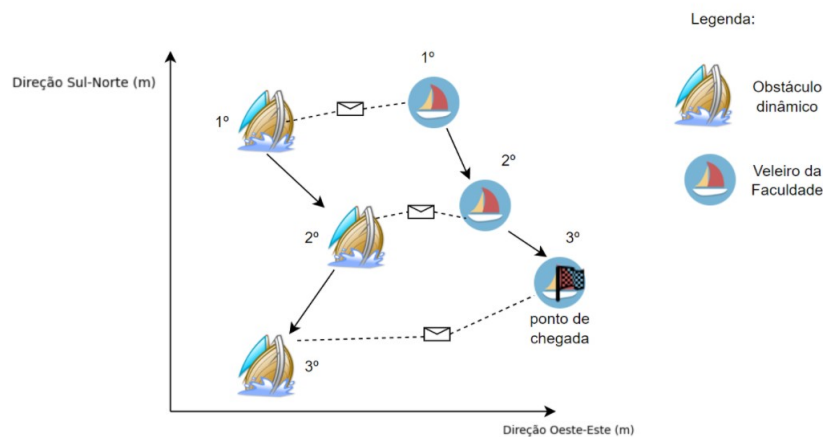


Figura 4.8: Exemplo de um cenário com um obstáculo móvel.

De notar que, com esta abordagem, a posição de um obstáculo móvel é atualizada sempre que uma mensagem é recebida, sendo considerado e interpretado pelo algoritmo como um obstáculo estático na última posição enviada.

4.2.0.2 Formato 2

Trata-se de um formato de mensagem utilizado pelos obstáculos estáticos, onde a mensagem enviada difere do formato anterior, pois, para além de existirem parâmetros como a largura e o comprimento (no caso de um pontão) que necessitam de ser fornecidas ao algoritmo, não existe a necessidade de atualizar a posição do obstáculo, logo, uma mensagem neste formato é recebida e analisada apenas uma vez pelo algoritmo por cada obstáculo estático. Caso seja enviada mais vezes, é identificada e ignorada.

A mensagem é composta por seis campos. De destacar que, os campos latitude e longitude referem a posição do centro geométrico do retângulo (no caso de um pontão ou obstáculos com tamanhos superiores) que representa o obstáculo.

Os campos que compõe a mensagem são apresentados e descritos seguidamente.

- **Tipo:** Campo que permite ao algoritmo identificar se a mensagem provem de um obstáculo móvel (Tipo = 0) ou de um obstáculo estático (Tipo = 1).
- **Latitude:** Representa a latitude do centro geométrico do obstáculo.
- **Longitude:** Representa a longitude do centro geométrico do obstáculo.

- **Comprimento:** Campo que contém o comprimento (metros) do retângulo que limita o obstáculo.
- **Largura:** Campo que contém a largura (metros) do retângulo que limita o obstáculo.
- **ID²:** Campo que tem como objetivo identificar o obstáculo em questão.

O algoritmo, quando recebe uma mensagem que é enviada por um obstáculo estático, a partir do comprimento, largura e centro do mesmo, calcula o número de obstáculos estáticos que o limitam e as respectivas posições, sendo posteriormente adicionados a uma lista que contém todos os obstáculos, onde o obstáculo estático em questão se trata de um pontão e os círculos a vermelho representam os obstáculos estáticos virtuais que o compõem.

Analisando cada um dos campos da mensagem enviada por um pontão na figura 4.9, verifica-se que: o **1º campo tem o valor 1** por se tratar de um obstáculo estático, o **2º campo com o valor -22.816419°** que representa a latitude do obstáculo, o **3º campo o valor -47.070563°** que indica a longitude do obstáculo, o **4º campo possui o valor 80** que representa o seu comprimento, o **5º campo com o valor 8** que indica a largura, e por último, o **6º campo, com o ID do obstáculo com o valor 1**.

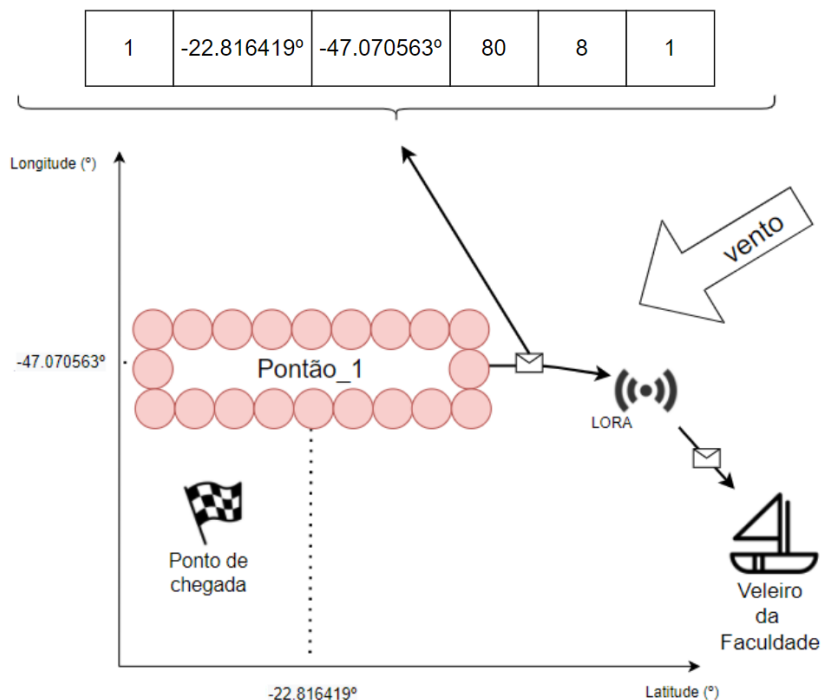


Figura 4.9: Formato da mensagem enviada por um pontão.

²Para este campo, foi atribuído um valor inteiro a cada obstáculo para efeitos de simulação, não representando uma estratégia bem suportada, pelo que se sugere que seja repensada a abordagem para este campo

Durante o desenvolvimento do algoritmo, surgiu a necessidade de controlar o número de mudanças de bordo que são realizadas para evitar colisões. Uma mudança de bordo para evitar um obstáculo, é realizada quando certas condições são verificadas. Como abordado no subcapítulo 3.2.5, nos casos onde é necessário evitar um obstáculo na 1ª zona da ZNN (por exemplo), a manobra é feita caso o veleiro se encontre dentro da área de influência do obstáculo, ora, como esta condição é verificada várias vezes num curto espaço de tempo, múltiplas mudanças de bordo são realizadas, onde, para além de se obter um caminho oscilante, representa um trajeto irreal. De modo a controlar o número de manobras que são realizadas sobre o mesmo obstáculo, adicionou-se o campo “Manobras” à lista de obstáculos, associado a cada obstáculo e com o valor inicial de 0.

Serão apresentados agora dois exemplos de simulações realizadas para o mesmo cenário, com (4.12) e sem (figura 4.10) a utilização do campo “Manobras”.

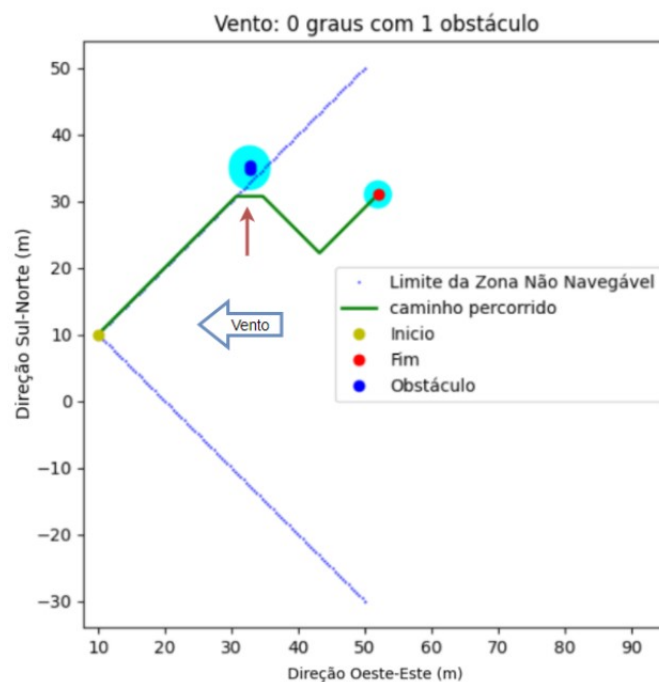


Figura 4.10: Simulação para um cenário com 1 obstáculo e com vento a 0°, **sem ter em consideração o campo Manobras**.

Em primeiro lugar, verifica-se que o caminho obtido na simulação é incorreto na zona assinalada pela seta a vermelho. Uma aproximação do lado esquerdo desse troço encontra-se representado na figura 4.11.

Como se pode verificar, nesta zona do trajeto, o veleiro encontra-se a uma distância onde existe risco de colisão, sendo realizada uma mudança de bordo. Quando a próxima iteração do algoritmo é feita, o veleiro ainda se encontra perto do obstáculo, sendo novamente detetado risco de colisão, sendo feita outra mudança de bordo, e assim sucessivamente, obtendo desta forma o caminho oscilante.

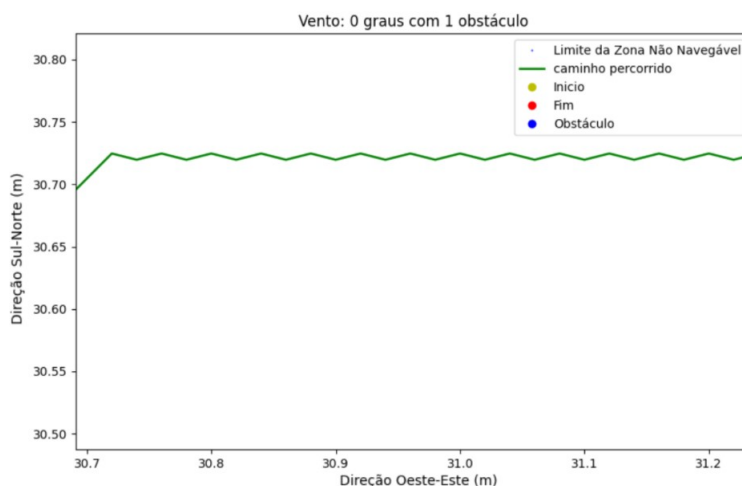


Figura 4.11: Simulação para um cenário com 1 obstáculo e com vento a 0°, **sem ter em consideração o campo Manobras**.

Observe-se agora, para o mesmo cenário, tendo em consideração o campo “Manobras” (Figura 4.12).

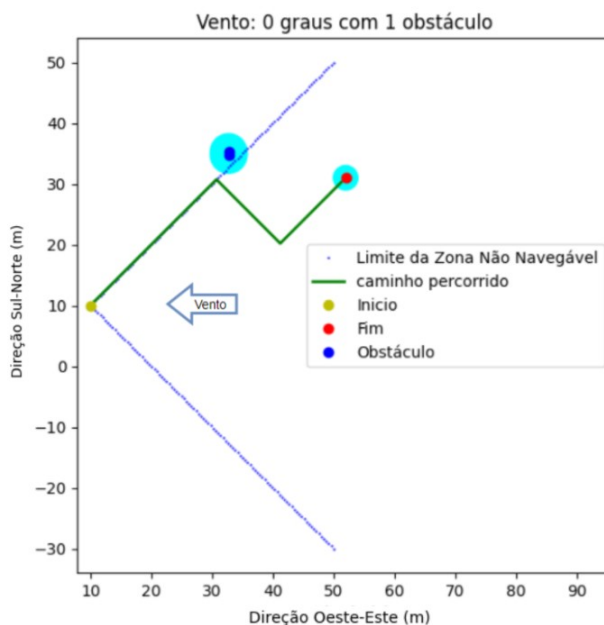


Figura 4.12: Simulação para um cenário com 1 obstáculo e com vento a 0°, **tendo em consideração o campo Manobras**.

Tendo agora em consideração o campo “Manobras”, verifica-se que é realizada apenas uma mudança de bordo para evitar a colisão com o obstáculo. Note-se que, **o campo referido**, apenas é tido em conta na lista de obstáculos para os casos dos **obstáculos estáticos**.

4.2.0.3 Estratégia de análise de mensagens de obstáculos

Depois de apresentados os diferentes formatos utilizados, o tratamento de cada uma das mensagens recebidas por parte do algoritmo fica sumariado no fluxograma da figura 4.13.

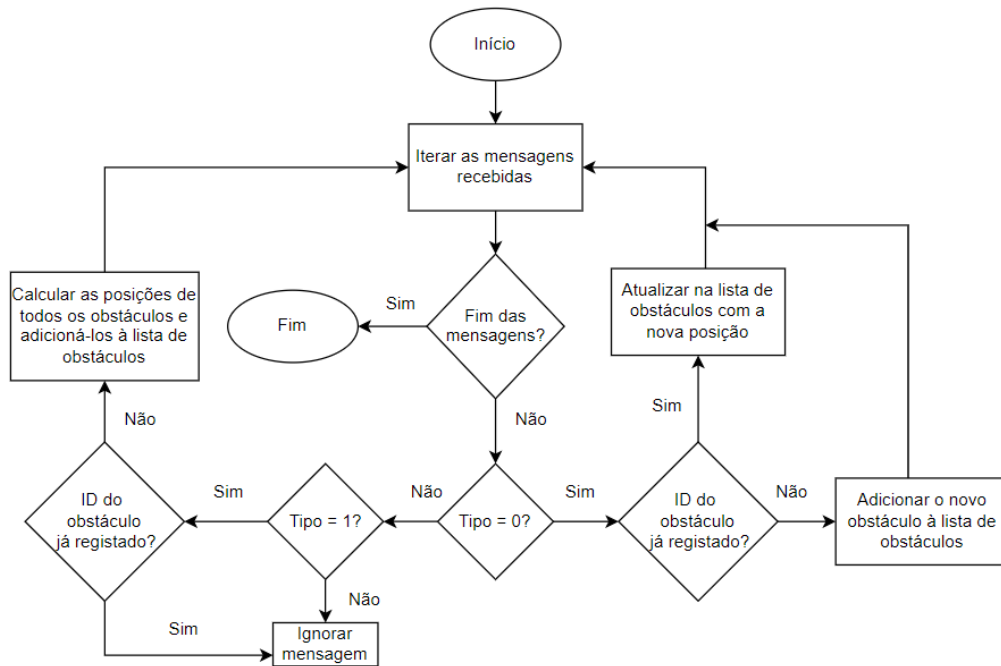


Figura 4.13: Fluxograma do tratamento dos diferentes tipos de mensagens por parte do algoritmo.

RESULTADOS

5.1 Simulações em Computador

As simulações em computador foram feitas na versão 1.59.1 do Visual Studio Code num Intel Core I7 (8ª geração) com um CPU de 1.8 GHz, 8Gb de RAM com o sistema operativo Windows 10. Serão apresentadas num referencial cartesiano com o valor da Latitude (°) no eixo das abcissas e a Longitude (°) no eixo das ordenadas. Para cada simulação, encontra-se representado com uma seta a direção do vento considerado.

5.1.1 Parâmetros iniciais e constantes

Ao longo das simulações e dos testes realizados, os parâmetros foram ajustados, obtendo os valores apresentados na tabela 5.1.

Tabela 5.1: Tabela de constantes

Parâmetro	Significado	Valor
k_atrativa	Fator de escalamento da força atrativa	0.005
separacao_gradientes	Distância a partir da qual se passa a considerar um gradiente atrativo cónico em vez do parabólico [m]	3
raio_chegada	Raio a partir da qual se considera que o veleiro chegou ao destino [m]	2
k_repulsiva	Fator de escalamento da força repulsiva	7
dist_efeito	Distância a partir da qual a força repulsiva começa a exercer influência [m]	3
zona_nao_navegavel	Representa o ângulo que cobre a zona não navegável [°]	90

5.1.2 Cenários sem obstáculos

Nesta primeira secção, serão apresentadas simulações obtidas para diferentes cenários sem obstáculos. Para cada simulação considerou-se um vento constante sempre na mesma direção.

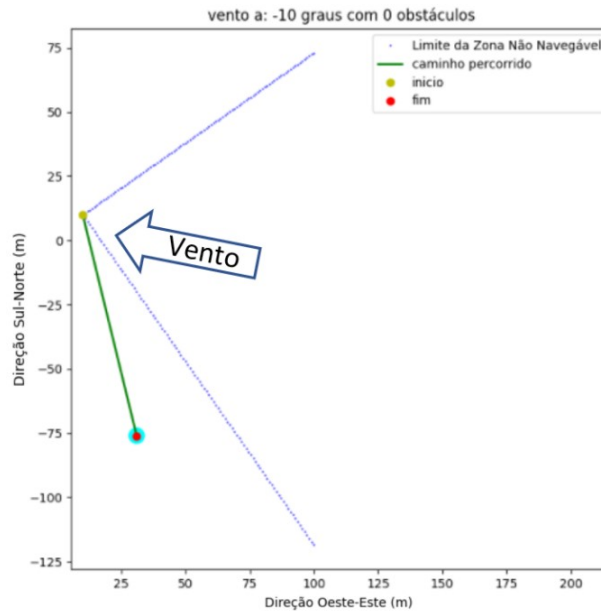


Figura 5.1: Simulação para um cenário com 0 obstáculos e com vento a -10° .

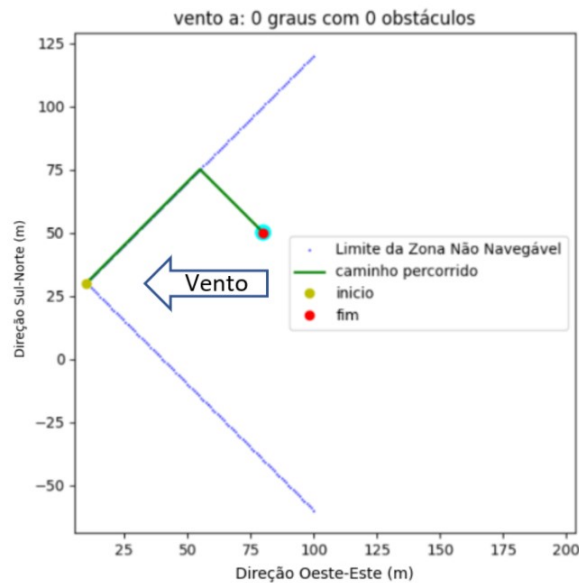


Figura 5.2: Simulação para um cenário com 0 obstáculos e com vento a 0°

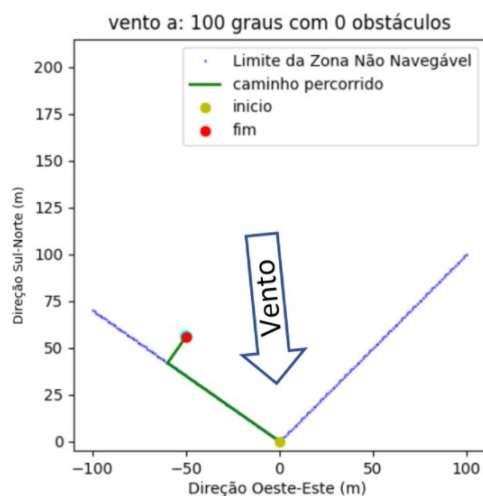


Figura 5.3: Simulação para um cenário com 0 obstáculos e com vento a 100°

Analisando as simulações apresentadas nas figuras 5.2, 5.3, verifica-se que, para os casos em que o ponto de chegada (ponto vermelho) se encontra dentro da ZNN, existe a necessidade de uma navegação na direção contrária ao vento (à bolina cerrada), onde o trajeto obtido é composto por dois troços que se sobrepõe aos limites da ZNN, separados por uma mudança de bordo, como esperado.

Nos cenários em que o ponto de chegada se encontra fora da ZNN, como é o caso da figura 5.1, e, não existindo qualquer tipo de restrição em termos de navegação (vento ou obstáculos), obtém-se uma reta como trajeto, que resulta exclusivamente da força atrativa.

5.1.3 Cenários com 1 obstáculo

Serão apresentadas seguidamente simulações relativamente a cenários com 1 obstáculo dentro e fora da ZNN.

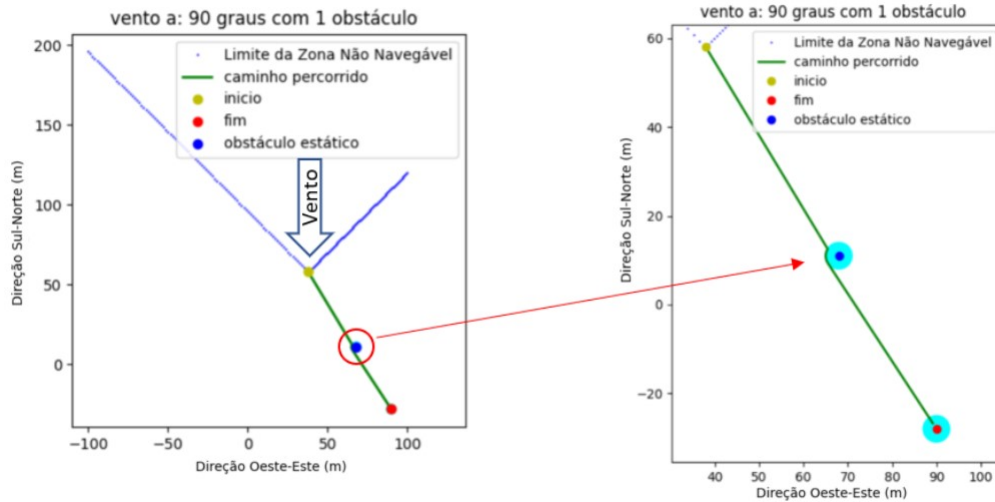


Figura 5.4: Simulação de um cenário com 1 obstáculo e com vento a 90°.

Encontra-se representado um cenário com um obstáculo e vento na direção dos 90° na figura 5.4. Como se pode observar no trajeto obtido, quando o veleiro entra na área de influência do obstáculo (área de cor azul-marinho), passa a ter exercido em si uma força repulsiva, que somada com a atrativa, faz com que o veleiro “contorne” o obstáculo, evitando assim uma colisão.

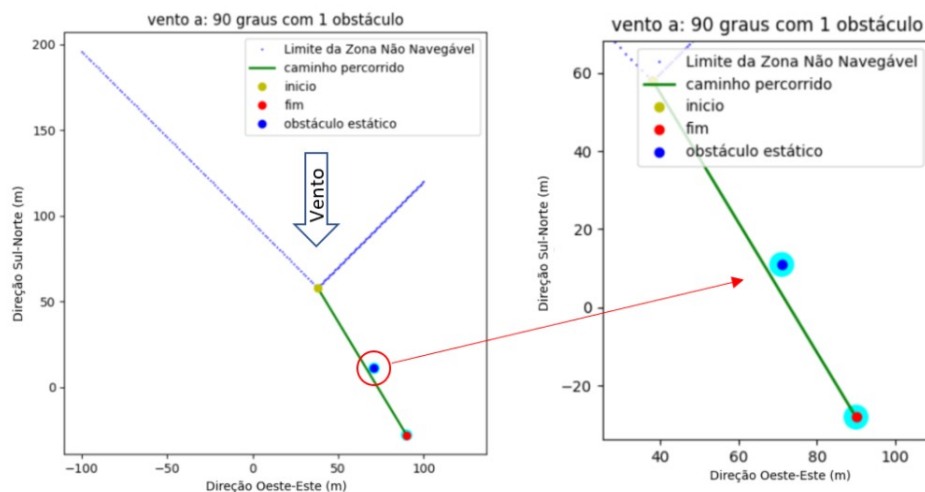


Figura 5.5: Simulação para um cenário com 1 obstáculo e com vento a 90°.

Muito semelhante ao cenário anterior, com a diferença da posição do obstáculo, o trajeto obtido é uma reta, pois como se pode observar na imagem da direita da figura 5.5, o veleiro nunca chega a “entrar” na zona de influência do obstáculo, logo, este não exerce qualquer tipo de força repulsiva no mesmo.

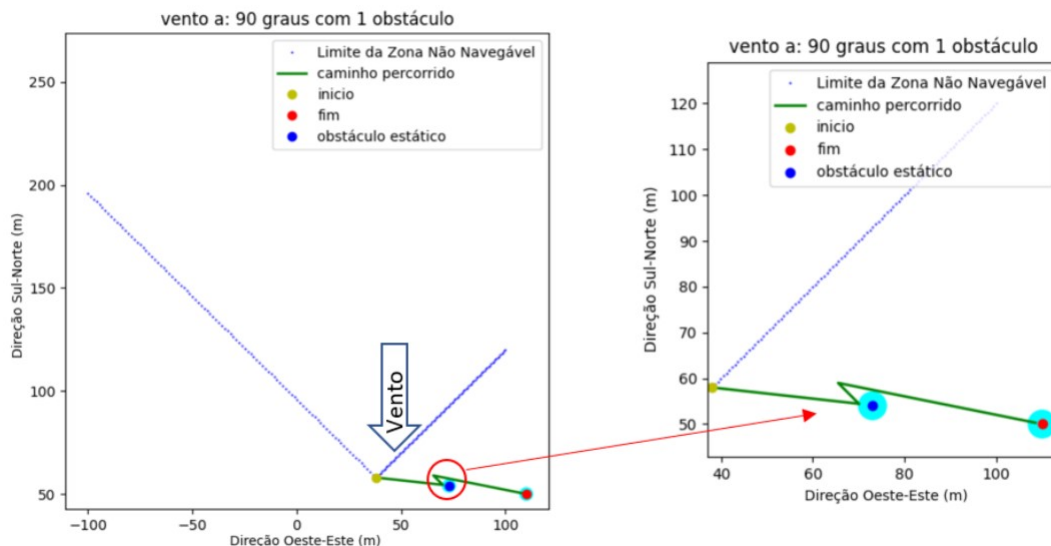


Figura 5.6: Simulação para um cenário com 1 obstáculo e com vento a 90°.

Um cenário onde o rumo do veleiro, ao contornar um obstáculo incorre na ZNN, pode ser observado na figura 5.6. Nestes casos, e como foi explicado, o veleiro realiza uma mudança de bordo quando em contacto com a área de influência do obstáculo, fazendo uma nova mudança em direção ao obstáculo quando estiver a uma distância de segurança do mesmo.

Serão seguidamente apresentadas simulações obtidas para cenários com um obstáculo **dentro da ZNN**.

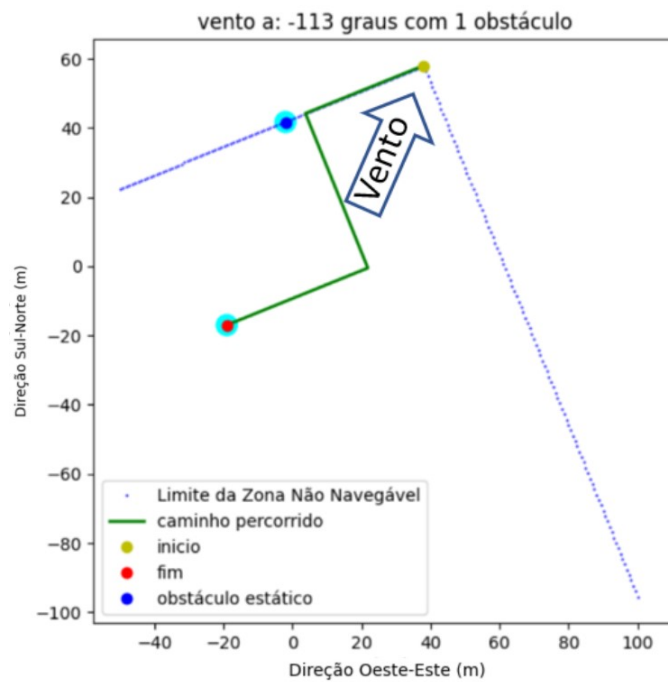


Figura 5.7: Simulação para um cenário com 1 obstáculo e com vento a -113° .

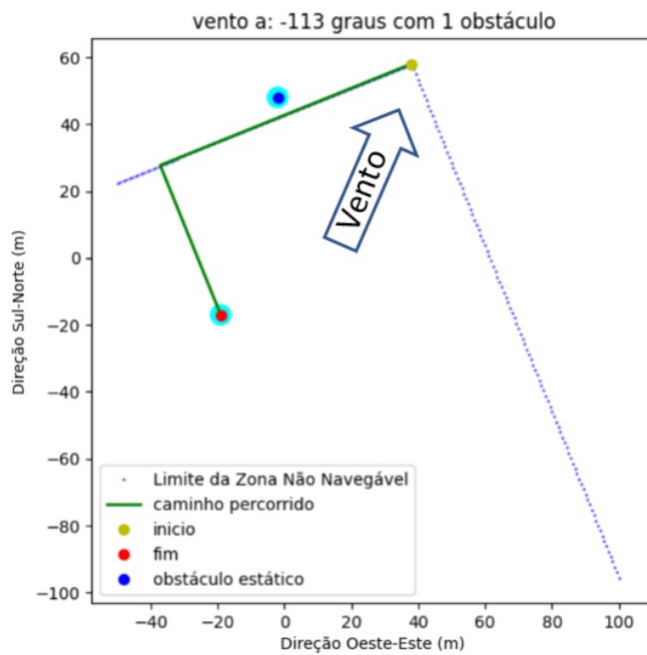


Figura 5.8: Simulação para um cenário com 1 obstáculo e com vento a -113° .

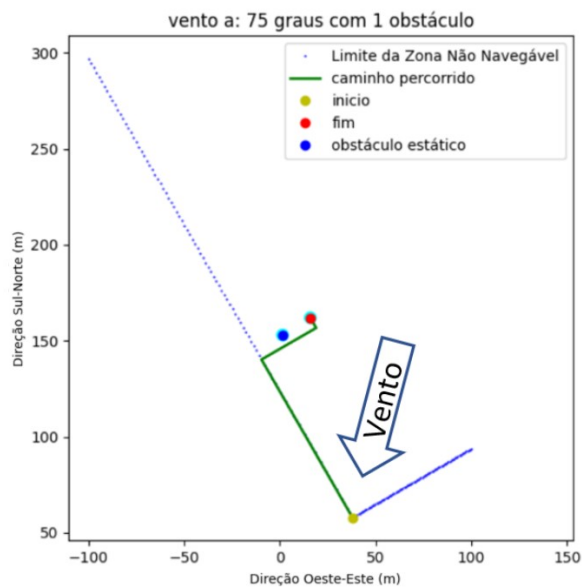


Figura 5.9: Simulação para um cenário com 1 obstáculo e com vento a 75°.

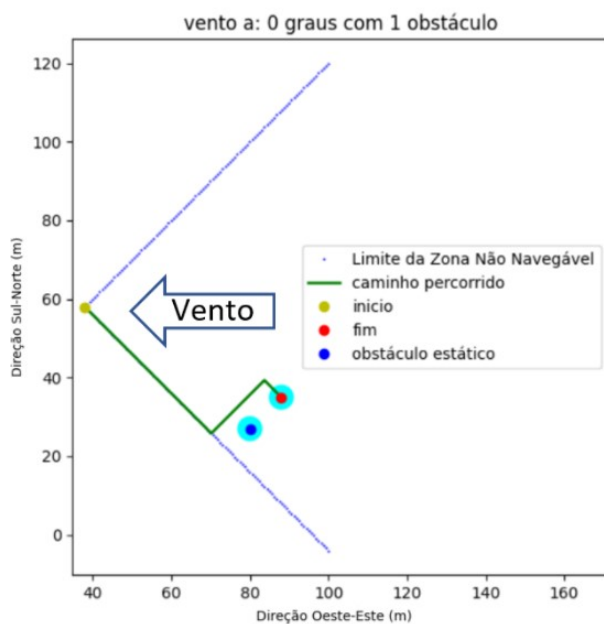


Figura 5.10: Simulação para um cenário com 1 obstáculo e com vento a 0°.

Na figura 5.7, encontra-se representado um cenário onde é necessário evitar uma colisão na 1ª zona da ZNN. Quando o veleiro se aproxima do obstáculo é realizada uma mudança de bordo. Esta manobra é realizada com alguma distância de segurança, distância esta que se foi ajustando ao longo da realização dos testes.

Como foi já visto, a mudança de bordo para evitar um obstáculo só é feita quando este

se encontra no caminho do veleiro, suficientemente próximo e caso exista risco de colisão, sendo este caso representado na figura 5.8 .

Nas figuras 5.9,5.10 encontram-se representados cenários onde é necessário evitar uma colisão na 2ª zona da ZNN, havendo para estes casos uma abordagem de previsão, resultando numa mudança de bordo antecipada.

No final, conclui-se que o veleiro alcança sempre o ponto de chegada obtendo um trajeto correto, evitando sempre os obstáculos em diferentes posições e zonas da ZNN, através de mudanças de bordo.

5.1.4 Cenários com aglomerados de obstáculos

Nesta secção serão apresentadas simulações com um obstáculo que é decomposto, formando um conjunto de obstáculos estáticos, em diferentes posições.

Considerando agora um obstáculo com dimensões de 20 metros de comprimento por 5 metros de largura, o que resulta num conjunto de 50 obstáculos estáticos.

São inicialmente apresentadas simulações onde o conjunto de obstáculos se encontra na 1ª zona da ZNN, variando a posição e orientação do conjunto assim como a direção do vento. Quando na presença de obstáculos nesta zona (caso da figura 5.11, 5.12 e 5.13) é realizada uma mudança de borda quando o veleiro se encontra próximo de um dos obstáculos virtuais existentes, com o objectivo de evitar uma colisão, e uma segunda mudança de bordo em direção ao ponto de chegada.

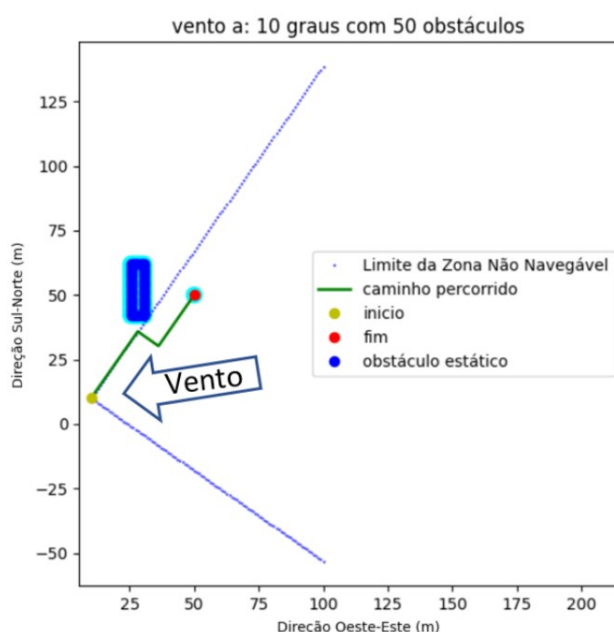


Figura 5.11: Simulação para um cenário com 50 obstáculos e com vento a 10°.

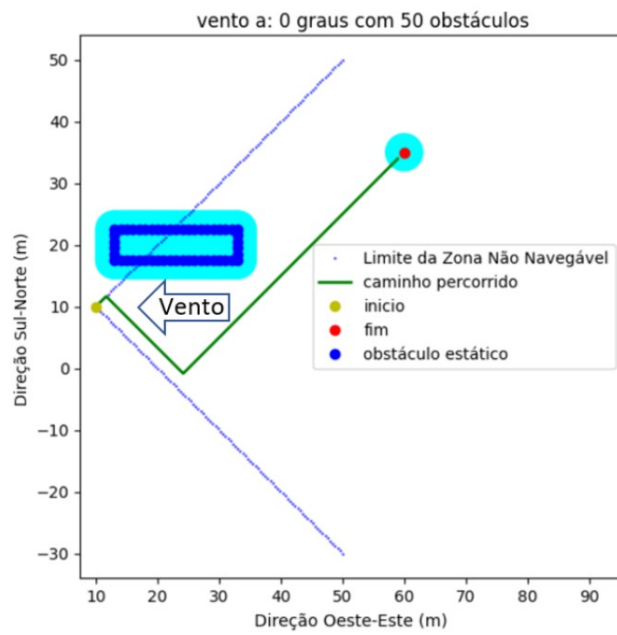


Figura 5.12: Simulação para um cenário com 50 obstáculos e com vento a 0°.

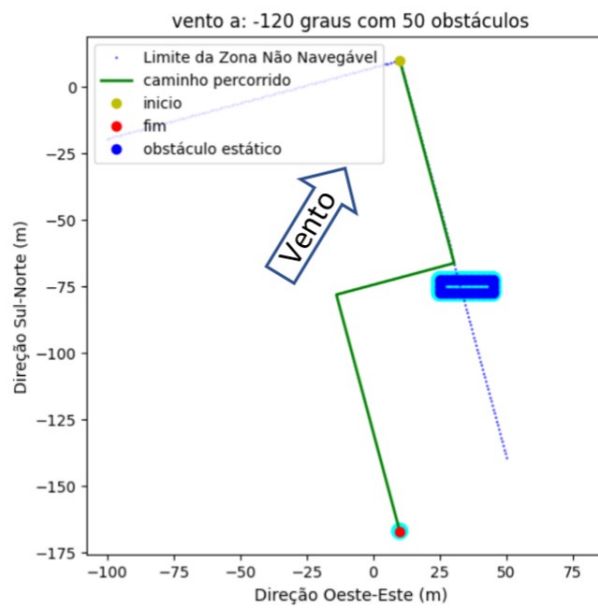


Figura 5.13: Simulação para um cenário com 50 obstáculos e com vento a -120°.

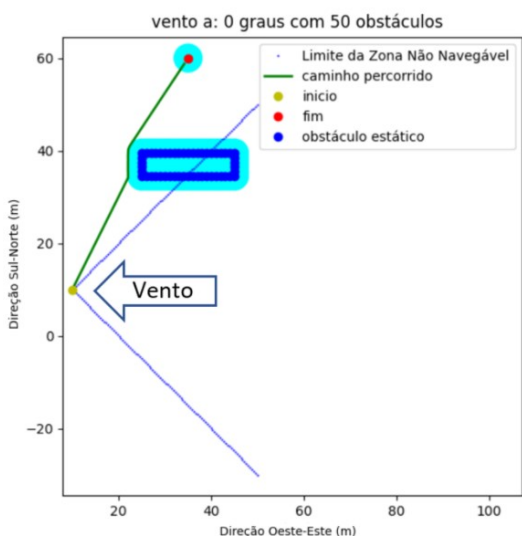


Figura 5.14: Simulação para um cenário com 50 obstáculos e com vento a 0° .

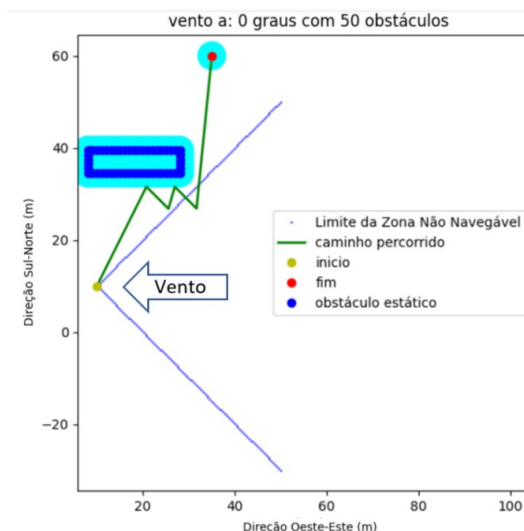


Figura 5.15: Simulação para um cenário com 50 obstáculos e com vento a 0° .

O resultado de uma simulação de cenário onde o ponto de chegada se encontra fora da ZNN, onde o veleiro contorna os obstáculos sem incorrer na ZNN, chegando ao destino pretendido, encontra-se representado na figura 5.14.

Na figura 5.15 é apresentado o mesmo cenário, alterando a localização do ponto de destino. Desta forma, o veleiro para contornar o conjunto de obstáculos incorrerá na ZNN. Quando este se encontra próximo dos obstáculos, é feita a 1ª manobra, memorizando o index do obstáculo, quando a uma distância de segurança desse mesmo obstáculo, o caminho em direção ao ponto de chegada é resumido, no entanto, ao verificar que existe perigo realiza outra manobra, repetindo todo o processo. No final, resume o caminho em direção ao ponto de destino, agora sem quaisquer obstáculos no caminho.

5.1.5 Cenário com obstáculos móveis

Para o caso das simulações com obstáculos móveis, foi criada uma função responsável por ir alterando a posição do obstáculo móvel na mensagem enviada. Não tendo sido considerada qualquer regra de navegação no encontro de veleiros, ou com outras embarcações, foi sempre assumido que os obstáculos móveis tinham prioridade e direito a rumo. Serão apresentados resultados já com o caminho final obtido, sendo adicionadas figuras que, para além de representarem uma atualização da posição do obstáculo em questão, indicam o caminho percorrido pelo mesmo.

No primeiro cenário apresentado, o veleiro da faculdade começa na posição de coordenadas $(10^{\circ}, 10^{\circ})$ e o obstáculo móvel (um navio neste caso) é inicialmente localizado na posição de coordenadas $(10^{\circ}, 30^{\circ})$. Como o ponto de chegada (fim) se encontra dentro da ZNN, o veleiro começa por navegar à bolina sobre o limite da zona superior. O obstáculo móvel, sendo um navio, não tem qualquer restrição com a direção do vento, seguindo o

trajeto apresentado pelas setas. O círculo vermelho representa o ponto em que o veleiro se encontra perto do navio, pelo que, para evitar uma colisão realiza uma mudança de bordo. O navio prossegue o seu caminho, e o veleiro, quando a sua próxima direção de navegação já não pertence à ZNN realiza outra mudança de bordo em direção ao ponto de chegada, como se pode observar na figura 5.16.

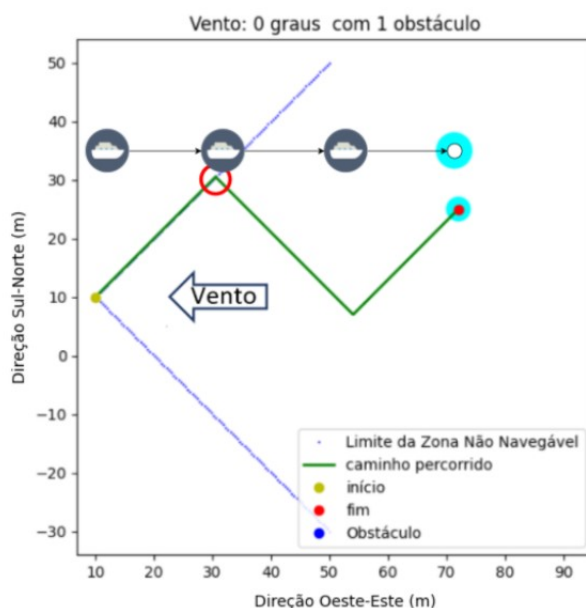


Figura 5.16: Simulação para um cenário com 1 obstáculo e com vento a 0°.

Atendendo agora a um cenário com dois obstáculos móveis, um veleiro e um navio de carga. Analisando o trajeto obtido verifica-se que existem três importantes momentos (assinalados por círculos coloridos) no planeamento da rota. Mais uma vez, tendo um ponto de chegada (fim) numa ZNN, é necessário navegar à bolina cerrada, neste caso, sobre o limite inferior (figura 5.17).

A 1ª mudança de bordo (assinalada pelo círculo a vermelho) é devido ao veleiro obstáculo quando este se aproxima do nosso veleiro, estando perante uma situação onde é necessário evitar uma colisão na 1ª zona. A 2ª mudança de bordo (assinalada pelo círculo a roxo), representa um cenário onde é necessário evitar uma colisão na 2ª zona, verifica-se que o navio de carga no 2º momento em que sua posição é atualizada, se encontra sobre o trajeto que seria seguido pelo veleiro em condições normais. A 3ª mudança de bordo (assinalada pelo círculo a amarelo), representa novamente um cenário onde é necessário evitar uma colisão na 2ª zona, pois o navio de carga no 3º momento em que atualiza a sua posição encontra-se novamente sobre o trajeto que seria seguido pelo veleiro. Por último, é realizada uma 4ª mudança de bordo (assinalada pelo círculo a azul) em direção ao ponto de chegada, mas agora sem ser por influência de qualquer obstáculo.

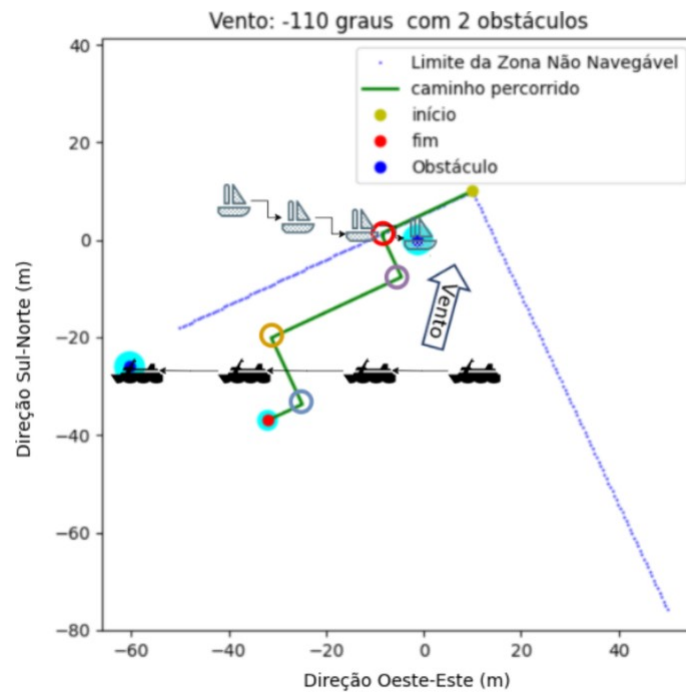


Figura 5.17: Simulação para um cenário com 2 obstáculos e com vento a -110° .

5.1.6 Cenários com vários obstáculos

Nesta última secção de resultados, são apresentadas simulações (figuras 5.18, 5.19, 5.20 e 5.21) que combinam todos os resultados apresentados anteriormente, variando o número, a localização e o tipo de obstáculos presentes, tornando o ambiente de navegação simulado o mais dinâmico possível, pondo à prova toda a robustez do algoritmo desenvolvido.

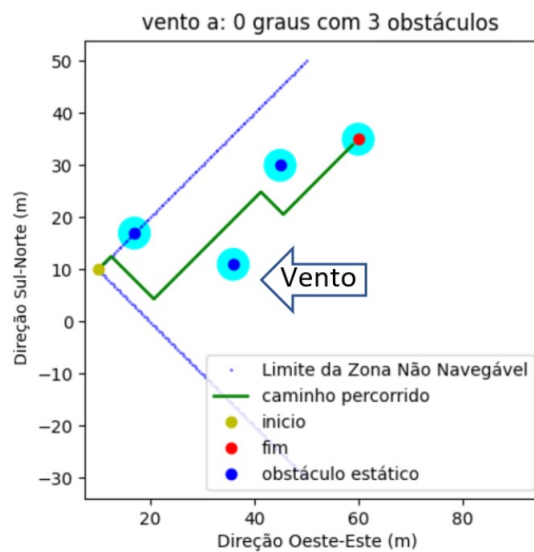


Figura 5.18: Simulação para um cenário com 3 obstáculos e com vento a 0°

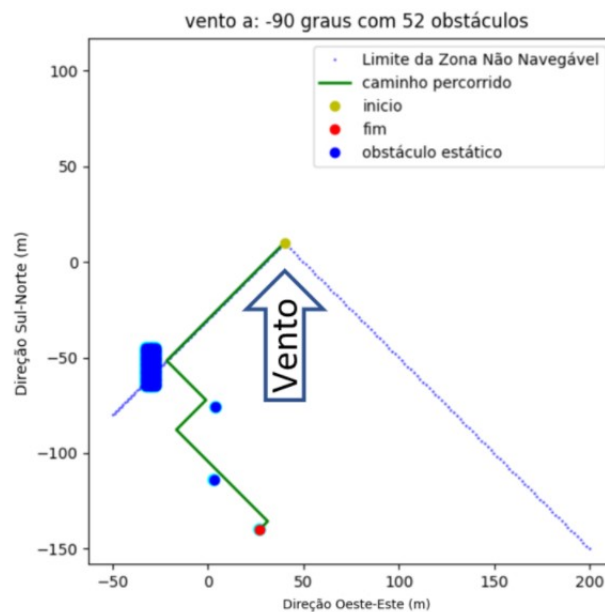


Figura 5.19: Simulação para um cenário com 52 obstáculos e com vento a -90° .

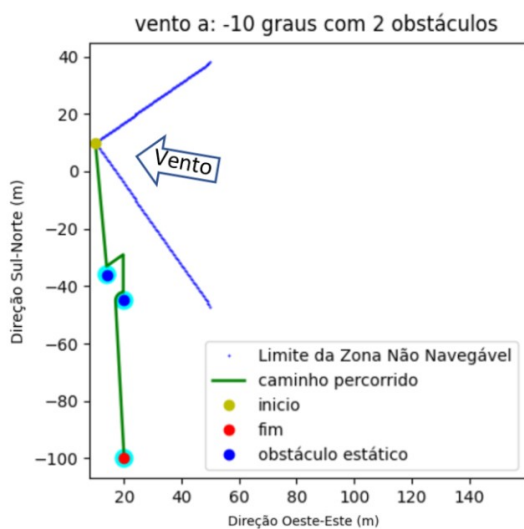


Figura 5.20: Simulação para um cenário com 2 obstáculos e com vento a -10° .

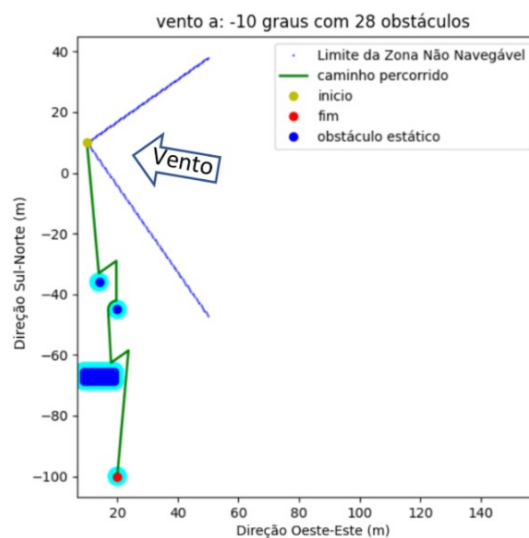


Figura 5.21: Simulação para um cenário com 28 obstáculos e com vento a -10° .

Por último, é apresentada uma simulação de um cenário com um navio de carga e um outro obstáculo. A 1ª mudança de bordo (assinalada pelo círculo vermelho) deve-se à existência do obstáculo no caminho do veleiro (perigo de colisão na 1ª zona), a 2ª mudança de bordo (assinalada pelo círculo a amarelo) é devido ao navio, que no 2º momento em que a sua posição é atualizada, obtém-se um cenário de perigo de colisão na 2ª zona. A 3ª e última mudança de bordo (assinalada pelo círculo a roxo) é feita em direção ao ponto

de chegada sem influência de qualquer obstáculo (figura 5.22).

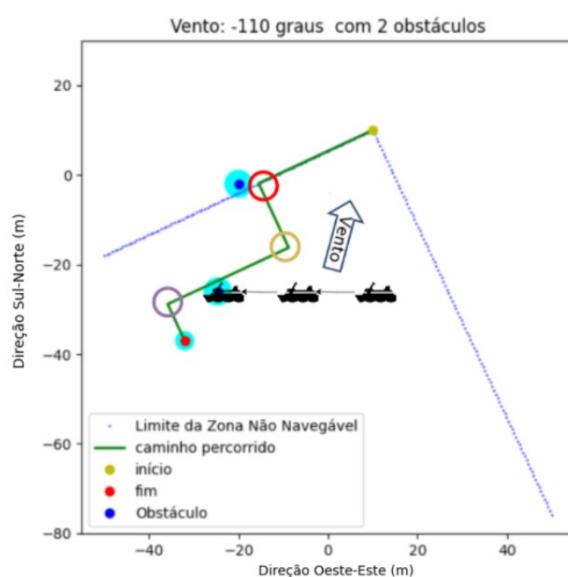


Figura 5.22: Simulação para um cenário com 2 obstáculos e com vento a -110° .

5.1.7 Comparação de Resultados

Não foi possível comparar os resultados obtidos nesta dissertação com os resultados obtidos em trabalhos relacionados, como por exemplo: [Plumet, Pêtrès, Gas e Ieng 2014] ou [Petres, Romero-Ramirez e Plumet 2011], pois nos trabalhos referidos é apresentado um número reduzido de simulações, não testando grande parte dos cenários apresentados neste capítulo.

5.2 Validações laboratoriais

De modo a testar o algoritmo com um cenário mais próximo da realidade e do tema proposto, utilizou-se um **Arduino Uno** para simular o comportamento dos obstáculos.

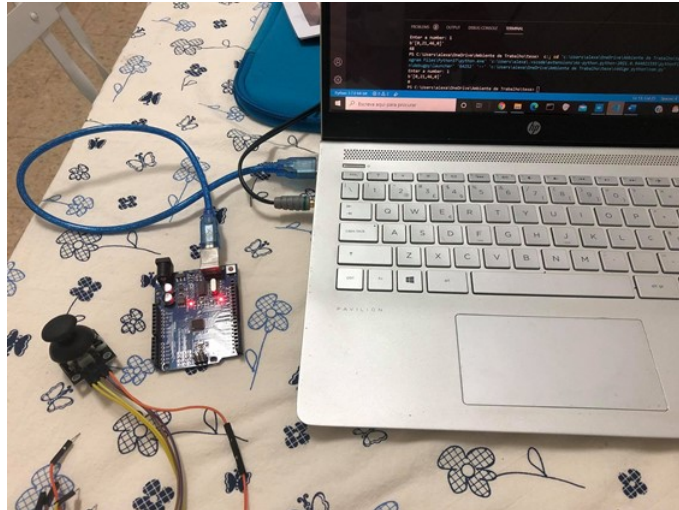


Figura 5.23: Ligação entre Arduino e computador.

Do lado do Arduino foi implementado um ciclo onde eram enviadas mensagens seguidas (com os diferentes formatos já descritos na dissertação). O esquema de ligação e de comunicação entre o Arduino e o algoritmo encontra-se descrito na figura 5.24.

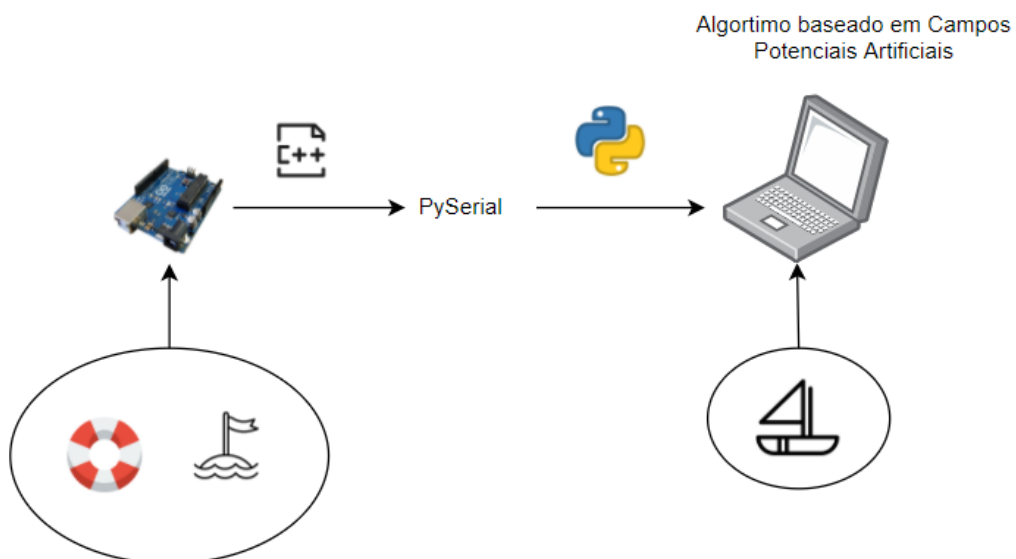


Figura 5.24: Esquema da Ligação entre Arduino e computador.

Para estas validações tornou-se necessário desenvolver funções responsáveis pela leitura da informação enviada pelo Arduino (obstáculo) e o tratamento da mesma, nomeadamente identificar os diferentes campos da mensagem de modo a atualizar a lista de obstáculos.

De seguida serão apresentadas algumas simulações com a mensagem enviada pelo Arduino e o respetivo resultado.

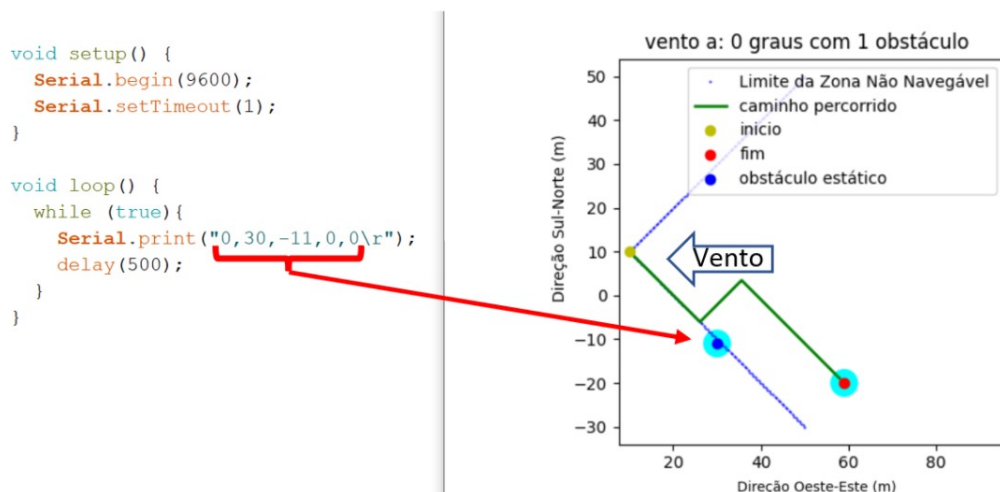


Figura 5.25: Simulação para um cenário com 1 obstáculo e com vento a 0°.

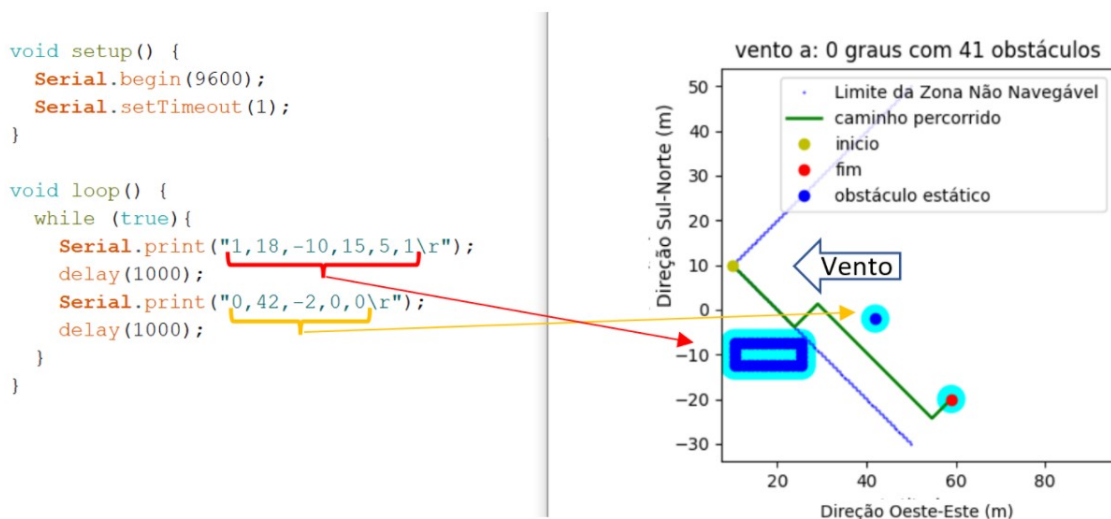


Figura 5.26: Simulação para um cenário com 41 obstáculos e com vento a 0°.

Para cada simulação apresentada, no lado esquerdo da figura encontra-se ilustrado o

código que fora implementado no Arduino para obter os resultados da simulação apresentados ao lado. Note-se ainda que, a mensagem enviada encontra-se assinalada na interface do Arduino, com uma seta correspondente a esse mesmo obstáculo no resultado da simulação obtido pelo algoritmo.

Com as simulações apresentadas nas figuras 5.25 e 5.26, pretende-se demonstrar que as funções responsáveis pela análise e extracção dos diferentes campos das mensagens recebidas funcionam corretamente, e que o algoritmo funciona igualmente de maneira correta ao utilizar um microcontrolador a simular o comportamento de obstáculos

No final, conclui-se que os resultados estão de acordo com o esperado, com uma navegação sem colisões até ao ponto de chegada, na presença de ambientes dinâmicos.

CONCLUSÕES E TRABALHOS FUTUROS

Neste capítulo serão apresentadas as conclusões referentes ao trabalho desenvolvido, assim como propostas para trabalhos futuros.

6.1 Conclusões

Esta dissertação tinha como principal objetivo o desenvolvimento de um algoritmo baseado em campos potenciais artificiais capaz de realizar o planeamento de uma rota de um ponto de partida até ao destino pretendido, através de dados recebidos em tempo real, como a direção do vento, posição atual do veleiro, o próximo ponto de chegada e informação (mensagens) vinda de obstáculos estáticos e móveis presentes no ambiente de navegação.

A primeira conclusão a ser retirada (e uma das principais vantagens desta abordagem) é o facto da estratégia baseada em campos potenciais artificiais ser intuitiva, simples de implementar e capaz de realizar ajustes no planeamento do caminho num curto espaço de tempo, evitando colisões. Concluiu-se também, que o tempo computacionalmente necessário para calcular todo o caminho sofre muito pouca alteração ao variar o número total de obstáculos.

Uma das principais dificuldades sentidas, residiu no facto do cenário de navegação se tratar de um ambiente dinâmico e imprevisível, onde, ao longo do desenvolvimento do algoritmo, foi adquirida uma certa sensibilidade sobre quais os cenários e as posições dos obstáculos que causavam problemas, criando limitações em algumas das estratégias abordadas, nomeadamente ao evitar obstáculos dentro e fora da ZNN, e ao lidar com aglomerados de obstáculos muito juntos.

Para ambientes de navegação simples, o algoritmo fornece bons resultados. Os problemas surgem quando se torna necessário ter em conta novas regras e exceções para lidar com obstáculos e zonas de navegação proibidas, tendo de garantir ao mesmo tempo o correto funcionamento nas situações de navegação menos complexas previamente consideradas.

Durante as simulações foi sentida ainda alguma dificuldade no ajuste dos diferentes

parâmetros de cada campo, até se chegarem a valores conducentes a bons desempenhos e obter um trajeto final correto.

No caso dos cálculos considerando diferentes valores de ângulos (ao utilizar a função trigonométrica $\arctg2$), foi necessário adotar uma abordagem específica para lidar com as situações em que existia uma passagem pelos ângulos de $(-180^\circ, 180^\circ)$.

No final, e através das simulações e dos resultados apresentados, é possível concluir que a estratégia desenvolvida apresenta resultados de simulações validadas com sucesso, realizando o planeamento de uma rota navegável, cumprindo as regras e evitando obstáculos estáticos e móveis como esperado, sendo ainda apresentada uma estratégia para evitar obstáculos e aglomerados de obstáculos dentro e fora da ZNN.

6.2 Trabalhos Futuros

Para possíveis trabalhos a serem desenvolvidos no futuro, poderão ser consideradas as sugestões seguidamente apresentadas.

- Em primeiro lugar, introduzir uma estratégia responsável por lidar com o problema dos mínimos locais nos cálculos realizados, de modo a tornar toda a estratégia mais robusta.
- Associar aos obstáculos um “tempo de vida”, de modo a serem removidos da lista de obstáculos quando deixarem de ser relevantes para a navegação.
- Considera-se relevante também melhorar a estratégia para evitar obstáculos dentro da ZNN, de forma a obter dados de uma forma mais precisa sobre a posição dos mesmos, e tornar a distância a que o veleiro começa a verificar se existe risco de colisão variável e dependente do tipo de obstáculos presentes.
- Em relação aos aglomerados de obstáculos, aperfeiçoar a estratégia de forma a que seja possível lidar com cenários com aglomerados de obstáculo em diferentes orientações, já que nesta dissertação foram consideradas apenas orientações nas direções dos pontos cardeais.
- Nos cenários onde ao evitar um obstáculo o veleiro incorre numa zona não navegável, o desempenho do algoritmo proposto poderá ser melhorado, sendo necessário redefinir a estratégia utilizada, pois a apresentada funciona mas com limitações, conduzindo a resultados não otimizados.
- Validar o algoritmo proposto num ambiente de condições reais, integrando o algoritmo no sistema de controlo de um veleiro autónomo.
- Por fim, introduzir estratégias de modo a que o veleiro cumpra as regras de navegação e prioridades (COLRGEES).

BIBLIOGRAFIA

- Alves, J. C. e N. A. Cruz (2008). «FASt - An autonomous sailing platform for oceanographic missions». Em: *Oceans 2008* October. DOI: [10.1109/OCEANS.2008.5152114](https://doi.org/10.1109/OCEANS.2008.5152114) (ver p. 16).
- Anderson, B. D. (2001). «The Physics of sailing». Em: *Physics Education* 36.5, pp. 388–93. ISSN: 0031-9120. URL: https://gewa.gsfc.nasa.gov/clubs/sailing/RESOURCES/Physics%7B%5C_%7Dof%7B%5C_%7Dsailing.pdf (ver p. 15).
- Anjos, F. (2020). «Introdução de melhorias em controlador para navegação autónoma de pequeno veleiro». Dissertação de Mestrado. Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa. URL: <http://hdl.handle.net/10362/124665> (ver p. 70).
- Azmi, M. Z. e T. Ito (2020). «Artificial potential field with discrete map transformation for feasible indoor path planning». Em: *Applied Sciences (Switzerland)* 10.24, pp. 1–13. ISSN: 20763417. DOI: [10.3390/app10248987](https://doi.org/10.3390/app10248987) (ver pp. 38, 45, 46).
- Bisbee, J. e P. Halloran (1995). «Technical information supplied by Harold “ Hatch ” Brown , MIT Sailing Master ; MITNA Staff ; and Members .» Em: (ver p. 13).
- Centre, W. C. L. S. (2015). «Manual de Iniciação Vela de Cruzeiro». Em: Acedido em: 2021-12-09. URL: https://www.azoresuperyachtservices.pt/images/Download%7B%5C_%7Dpt/MANUAL%20DE%20VELA%20DE%20CRUZEIRO/Manual%20de%20Vela%20de%20Cruzeiro%20-%20Lisbon%20Sailing%20Center.pdf (ver p. 5).
- charterworld (2021). Acedido em: 2020-10-22. URL: <https://www.charterworld.com> (ver p. 14).
- Choset, H. (2006). «Robotic Motion Planning: Potential Functions». Em: pp. 1–54. URL: <http://voronoi.sbp.ri.cmu.edu/%7B~%7Dmotion> (ver p. 35).
- Colin Sauze, R. Z. (2021). Acedido em: 2021-02-11. URL: <https://sourceforge.net/p/microtransat/wiki/2021> (ver p. 17).
- Cruz, N. A. e J. C. Alves (2008). «Autonomous sailboats: An emerging technology for ocean sampling and surveillance». Em: *Oceans 2008* April 2014. DOI: [10.1109/OCEANS.2008.5152113](https://doi.org/10.1109/OCEANS.2008.5152113) (ver p. 1).

- Daniel, K. et al. (2010). «Theta*: Any-angle path planning on grids». Em: *Journal of Artificial Intelligence Research* 39.January, pp. 533–579. ISSN: 10769757. DOI: [10.1613/jair.2994](https://doi.org/10.1613/jair.2994) (ver pp. 18, 20).
- Davis, M. R. e D. S. Holloway (2007). «A comparison of the motions of trimarans, catamarans and monohulls». Em: *Australian Journal of Mechanical Engineering* 4.2, pp. 183–195. ISSN: 1448-4846. DOI: [10.1080/14484846.2007.11464525](https://doi.org/10.1080/14484846.2007.11464525) (ver p. 6).
- De, L. e G. Guglieri (2012). «Advanced Graph Search Algorithms for Path Planning of Flight Vehicles». Em: *Recent Advances in Aircraft Technology*. DOI: [10.5772/37033](https://doi.org/10.5772/37033) (ver p. 18).
- Dijkstra, E. W. (1959). «Dijkstra.Pptx». Em: *Numer. Math.* 271, pp. 269–271 (ver p. 18).
- Dolgov, D. et al. (2010). «Path planning for autonomous vehicles in unknown semi-structured environments». Em: *International Journal of Robotics Research* 29.5, pp. 485–501. ISSN: 02783649. DOI: [10.1177/0278364909359210](https://doi.org/10.1177/0278364909359210) (ver p. 35).
- dreamstime (2021). Acedido em: 2020-10-20. URL: <https://www.dreamstime.com> (ver pp. 12, 13).
- Drozдов, V. N., V. A. Kim e L. B. Lazebnik (2011). [*Modern approach to the prevention and treatment of NSAID-gastropathy*]. 2, pp. 106–110. ISBN: 0131038052 (ver p. 19).
- EMSA (2020). «Annual Overview Of Marine Casualties And Incidents 2020». Em: ISSN: 1098-6596. arXiv: [arXiv : 1011 . 1669v3](https://arxiv.org/abs/1669v3). URL: https://safety4sea.com/wp-content/uploads/2019/11/EMSA-Annual-Overview-of-Marine-Casualties-and-Incidents-2019-2019%7B%5C_%7D11.pdf (ver p. 1).
- Erckens, H. et al. (2010). «Avalon: Navigation strategy and trajectory following controller for an autonomous sailing vessel». Em: *IEEE Robotics and Automation Magazine* 17.1, pp. 45–54. ISSN: 10709932. DOI: [10.1109/MRA.2010.935792](https://doi.org/10.1109/MRA.2010.935792) (ver p. 20).
- Gomes, L. et al. (2015). «Attracting students to engineering through autonomous sailing yacht development». Em: *Proceedings of the IEEE International Conference on Industrial Technology* 2015-June.June, pp. 3252–3257. DOI: [10.1109/ICIT.2015.7125579](https://doi.org/10.1109/ICIT.2015.7125579) (ver pp. 70, 72).
- Gomes, L. et al. (2015). «Model-based development of an autonomous sailing yacht controller». Em: *Proceedings - 2015 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC 2015*, pp. 103–108. DOI: [10.1109/ICARSC.2015.20](https://doi.org/10.1109/ICARSC.2015.20) (ver p. 70).
- Hong, M. J. e M. R. Arshad (2015). «A Balance-Artificial Potential Field Method for Autonomous Surface Vessel Navigation in Unstructured Riverine Environment». Em: *Procedia Computer Science* 76.Iris, pp. 198–202. ISSN: 18770509. DOI: [10.1016/j.procs.2015.12.341](https://doi.org/10.1016/j.procs.2015.12.341) (ver pp. 23, 26, 27, 33, 34).
- Jansen, A., B. Van Deursen e C. Howe (2012). «The aerodynamics of sailing apparel». Em: *Procedia Engineering* 34, pp. 50–55. ISSN: 18777058. DOI: [10.1016/j.proeng.2012.04.010](https://doi.org/10.1016/j.proeng.2012.04.010). URL: <http://dx.doi.org/10.1016/j.proeng.2012.04.010> (ver p. 12).
- jeanneau (2021). URL: <https://www.jeanneau.com/en/boats/sailboat/4-jeanneau-yachts/653-jeanneau-yachts-60/> (acedido em 15/07/2021) (ver pp. 5, 9–11).

- Jing, W. et al. (2020). «Path Planning and Navigation of Oceanic Autonomous Sailboats and Vessels: A Survey». Em: *Journal of Ocean University of China* 19.3, pp. 609–621. ISSN: 19935021. DOI: [10.1007/s11802-020-4144-7](https://doi.org/10.1007/s11802-020-4144-7) (ver p. 18).
- Khatib, O. (1985). «Real-time obstacle avoidance for manipulators and mobile robots». Em: *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 500–505. ISSN: 10504729. DOI: [10.1109/ROBOT.1985.1087247](https://doi.org/10.1109/ROBOT.1985.1087247) (ver pp. 21, 35, 36, 45).
- Likhachev, M. et al. (2005). «Anytime dynamic a*: An anytime, replanning algorithm». Em: *ICAPS 2005 - Proceedings of the 15th International Conference on Automated Planning and Scheduling*, pp. 262–271 (ver p. 20).
- Liu, Y., B. Cao e H. Li (2021). «Improving ant colony optimization algorithm with epsilon greedy and Levy flight». Em: *Complex & Intelligent Systems* 7.4, pp. 1711–1722. DOI: [10.1007/s40747-020-00138-3](https://doi.org/10.1007/s40747-020-00138-3). URL: <https://doi.org/10.1007/s40747-020-00138-3> (ver pp. 20, 22).
- Lyu, H. e Y. Yin (2019). «COLREGS-Constrained Real-Time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields». Em: *Journal of Navigation* 72.3, pp. 588–608. ISSN: 14697785. DOI: [10.1017/S0373463318000796](https://doi.org/10.1017/S0373463318000796) (ver pp. 23, 25, 27, 28, 33).
- Maria Medeiros Alves, J. (2017). «Path Planning and Collision Avoidance Algorithms for Small RPAS Aerospace Engineering Examination Committee». Dissertação de Mestrado. Instituto Superior Técnico (ver pp. 18, 20, 21).
- Matoui, F., B. Boussaid e M. N. Abdelkrim (2016). «Local minimum solution for the potential field method in multiple robot motion planning task». Em: *16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering, STA 2015* December, pp. 452–457. DOI: [10.1109/STA.2015.7505223](https://doi.org/10.1109/STA.2015.7505223) (ver p. 39).
- mcgill (2021). Acedido em: 2020-12-1. URL: <https://www.cs.mcgill.ca> (ver p. 21).
- Mendonça, S. (2013). «The "sailing ship effect": Reassessing history as a source of insight on technical change». Em: *Research Policy* 42.10, pp. 1724–1738. ISSN: 00487333. DOI: [10.1016/j.respol.2012.12.009](https://doi.org/10.1016/j.respol.2012.12.009). URL: <http://dx.doi.org/10.1016/j.respol.2012.12.009> (ver p. 12).
- microtransat (2021). Acedido em: 2020-11-20. URL: <https://www.microtransat.org/history.php> (ver p. 17).
- Naeem, W., S. C. Henrique e L. Hu (2016). «A Reactive COLREGs-Compliant Navigation Strategy for Autonomous Maritime Navigation». Em: *IFAC-PapersOnLine* 49.23, pp. 207–213. ISSN: 24058963. DOI: [10.1016/j.ifacol.2016.10.344](https://doi.org/10.1016/j.ifacol.2016.10.344). URL: <http://dx.doi.org/10.1016/j.ifacol.2016.10.344> (ver pp. 23, 28, 33, 57).
- Noel, A. et al. (nov. de 2019). «Autonomous Ship Navigation Methods: A Review». Em: DOI: [10.24868/icmet.oman.2019.028](https://doi.org/10.24868/icmet.oman.2019.028) (ver p. 1).
- Park, M. G. e M. C. Lee (2003). «A new technique to escape local minimum in artificial potential field based path planning». Em: *KSME International Journal* 17.12, pp. 1876–1885. ISSN: 1226-4865. DOI: [10.1007/bf02982426](https://doi.org/10.1007/bf02982426) (ver pp. 21, 38).

- Petres, C., M.-A. Romero-Ramirez e F. Plumet (2011). «Modeling and reactive navigation of an autonomous sailboat». Em: pp. 3571–3576. DOI: [10.1109/iroso.2011.6094912](https://doi.org/10.1109/iroso.2011.6094912) (ver pp. 21, 23, 31–34, 48, 95).
- pexels (2021). Acedido em: 2020-10-21. URL: <https://www.pexels.com> (ver pp. 13, 14).
- pinterest (2021). Acedido em: 2020-10-20. URL: <https://www.pinterest.pt/pin/227713324881756082/> (ver p. 8).
- pixabay (2021). Acedido em: 2020-10-5. URL: <https://pixabay.com/photos/catamaran-sailing-ship-ship-sea-326555/> (ver p. 7).
- Plumet, F., H. Saoud e M. D. Hua (2013). «Line following for an autonomous sailboat using potential fields method». Em: *OCEANS 2013 MTS/IEEE Bergen: The Challenges of the Northern Dimension*. DOI: [10.1109/OCEANS-Bergen.2013.6607961](https://doi.org/10.1109/OCEANS-Bergen.2013.6607961) (ver pp. 21, 22, 38).
- Plumet, F. et al. (2014). «Toward an Autonomous Sailing Boat 2015 Romero». Em: pp. 1–11 (ver pp. 21, 23, 28–31, 33, 34, 43, 59, 95).
- Quant, K. (2009). *The basic sailboat racing rules all racers should know*. Acedido em: 2021-10-30. URL: <https://cvmyclub.com/images/pdf/BasicRacingRules.pdf> (ver pp. 6, 7).
- Reshamwala, A. (2013). «Robot Path Planning using An Ant Colony Optimization Approach : A Survey». Em: 2.3, pp. 65–71 (ver p. 21).
- roboticsailing (2021). Acedido em: 2020-11-20. URL: <https://www.roboticsailing.org/index.html> (ver pp. 1, 16, 17).
- Rosen, J. B. (1961). «The Gradient Projection Method for Nonlinear Programming. Part II. Nonlinear Constraints». Em: *Journal of the Society for Industrial and Applied Mathematics* 9.4, pp. 514–532. ISSN: 0368-4245. DOI: [10.1137/0109044](https://doi.org/10.1137/0109044) (ver p. 19).
- Rutledge, K. (2011). Acedido em: 2021-04-25. URL: <https://www.nationalgeographic.org/encyclopedia/jetty/> (ver p. 57).
- sailbot (2021). Acedido em: 2020-11-20. URL: <https://www.sailbot.org/about/> (ver p. 17).
- SailingEurope (2021). Acedido em: 2020-10-5. URL: <https://www.sailingeurope.com/blog/monohull-sailing-yacht-pros-and-cons> (ver p. 6).
- sailnator (2021). Acedido em: 2020-10-20. URL: <http://sailnator.com> (ver p. 10).
- SailRaceCrew (2021). Acedido em: 2020-10-5. URL: <https://sailracecrew.com/sailing/boats/trimaran-lultim-emotion-77ft/> (ver p. 7).
- Saoud, H. e M.-d. Hua (s.d.). «Routing and course control of an autonomous sailboat». Em: (). DOI: [10.1109/ECMR.2015.7324218](https://doi.org/10.1109/ECMR.2015.7324218) (ver p. 20).
- Silva, J. B., C. A. Siebra e T. P. Nascimento (2016). «A New Cost Function Heuristic Applied to A* Based Path Planning in Static and Dynamic Environments». Em: *Proceedings - 12th LARS Latin American Robotics Symposium and 3rd SBR Brazilian Robotics Symposium, LARS-SBR 2015 - Part of the Robotics Conferences 2015*, pp. 37–42. DOI: [10.1109/LARS-SBR.2015.35](https://doi.org/10.1109/LARS-SBR.2015.35) (ver p. 20).

-
- stanford (2020). Acedido em: 2020-12-08. URL: <http://theory.stanford.edu/~amitp/GameProgramming/AStarComparison.html> (ver p. 19).
- (2021). Acedido em: 2020-11-30. URL: <http://theory.stanford.edu/> (ver pp. 18, 19).
- Stelzer, R. e K. Jafarmadar (2011). «History and Recent Developments in Robotic Sailing». Em: *Robotic Sailing*, pp. 2–3. DOI: [10.1007/978-3-642-22836-0_1](https://doi.org/10.1007/978-3-642-22836-0_1) (ver p. 16).
- Thundertiger (2021). Acedido em: 2020-10-5. URL: thundertiger.com/productsdetail.php?id=11 (ver p. 4).
- Ukil, A., V. H. Shah e B. Deck (2011). «Fast computation of arctangent functions for embedded applications: A comparative analysis». Em: *Proceedings - ISIE 2011: 2011 IEEE International Symposium on Industrial Electronics* June, pp. 1206–1211. DOI: [10.1109/ISIE.2011.5984330](https://doi.org/10.1109/ISIE.2011.5984330) (ver p. 49).
- Universal_yachting (2021). Acedido em: 2020-11-30. URL: <https://www.universalyachting.com/sailing-school-points-of-sail/> (ver p. 8).
- Vasconcelos, J. F. F. C. (2015). «Design of Autonomous Surface Vessels». Em: p. 128. URL: <https://pdfs.semanticscholar.org/8047/ca6d92af12032bf47ba81e571c4e4566c071.pdf%7B%5C%7D0Ahttps://fenix.tecnico.ulisboa.pt/downloadFile/563345090415044/Thesis%20-%20Design%20of%20Autonomous%20Surface%20Vessels.pdf> (ver pp. 6, 7).
- Website, N. (2018). URL: <https://www.nauticed.org/sailing-blog/how-do-polar-plots-work-on-a-sailboat/> (acedido em 28/08/2021) (ver p. 48).
- Website, P. D. (2015). URL: <https://www.nwp.usace.army.mil/Media/News-Releases/Article/587629/critical-repairs-beginning-at-mcr-north-jetty/> (acedido em 25/08/2021) (ver p. 57).
- Wikipedia (2021). Acedido em: 2021-02-11. URL: https://en.wikipedia.org/wiki/Distance_from_a_point_to_a_line (ver p. 64).

